

NEALT

Northern European Association for Language Technology

NEALT Proceedings Series No. 57

Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)

March 3 – 4, 2025

Tallinn, Estonia

Editors: Richard Johansson and Sara Stymne

NoDaLiDa/Baltic-HLT 2025

**Joint 25th Nordic Conference on Computational Linguistics
and 11th Baltic Conference on Human Language
Technologies (NoDaLiDa/Baltic-HLT 2025)**

Proceedings of the Conference

March 3-4, 2025

The NoDaLiDa/Baltic-HLT 2025 organizers gratefully acknowledge the support from the following sponsors.



REPUBLIC OF ESTONIA
MINISTRY OF EDUCATION
AND RESEARCH

SPRÅKBANKEN
A research infrastructure for language data

©2025 University of Tartu Library

Front-cover photo: **Rasmus Jurkatam**

Published by:

University of Tartu Library, Estonia
NEALT Proceedings Series, No. 57
Indexed in the ACL Anthology

ISBN: 978-9908-53-109-0
ISSN: 1736-8197 (Print)
ISSN: 1736-6305 (Online)

Volume Editors:
Richard Johansson and Sara Stymne

Message from the General Chair

Welcome to the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025) to be held in beautiful Tallinn, Estonia, on March 2–5, 2025.

It is 48 years since the first NoDaLiDa was held and 21 years since the first Baltic HLT was held. Now, for the first time, the two major conferences on computational linguistics and language technology in the Nordic and Baltic regions have joined forces as a joint event. Both conferences aim to bring together researchers in the Nordic and Baltic countries interested in any aspect related to human language and speech technology. As a joint event, we extended the conference with one extra workshop day so that we have one day before and one day after the main two-day conference. It is a great honor for me to serve as the general chair of this joint event.

We solicited three different types of papers (long, short, and demo papers) and received 127 valid submissions, of which 4 were withdrawn during the process. In total, we accepted 81 papers (acceptance rate: 66%; long papers 65%, short papers: 66%, demos 80%), which will be presented as 43 oral presentations, 34 posters, and 4 demos. More than half of the accepted papers are student papers, in which the first author is a student (29 long, 19 short, and 2 demo papers). Each paper was reviewed by three experts. We are extremely grateful to the 155 Programme Committee members for their detailed and helpful reviews.

The 81 accepted papers are organized into 12 oral sessions and 2 poster and demo sessions. In addition to these regular sessions, the conference program includes three keynote talks. We would like to extend our gratitude to the keynote speakers for agreeing to present their work at NoDaLiDa/Baltic-HLT. Arianna Bisazza from the University of Groningen will talk on the topic of “Not all Language Models Need to be Large: Studying Language Evolution and Acquisition with Modern Neural Networks.” Dirk Hovy from Bocconi University will talk about “The Illusion of Understanding – Unpacking the True Capabilities of Language Models.” Arvi Tavast from the Institute of the Estonian Language will talk about “No Sex, No Future: On the Status of Estonian in a Changing World,” continuing the NoDaLiDa tradition of featuring a presentation about the local language.

The main conference is complemented by 6 workshops on a diverse set of topics. On March 2, preceding the main conference: Resources and representations for under-resourced languages and domains (RESOURCEFUL-2025); Nordic-Baltic Responsible Evaluation and Alignment of Language models (NB-REAL); and The 1st Workshop on Ecology, Environment, and Natural Language Processing (NLP4Ecology). On March 5, after the main conference: Constraint Grammar and Finite State NLP – Rule-based and hybrid methods and tools for user communities; The 13th Workshop on Natural Language Processing for Computer Assisted Language Learning (NLP4CALL); and Automatic Assessment of Atypical Speech (AAAS). The workshop themes illustrate the breadth of topics that can be found in language technology, and we are extremely happy and grateful to the workshop organizers for complementing the main program.

I would like to thank the entire team that made NoDaLiDa/Baltic-HLT possible. I was honored to receive the invitation to serve as the general chair from Jörg Tiedemann and the NEALT board; thank you for trusting me in this role. My deepest gratitude goes to the organizing committee. Thank you to the program chair committee Daniel Hershcovich, Jenna Kanerva, Pierre Lison, and Andrius Utka, for working hard on putting the program together, especially for your great effort in leading the reviewing process and shepherding papers from submission to the final decision. Thank you to the program chair advisors Mark Fišel and Inguna Skadiņa, for your valuable advice about previous editions of NoDaLiDa and Baltic-HLT. Thank you to Richard Johansson for leading the publication efforts that led to this volume, as well as the coordination of the workshop proceedings. Thank you to the workshop chairs,

Normunds Grūzītis and Samia Touileb, for leading the workshop selection. Thank you to Mike Zhang, our social media chair, for all your posts and for spreading information about the conference. My ultimate thank you goes to the local organizer team, Helen Kaljumäe, Merily Remma, and Kadri Vare, for a truly amazing job; the conference wouldn't have happened without your effort! It was a pleasure to work together.

On behalf of the organizing committee, we would like to thank the NoDaLiDa/Baltic-HLT sponsors for their generous financial support that helped us organize an affordable conference. We would also like to thank all the conference speakers and participants. Your interactions and enthusiasm are what will make the actual conference into a forum for fruitful conversations and discussions that contribute to connections for years to come.

Welcome, and I hope you enjoy the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies

Sara Stymne
Uppsala
March 2025

Message from the Local Organizers

When we began planning this event, we initially set out to host Baltic-HLT, a key conference for the Baltic language technology community. However, it ended up that this year marks the first time that Baltic-HLT and NoDaLiDa have been brought together, uniting two established traditions into one joint conference. In an era shaped by rapid advancements in language technologies – especially the rise of large language models – collaboration across regions and disciplines is more important than ever. Together we can address common challenges and ensure that every language, large or small, has a place in the digital future. We are pleased to welcome you to Tallinn, where cultural richness and innovation come together! We hope your time here brings new ideas, valuable connections, and leaves you with great memories.

Thank you for everyone for being part of NoDaLiDa/Baltic-HLT 2025, and we wish you a wonderful conference!

Organizing Committee

General Chair

Sara Stymne, Uppsala University, Sweden

Program Chair Advisors

Mark Fišel, University of Tartu, Estonia

Inguna Skadiņa, University of Latvia/Tilde, Latvia

Program Chairs

Daniel Hershcovich, University of Copenhagen, Denmark

Jenna Kanerva, University of Turku, Finland

Pierre Lison, Norwegian Computing Centre, Norway

Andrius Utkas, Vytautas Magnus University, Lithuania

Workshop Chairs

Normunds Grūzītis, University of Latvia, Latvia

Samia Touileb, University of Bergen, Norway

Publication Chair

Richard Johansson, Chalmers University of Technology and University of Gothenburg, Sweden

Social Media Chair

Mike Zhang, Aalborg University, Denmark

Local Organizers

Helen Kaljumäe, Institute of the Estonian Language, Estonia

Merily Remma, Institute of the Estonian Language, Estonia

Kadri Vare, Institute of the Estonian Language, Estonia

Reviewers

Eleri Aedmaa, Manex Agirrezabal, Lars Ahrenberg, David Alfter, Ali Al-Laith, Sven Aller, Kais Allkivi, Tanel Alumäe, Mark Anderson, Mattias Appelgren, Ilze Auzina, Eduard Barbu, Guntis Barzdins, Ali Basirat, Elisa Bassignana, Timo Baumann, Aleksandrs Berdicevskis, Johanna Björklund, Jari Björne, Gerlof Bouma, Johan Boye, Stephanie Brandl, Chloé Braud, Micaella Bruton, Maja Buljan, Laura Cabello, Lucas Georges Gabriel Charpentier, Yiyi Chen, Mathias Creutz, David Dale, Vera Danilova, Iben Nyholm Debess, Hannah Devinney, Ruchira Dhar, Stefanie Dipper, Simon Dobnik, Lucia Donatelli, Aleksei Dorkin, Luise Dürlich, Jens Edlund, Antske Fokkens, Filip Ginter, Evangelia Gogoulou, Rob van der Goot, Tamás Grósz, Hugo Lewi Hammer, Stefan Heinrich, Aron Henriksson, Erik Henriksson, Oskar Holmström, David House, Christine Howes, Guimin Hu, Maciej Janicki, Moa Johansson, Arne Jönsson, Heiki-Jaan Kaalep, Emil Kalbaliyev, Danguolė Kalinauskaitė, Kaarel Kaljurand, Amanda Kann, Antti Kanner, Jurgita Kapočiūtė-Dzikienė, Jussi Karlgren, Andreas Kirkedal, Mare Koit, Tomas Krilavičius, Marco Kuhlmann, Jenny Kunz, Murathan Kurfalı, Mikko Kurimo, Robin Kurtz, Andrey Kutuzov, Hele-Andra Kuulmets, Harm Lameris, Ekaterina Lapshinova-Koltunski, Heather Lent, Krister Lindén, Ellinor Lindqvist, Sharid Loáiciga, Agnes Luhtaru, Petter Mæhlum, Giacomo Magnifico, Arianna Masciolini, Timothee Mickus, Vladislav Mikhailov, Kirill Milintsevich, Hans Moen, Anssi Moisio, Kadri Muischnek, Kaili Müürisep, Arild Brandrud Næss, Costanza Navarretta, Anna Björk Nikulasdóttir, Joakim Nivre, Bill Noble, Emily Öhman, Siim Orasmaa, Jim O'Regan, Petya Osenova, Robert Östling, Siddhesh Milind Pawar, Bolette S. Pedersen, Qiwei Peng, Eva Pettersson, Mārcis Pinnis, Flammie A Pirinen, Taïdo Purason, Alessandro Raganato, Liisa Rätsep, Sebastian Reimann, Matīss Rikters, Egil Rønningstad, Ahmed Ruby, Askars Salimbajevs, David Samuel, Ricardo Muñoz Sánchez, David Sasu, Baiba Valkovska Saulīte, Denitsa Saynova, Barbara Scalvini, Yves Scherrer, Djamé Seddah, Kiril Ivanov Simov, Kairit Sirts, Maria Skeppstedt, Somnath, Steinþór Steingrímsson, Felix Stollenwerk, Antti Suni, Torbjørn Svendsen, Monorama Swain, Nina Tahmasebi, Gongbo Tang, Timothy R Tangherlini, Arvi Tavast, Jörg Tiedemann, Crina Tudor, Dennis Ulmer, Teemu Vahtola, Jurgita Vaičenonienė, Thomas Vakili, Daniel Varab, Erik Velldal, Martin Volk, Elena Volodina, Fredrik Wahlberg, Nicholas Thomas Walker, Sondre Wold, Lisa Yankovskaya, Huiling You, Oreen Yousuf, Niklas Zechner, Ingo Ziegler, Heike Zinsmeister

Invited Talk: Not all Language Models Need to be Large – Studying Language Evolution and Acquisition with Modern Neural Networks

Arianna Bisazza
University of Groningen

Why do languages look the way they do? And what makes us so good at learning language as we grow up? Since the early days of connectionism, outstanding questions about human language have been investigated by means of simulations involving small neural networks (NNs) and toy languages. Is this still possible and meaningful in the age of Large pre-trained Language Models (LLMs)?

In this talk, I'll propose that modern NNs can indeed be a valuable tool to simulate and study processes of language evolution and acquisition. This, however, requires having control of training data, model architecture, and learning setup, which is typically not possible with LLMs.

I will then present two lines of research following these principles, namely: (1) simulating language change using small NN-agents that learn to communicate with pre-defined artificial languages, and (2) simulating the acquisition of syntax by training LMs on child-directed language. I'll end with a discussion of the value of interdisciplinarity and the importance of experimenting in controlled setups, rather than focusing all our research efforts on the evaluation of LLMs.

Invited Talk: The Illusion of Understanding – Unpacking the True Capabilities of Language Models

Dirk Hovy
Bocconi University

The rapid development of large language models in recent years has transformed the field of NLP. Many people are concerned that it has trivialized the field or even rendered it obsolete. In this talk, I'll argue that neither is true: NLP has a long way to go, and LLMs are the most recent in a long line of methods that have advanced the field. LLMs have freed us from many of the nitty-gritty details that previously hampered NLP research, allowing us to focus on larger and more interesting questions.

One of the most fundamental questions is what it means to “understand” language. In a world where AI can generate anything from translations to poetry and code, it's easy to believe these models genuinely understand us. However, despite its linguistic abilities, today's generative AI still resembles a skilled mimic rather than a genuine linguist. We will look at thought experiments and real-world examples to demonstrate the limitations of statistical models' knowledge, their inability to grasp context and nuance, and the dangers of overestimating their abilities. I will emphasize the theoretical and practical implications for future language technology, with a focus on social context. Drawing on philosophy, linguistics, and NLP history, we will investigate what it truly means to ‘understand’ a language beyond the words and the implications for safety and utility in LLMs.

Invited Talk: No Sex, no Future – On the Status of Estonian in a Changing World

Arvi Tavast

Institute of the Estonian Language

Apart from well-known anecdotes about the absence of gender marking and future tense, the most peculiar feature of Estonian is its number of speakers. Being one of the smallest fully functional languages in the world, it is a source of pride for its speakers, as well as a central part of their identity. The resulting puristic attitudes towards language also enjoy strong legal support. One of the enablers of this ideological stance is the channel metaphor of communication: that language as a system exists independently of its speakers, and communication works in virtue of using a shared code to encode and decode messages. This metaphor is still going strong in folk linguistics despite all evidence to the contrary, including recent advances in language modelling. A completely different reading for the title of the talk is provided by more recent learning- and prediction-based accounts of why we understand each other. Language, like any naturally evolving system, is vitally dependent on the random variability that is so conveniently present in linguistic data. This makes openness to new information a precondition to having a future, also for languages.

Table of Contents

<i>Annotating and Classifying Direct Speech in Historical Danish and Norwegian Literary Texts</i> Ali Al-Laith, Alexander Conroy, Kirstine Nielsen Degn, Jens Bjerring-Hansen and Daniel Hershcovich	1
<i>Diachronic Analysis of Phrasal Verbs in English Scientific Writing</i> Diego Alves	8
<i>Applying and Optimising a Multi-Scale Probit Model for Cross-Source Text Complexity Classification and Ranking in Swedish</i> Elsa Andersson, Johan Falkenjack and Arne Jönsson	17
<i>Playing by the Rules: A Benchmark Set for Standardized Icelandic Orthography</i> Bjarki Ármannsson, Hinrik Hafsteinsson, Jóhannes B. Sigtryggsson, Atli Jasonarson, Einar Freyr Sigurðsson and Steinþór Steingrímsson	28
<i>An Icelandic Linguistic Benchmark for Large Language Models</i> Bjarki Ármannsson, Finnur Ágúst Ingimundarson and Einar Freyr Sigurðsson	37
<i>Transfer-Learning German Metaphors Inspired by Second Language Acquisition</i> Maria Berger	48
<i>Comparative Concepts or Descriptive Categories: a UD Case study</i> Mathieu Pierre Boyer and Mathieu Dehouck	55
<i>Investigating the effectiveness of Data Augmentation and Contrastive Learning for Named Entity Recognition</i> Noel Chia, Ines Rehbein and Simone Paolo Ponzetto	66
<i>Comparing Human and Machine Translations of Generative Language Model Evaluation Datasets</i> Sander Bijl de Vroe, George Stampoulidis, Kai Hakala, Aku Rouhe, Mark van Heeswijk and Jussi Karlgren	80
<i>GliLem: Leveraging GliNER for Contextualized Lemmatization in Estonian</i> Aleksi Dorkin and Kairit Sirts	86
<i>Comparative analysis of optical character recognition methods for Sámi texts from the National Library of Norway</i> Tita Enstad, Trond Trosterud, Marie Iversdatter Røsok, Yngvil Beyer and Marie Roald	98
<i>LAG-MMLU: Benchmarking Frontier LLM Understanding in Latvian and Giriama</i> Naome A. Etori, Arturs Kanepajis, Kevin Lu and Randu Karisa	109
<i>Better Benchmarking LLMs for Zero-Shot Dependency Parsing</i> Ana Ezquerro, Carlos Gómez-Rodríguez and David Vilares	121
<i>Optimizing Estonian TV Subtitles with Semi-supervised Learning and LLMs</i> Artem Fedorchenko and Tanel Alumäe	136
<i>Modeling Multilayered Complexity in Literary Texts</i> Pascale Feldkamp, Márton Kardos, Kristoffer Nielbo and Yuri Bizzoni	142
<i>Does Preprocessing Matter? An Analysis of Acoustic Feature Importance in Deep Learning for Dialect Classification</i> Lea Fischbach, Caroline Kleen, Lucie Flek and Alfred Lameli	159

<i>Language of the Swedish Manosphere with Swedish FrameNet</i>	
Emilie Marie Carreau Francis	170
<i>Hotter and Colder: A New Approach to Annotating Sentiment, Emotions, and Bias in Icelandic Blog Comments</i>	
Steinunn Rut Friðriksdóttir, Dan Saatrup Nielsen and Hafsteinn Einarsson	181
<i>Towards large-scale speech foundation models for a low-resource minority language</i>	
Yaroslav Getman, Tamás Grósz, Katri Hiovain-Asikainen, Tommi Lehtonen and Mikko Kurimo	192
<i>OpusDistillery: A Configurable End-to-End Pipeline for Systematic Multilingual Distillation of Open NMT Models</i>	
Ona de Gibert, Tommi Nieminen, Yves Scherrer and Jörg Tiedemann	201
<i>Mind the Gap: Diverse NMT Models for Resource-Constrained Environments</i>	
Ona de Gibert, Dayyán O'Brien, Dušan Variš and Jörg Tiedemann	209
<i>Testing relevant linguistic features in automatic CEFR skill level classification for Icelandic</i>	
Isidora Glišić, Caitlin Laura Richter and Anton Karl Ingason	217
<i>MorSeD: Morphological Segmentation of Danish and its Effect on Language Modeling</i>	
Rob van der Goot, Anette Jensen, Emil Allerslev Schledermann, Mikkel Wildner Kildeberg, Nicolaj Larsen, Mike Zhang and Elisa Bassignana	223
<i>Opinion Units: Concise and Contextualized Representations for Aspect-Based Sentiment Analysis</i>	
Emil Häglund and Johanna Björklund	230
<i>Aligning Language Models for Icelandic Legal Text Summarization</i>	
Pórir Hrafn Harðarson, Hrafn Loftsson and Stefán Ólafsson	241
<i>Question-parsing with Abstract Meaning Representation enhanced by adding small datasets</i>	
Johannes Heinecke, Maria Boritchev and Frédéric Herledan	252
<i>FinerWeb-10BT: Refining Web Data with LLM-Based Line-Level Filtering</i>	
Erik Henriksson, Otto Tarkka and Filip Ginter	258
<i>Margins in Contrastive Learning: Evaluating Multi-task Retrieval for Sentence Embeddings</i>	
Tollef Emil Jørgensen and Jens Breitung	269
<i>Database of Latvian Morphemes and Derivational Models: ideas and expected results</i>	
Andra Kalnača, Tatjana Pakalne and Kristīne Levāne-Petrova	279
<i>Localizing AI: Evaluating Open-Weight Language Models for Languages of Baltic States</i>	
Jurgita Kapočiūtė-Dzikienė, Toms Bergmanis and Mārcis Pinnis	287
<i>How Aunt-Like Are You? Exploring Gender Bias in the Genderless Estonian Language: A Case Study</i>	
Elisabeth Kaukonen, Ahmed Sabir and Rajesh Sharma	296
<i>Estonian isolated-word text-to-speech synthesiser</i>	
Indrek Kiissel, Liisi Piits, Heete Sahkai, Indrek Hein, Liis Ermus and Meelis Mihkla	302
<i>BiaSWE: An Expert Annotated Dataset for Misogyny Detection in Swedish</i>	
Kättriin Kukk, Danila Petrelli, Judit Casademont, Eric J. W. Orlowski, Michal Dzielinski and Maria Jacobson	307
<i>Predictability of Microsyntactic Units across Slavic Languages: A translation-based Study</i>	
Maria Kunilovskaya, Iuliia Zaitova, Wei Xue, Irina Stenger and Tania Avgustinova	313

<i>Train More Parameters But Mind Their Placement: Insights into Language Adaptation with PEFT</i> Jenny Kunz	323
<i>SweSAT-1.0: The Swedish University Entrance Exam as a Benchmark for Large Language Models</i> Murathan Kurfalı, Shorouq Zahra, Evangelia Gogoulou, Luise Dürlich, Fredrik Carlsson and Joakim Nivre	331
<i>How Well do LLMs know Finno-Ugric Languages? A Systematic Assessment</i> Hele-Andra Kuulmets, Taido Purason and Mark Fishel	340
<i>Mapping Faroese in the Multilingual Representation Space: Insights for ASR Model Optimization</i> Dávid í Lág, Barbara Scalvini and Jon Gudnason	354
<i>Towards a Derivational Semantics Resource for Latvian</i> Ilze Lokmane, Mikus Grasmanis, Agute Klints, Gunta Nešpore-Bērzkalne, Pēteris Paikens, Lauma Pretkalniņa, Laura Rituma, Madara Stāde and Evelīna Tauriņa	359
<i>Poro 34B and the Blessing of Multilinguality</i> Risto Luukkonen, Jonathan Burdge, Elaine Zosa, Aarne Talman, Ville Komulainen, Väinö Hatan- pää, Peter Sarlin and Sampo Pyysalo	367
<i>Can summarization approximate simplification? A gold standard comparison</i> Giacomo Magnifico and Eduard Barbu	383
<i>A Comparative Study of PEFT Methods for Python Code Generation</i> Johanna Männistö, Joseph Attieh and Jörg Tiedemann	390
<i>A Collection of Question Answering Datasets for Norwegian</i> Vladislav Mikhailov, Petter Mæhlum, Victoria Ovedie Chruickshank Langø, Erik Velldal and Lilja Øvreid	397
<i>Incorporating Target Fuzzy Matches into Neural Fuzzy Repair</i> Tommi Nieminen, Jörg Tiedemann and Sami Virpioja	408
<i>Constructions and Strategies in Universal Dependencies</i> Joakim Nivre	419
<i>Finnish SQuAD: A Simple Approach to Machine Translation of Span Annotations</i> Emil Nuutinen, Iiro Rastas and Filip Ginter	424
<i>How to Tune a Multilingual Encoder Model for Germanic Languages: A Study of PEFT, Full Fine- Tuning, and Language Adapters</i> Romina Oji and Jenny Kunz	433
<i>Match ‘em: Multi-Tiered Alignment for Error Analysis in ASR</i> Phoebe Parsons, Knut Kvale, Torbjørn Svendsen and Giampiero Salvi	440
<i>Adding Metadata to Existing Parliamentary Speech Corpus</i> Phoebe Parsons, Per Erik Solberg, Knut Kvale, Torbjørn Svendsen and Giampiero Salvi	448
<i>Paragraph-Level Machine Translation for Low-Resource Finno-Ugric Languages</i> Dmytro Pashchenko, Lisa Yankovskaya and Mark Fishel	458
<i>Evaluating LLM-Generated Explanations of Metaphors – A Culture-Sensitive Study of Danish</i> Bolette S. Pedersen, Nathalie Sørensen, Sanni Nimb, Dorte Haltrup Hansen, Sussi Olsen and Ali Al-Laith	470

<i>Tokenization on Trial: The Case of Kalaallisut–Danish Legal Machine Translation</i>	
Esther Ploeger, Paola Saucedo, Johannes Bjerva, Ross Deans Kristensen-McLachlan and Heather Lent	480
<i>The Roles of English in Evaluating Multilingual Language Models</i>	
Wessel Poelman and Miryam de Lhoneux	492
<i>Revisiting Projection-based Data Transfer for Cross-Lingual Named Entity Recognition in Low-Resource Languages</i>	
Andrei Politov, Oleh Shkalikov, Rene Jäkel and Michael Färber	499
<i>Empathy vs Neutrality: Designing and Evaluating a Natural Chatbot for the Healthcare Domain</i>	
Cristina Reguera-Gómez, Denis Paperno and Maaïke H. T. de Boer	508
<i>Assessed and Annotated Vowel Lengths in Spoken Icelandic Sentences for L1 and L2 Speakers: A Resource for Pronunciation Training</i>	
Caitlin Laura Richter, Kolbrún Friðriksdóttir, Kormákur Logi Bergsson, Erik Anders Maher, Ragnheiður María Benediktsdóttir and Jon Guðnason	518
<i>The BRAGE Benchmark: Evaluating Zero-shot Learning Capabilities of Large Language Models for Norwegian Customer Service Dialogues</i>	
Mike Riess and Tollef Emil Jørgensen	525
<i>Mixed Feelings: Cross-Domain Sentiment Classification of Patient Feedback</i>	
Egil Rønningstad, Lilja Charlotte Storset, Petter Mæhlum, Lilja Øvrelid and Erik Velldal	537
<i>The Impact of Copyrighted Material on Large Language Models: A Norwegian Perspective</i>	
Javier de la Rosa, Vladislav Mikhailov, Lemei Zhang, Freddy Wetjen, David Samuel, Peng Liu, Rolv-Arild Braaten, Petter Mæhlum, Magnus Breder Birkenes, Andrey Kutuzov, Tita Enstad, Hans Christian Farsethås, Svein Arne Brygfeld, Jon Atle Gulla, Stephan Oepen, Erik Velldal, Wilfred Østgulen, Lilja Øvrelid and Aslak Sira Myhre	544
<i>Encoder vs Decoder: Comparative Analysis of Encoder and Decoder Language Models on Multilingual NLU Tasks</i>	
Dan Saattrup Nielsen, Kenneth Enevoldsen and Peter Schneider-Kamp	561
<i>Small Languages, Big Models: A Study of Continual Training on Languages of Norway</i>	
David Samuel, Vladislav Mikhailov, Erik Velldal, Lilja Øvrelid, Lucas Georges Gabriel Charpentier, Andrey Kutuzov and Stephan Oepen	573
<i>Rethinking Low-Resource MT: The Surprising Effectiveness of Fine-Tuned Multilingual Models in the LLM Age</i>	
Barbara Scalvini, Iben Nyholm Debess, Annika Simonsen and Hafsteinn Einarsson	609
<i>Prompt Engineering Enhances Faroese MT, but Only Humans Can Tell</i>	
Barbara Scalvini, Annika Simonsen, Iben Nyholm Debess and Hafsteinn Einarsson	622
<i>Interactive maps for corpus-based dialectology</i>	
Yves Scherrer and Olli Kuparinen	634
<i>Profiling Bias in LLMs: Stereotype Dimensions in Contextual Word Embeddings</i>	
Carolyn M. Schuster, Maria-Alexandra Roman, Shashwat Ghatwala and Georg Groh	639
<i>Entailment Progressions: A Robust Approach to Evaluating Reasoning Within Larger Discourse</i>	
Rishabh Shastri, Patricia Chiril, Joshua Charney and David Uminsky	651

<i>Generative AI for Technical Writing: Comparing Human and LLM Assessments of Generated Content</i> Karen de Souza, Alexandre Nikolaev and Maarit Koponen	661
<i>MC-19: A Corpus of 19th Century Icelandic Texts</i> Steinþór Steingrímsson, Einar Freyr Sigurðsson and Atli Jasonarson	680
<i>Surface-Level Morphological Segmentation of Low-resource Inuktitut Using Pre-trained Large Language Models</i> Mathias Stenlund, Hemanadhan Myneni and Morris Riedel	688
<i>The Devil's in the Details: the Detailedness of Classes Influences Personal Information Detection and Labeling</i> Maria Irena Szawerna, Simon Dobnik, Ricardo Muñoz Sánchez and Elena Volodina	697
<i>Braxen 1.0</i> Christina Tännander and Jens Edlund	709
<i>Temporal Relation Classification: An XAI Perspective</i> Sofia Elena Terenziani	714
<i>Benchmarking Abstractive Summarisation: A Dataset of Human-authored Summaries of Norwegian News Articles</i> Samia Touileb, Vladislav Mikhailov, Marie Ingeborg Kroka, Lilja Øvrelid and Erik Velldal ..	729
<i>Efficient Elicitation of Fictitious Nursing Notes from Volunteer Healthcare Professionals</i> Jesper Vaaben Bornerup and Christian Hardmeier	739
<i>Analyzing the Effect of Linguistic Instructions on Paraphrase Generation</i> Teemu Vahtola, Songbo Hu, Mathias Creutz, Ivan Vulić, Anna Korhonen and Jörg Tiedemann	755
<i>SweClinEval: A Benchmark for Swedish Clinical Natural Language Processing</i> Thomas Vakili, Martin Hansson and Aron Henriksson	767
<i>Dialectal treebanks and their relation with the standard variety: The case of East Cretan and Standard Modern Greek.</i> Socrates Vakirtzian, Vivian Stamou, Yannis Kazos and Stella Markantonatou	776
<i>Danoliteracy of Generative Large Language Models</i> Søren Vejlgård Holm, Lars Kai Hansen and Martin Carsten Nielsen	785
<i>NorEventGen: generative event extraction from Norwegian news</i> Huiling You, Samia Touileb, Erik Velldal and Lilja Øvrelid	801
<i>SnakModel: Lessons Learned from Training an Open Danish Large Language Model</i> Mike Zhang, Max Müller-Eberstein, Elisa Bassignana and Rob van der Goot	812
<i>Got Compute, but No Data: Lessons From Post-training a Finnish LLM</i> Elaine Zosa, Ville Komulainen and Sampo Pyysalo	826

Annotating and Classifying Direct Speech in Historical Danish and Norwegian Literary Texts

Ali Al-Laith^{1,2}, Alexander Conroy¹, Kirstine Nielsen Degn¹,
Jens Bjerring-Hansen¹ and Daniel Hershcovich²

Department of Nordic Studies and Linguistics, University of Copenhagen¹

Department of Computer Science, University of Copenhagen²

alal@di.ku.dk, knd@hum.ku.dk, alc@hum.ku.dk,

bspedersen@hum.ku.dk, jbh@hum.ku.dk, dh@di.ku.dk

Abstract

Analyzing direct speech in historical literary texts provides insights into character dynamics, narrative style, and discourse patterns. In late 19th century Danish and Norwegian fiction direct speech reflects characters’ social and geographical backgrounds. However, inconsistent typographic conventions in Scandinavian literature complicate computational methods for distinguishing direct speech from other narrative elements. To address this, we introduce an annotated dataset from the MeMo corpus, capturing speech markers and tags in Danish and Norwegian novels. We evaluate pre-trained language models for classifying direct speech, with results showing that a Danish Foundation Model (DFM), trained on extensive Danish data, has the highest performance. Finally, we conduct a classifier-assisted quantitative corpus analysis and find a downward trend in the prevalence of speech over time.

1 Introduction

The analysis of direct speech in literary texts provides valuable insights into narrative style, character dynamics as well as aesthetic developments and other broader discourse patterns. In the context of literary history, it has been argued that direct speech, understood as a narrative element that purports to quote a character’s speech (Cohen and Green, 2019), is one of the most distinctive components of modern Danish fiction from the late 19th century (Kristensen, 1955). Realist authors of the period use direct speech to reflect characters’ social and geographical backgrounds through dialogue rather than explicit description, aiming to portray the fictional world with verisimilitude, i.e. a touch of the real. In Scandinavian litera-

ture, typographic marking of speech—such as quotation marks, dashes, and colons—is often inconsistent, complicating the task of distinguishing direct speech from narrative text. This is especially true for Danish and Norwegian novels from the late 19th century, where typographic conventions are highly variable. While readers can often intuitively recognize direct speech, computational approaches require structured annotation to accurately capture these nuanced typographic and linguistic features (Stymne, 2024).

We introduce a newly annotated dataset derived from the MeMo corpus (Bjerring-Hansen et al., 2022), which includes annotations of speech markers, speech tags and speech separated from other narrative elements across Danish and Norwegian novels from the late 19th century. This dataset, annotated on the word level, facilitates the segmentation of direct speech from other narrative elements, enabling sequence tagging model training for the automated detection of these elements. We evaluate several pre-trained language models tailored for Danish and Norwegian, including the Danish Foundation Models (DFM; Enevoldsen et al., 2023) and MeMo-BERT (Al-Laith et al., 2024a), to assess their ability to detect direct speech in historical Scandinavian texts, and find DFM particularly effective. Our findings are of importance to not only literary scholars, but also (socio)linguists who are allowed an indirect access to spoken language from before modern recording technologies (Culpeper and Kytö, 2010).

Our contributions are threefold: (1) we present an annotated dataset that captures the typographic and linguistic indicators of direct speech in 19th century Danish and Norwegian literature, (2) we conduct an empirical evaluation of state-of-the-art language models fine-tuned on this dataset, and (3) we provide insights into the performance and generalization capabilities of these models for classifying direct speech.

Author	Novel	Type	Example
Kamillo Karstens	Grevinde Danner	German quotation marks	„Læs , “udbrød han
Michael Rosing	En Romantiker	Guillemet-form, Danish	»Kom Jomfru! lad os faa en Dans til Afsked «
Ragnhild Goldschmidt	En Kvindehistorie	Guillemet-form, French	«Laura, Din Kjole er vaad; regner det? »
Herman Bang	Tine	Dash	— Farvel!
Holger Drachmann	Forskrevet	Unmarked	Jeg husker Dem meget godt ! svarede han

Table 1: Excerpts from five novels from the MeMo corpus with different quotation styles.

2 Related Work

Direct speech identification. Identifying direct speech in literary texts has been a focal area in NLP, with various resources and methodologies addressing typographic and linguistic challenges across languages. The Swedish Literary corpus of Narrative and Dialogue (SLäNDa) exemplifies these efforts, providing annotated excerpts from Swedish novels between 1809 and 1940 that capture speech segments, tags, and speaker identification (Stymne and Östman, 2020, 2022). In a similar vein, Troiano and Vossen (2024) introduced CLAUSE-ATLAS, a corpus designed to study narrative structure in 19th and 20th century English novels, leveraging large language models for clause-based annotation.

Recent studies have also explored annotation challenges in texts lacking quotation marks. Stymne (2024) compared manual (gold) and automated (silver) annotation methods, finding that gold data yields better model performance at the token level, while silver data often excels at capturing speech spans. Despite these advancements, most methods rely on monolingual, genre-specific corpora that may not extend well to historical Scandinavian languages.

Historical literary Scandinavian NLP. Our study builds on recent advances in computational approaches for analyzing historical Scandinavian literature, emphasizing the need for tailored datasets and models suited to under-resourced languages. Allaith et al. (2023); Al-Laith et al. (2024b) developed NLP methods specifically adapted to the unique linguistic characteristics of 19th century Danish and Norwegian texts, addressing challenges such as archaic vocabulary, inconsistent orthography, and noisy data. Further studies, such as Feldkamp et al. (2024) and Lindhardt Overgaard et al. (2024), underscore that models and datasets customized for genre-specific nuances enhance the analysis of specialized text types. Bjerring-Hansen et al. (2024) also con-

tributed by distinguishing between contemporary and historical novels, underscoring the value of domain-specific resources for genre classification in historical corpora. These efforts highlight the importance of customized NLP frameworks for advancing computational humanities, particularly in historical Scandinavian literature.

3 Dataset

3.1 Main Corpus

We use the MeMo corpus (Bjerring-Hansen et al., 2022), comprising 859 Danish and Norwegian novels spanning the last 30 years of the 19th century, with more than 64 million tokens. We refer to this corpus as the ‘main corpus’. It should be noted that, until 1907, written Norwegian was practically identical to written Danish (Vikør, 2022).

3.2 Speech Corpus

Segment extraction. We randomly extract 100 segments, each consisting of three consecutive paragraphs, from 100 different novels in the MeMo corpus. For the selection of the target novels, five novels are handpicked by literary experts specifically to represent diverse quotation styles (see Table 1), while the remaining 95 are selected at random, ensuring diverse and comprehensive coverage of quotation styles.

Annotation guidelines. To address the challenges described in §1, we develop clear annotation criteria to ensure consistency and accuracy in identifying speech-related elements:

1. **Speech (“SP”):** All words and punctuation that are part of direct speech are labeled as “SP”. We do not differentiate embedded speech (e.g., quotations within speech) as both the outer and inner quotations are labeled as “SP”.
2. **Speech Marker (“SM”):** Any typographical markers indicating speech, such as quotation marks, colons, or dashes, are labeled

as “SM”. If a colon appears directly before quotation marks, it is also labelled “SM”. For example, in the following:

He shook his head and said: “Certainly, but the stones must be examined first”,
both the colon and quotation marks are labeled as “SM”.

3. **Speech Tag (“ST”)**: Speech tags (or inquit phrases), such as “he said,” “she asked,” or “they replied,” are labeled as “ST”. This label applies only to the verb and subject, excluding any adverbs or adverbial phrases, e.g., in *And then he whispered almost inaudibly* only “he whispered” is labeled as “ST”. Punctuation immediately preceding or following the tag is also considered part of the “ST” if it is not eligible to be marked as “SM”.
4. **Other (“O”)**: All other words and punctuation not categorized under the above labels are marked as “O”. This includes indirect speech and free indirect discourse. Additionally, inner thoughts and citations from letters or documents are also labelled as “O”.

Annotation process. The annotation is carried out on the INCEpTION platform (Klie et al., 2018) by three literary scholars with domain expertise in late 19th century Scandinavian fiction. For agreement calculation and in order to obtain a high-quality testing set, we select 15% of samples for multiple annotation by all three experts. These consist of 15 segments from 15 different novels from the last four years of the period, 1896–1899. In total, they contain 2,530 words. After separate annotation by the three experts, these are consolidated by word-level label majority vote for the final testing set. The rest of the segments in the dataset (75 segments from 75 different novels) are equally split among annotators to be annotated individually.

Annotation results. The annotation results demonstrate a clear prevalence of non-speech elements in the dataset, with a majority of words categorized as “Other”. Despite the lower representation of speech-related annotations, the presence of direct speech is still significant, indicating that dialogue plays an important role in the corpus. The minimal occurrences of “Speech Marker” and “Speech Tag” highlight

Class	#Words	%
Speech (“SP”)	7,655	32.6%
Speech Marker (“SM”)	579	2.5%
Speech Tag (“ST”)	363	1.5%
Other (“O”)	14,861	63.4%
Total	23,458	100%

Table 2: Distribution of annotated dataset.

the challenges in identifying these features. This distribution underscores the complexity of the dataset, as a result of diversity in both literary styles and typographical conventions, and the necessity for careful annotation to capture the nuances of speech within the text. Table 2 shows statistics about the manually annotated dataset.

Agreement. We use pairwise Cohen’s Kappa to assess Inter-Annotator Agreement (IAA) on the subset annotated by all three experts prior to consolidation. The pairwise comparisons between annotators resulted in an average Cohen’s Kappa score of 0.92, indicating substantial agreement among annotators in classifying direct speech from other narrative elements.

4 Experiments and Results

We model direct speech identification as token classification, i.e. sequence tagging, with the tags described in §3. We fine-tune and evaluate pre-trained language models for token classification.

4.1 Pre-trained Language Models

We select models pre-trained on Danish and Norwegian text, based on their performance on Danish and Norwegian literary benchmark datasets (Al-Laith et al., 2024a) and ScandEval (Nielsen, 2023). We experiment with models that are not primarily trained on historical/literary Danish or Norwegian. These include DanskBERT and DFM (Large), the Danish Foundation Models sentence encoder, both trained on the Danish Gigaword Corpus; NB-BERT-base, trained on the extensive digital collection at the National Library of Norway; and MeMo-BERT-03, which was developed through continued pre-training of DanskBERT on the MeMo corpus. The following provides an explanation of each model used in this research.

DanskBERT. DanskBERT,¹ a top-performing Danish language model noted for its success on the ScandEval benchmark (Snæbjarnarson et al., 2023), is based on the XLM-RoBERTa architecture and trained on the Danish Gigaword Corpus (Strømberg-Derczynski et al., 2021). It features 24 layers, a hidden dimension of 1024, 16 attention heads, and a subword vocabulary of 250,000. The model was trained with a batch size of 2,000 for 500,000 steps on 16 V100 GPUs over two weeks.

Danish Foundation Models sentence encoder.

A sentence-transformers model (Enevoldsen et al., 2023) based on the BERT architecture, featuring 24 layers, 16 attention heads, and a hidden size of 1024. It incorporates a dropout rate of 0.1 for attention probabilities and hidden states, using GELU activation and supporting up to 512 position embeddings. With a vocabulary size of 50,000 tokens, this model, referred to as DFM (Large), excels in some NLP downstream tasks such as sentiment analysis and named entity recognition.²

MeMo-BERT-03. Developed by continuing the pre-training of the pre-trained Transformer language model DanskBERT (Al-Laith et al., 2024a).³ This foundation allows MeMo-BERT-03 to leverage extensive linguistic knowledge for NLP tasks in historical literary Danish including sentiment analysis and word sense disambiguation. The model outperformed different models in sentiment analysis and word sense disambiguation tasks (Al-Laith et al., 2024a).

NB-BERT-base. A general-purpose BERT-base model was developed using the extensive digital collection at the National Library of Norway (Kummervold et al., 2021).⁴ It follows the architecture of the BERT Cased multilingual model and has been trained on a diverse range of Norwegian texts, encompassing both Bokmål and Nynorsk from the past 200 years. This comprehensive training allows the NB-BERT-base to effectively handle a wide array of NLP tasks in Norwegian. The model achieved the second-highest perfor-

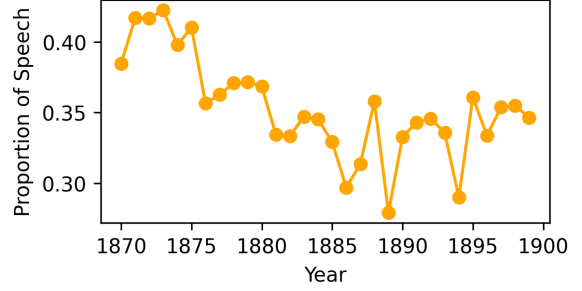


Figure 1: Proportion of speech tokens, predicted by fine-tuned DFM (Large), by publication year.

mance ranking in the Norwegian Named Entity Recognition task compared to other models listed on the ScandEval benchmark for Norwegian natural language understanding.

4.2 Experimental Setup

To fine-tune the models, we use a batch size of 32, and train for 20 epochs with the AdamW optimizer at a learning rate of 10^{-3} , choosing the best epoch based on validation loss. For evaluation, we employ word-level weighted average F1-score. We select for testing the 15% of the dataset annotated by all three experts, and randomly split the rest such that 70% of the overall annotated dataset is used for training and 15% for development.

4.3 Speech Classification Results

Fine-tuning results in notable performance variations, as shown in Table 3. DFM (Large) achieves the best results, indicating strong generalization. NB-BERT-base follows closely, but DanskBERT and MeMo-BERT-03 perform moderately, showing a notable drop from validation to test scores, suggesting less robust generalization. As described in §3.2, the testing set consists of segments from the last four years of the period, while (as described in §4.2) the validation set is randomly sampled from the rest of the period. The testing set therefore represents a time shift from training and is more challenging.

5 Classifier-assisted Corpus Analysis

We use the top-performing model, DFM (Large), to tag all unlabeled segments in the main corpus. This results in 35% of words labeled as speech, 61% as non-speech, 2% as speech markers and 2% as speech tags. Figure 1 shows the proportion of speech and non-speech labels over years,

¹<https://huggingface.co/vesteinn/DanskBERT>

²<https://huggingface.co/KennethEnevoldsen/dfm-sentence-encoder-large-exp2-no-lang-align>

³<https://huggingface.co/MiMe-MeMo/MeMo-BERT-03>

⁴<https://huggingface.co/NbAiLab/nb-bert-base>

Model	Validation	Testing		
	F1-score	F1-score	Precision	Recall
DanskBERT	0.82	0.71	0.71	0.72
DFM (Large)	0.94	0.89	0.89	0.90
MeMo-BERT-03	0.81	0.73	0.73	0.74
NB-BERT-base	0.93	0.87	0.87	0.87

Table 3: Fine-tuned models’ word-level results on validation and testing sets, of 15 segments each.

illustrating a decreasing trend in the proportion of direct speech over time, with the highest point at 42% in 1874, declining to a low of 29% by 1889.⁵ This downward trend stands in contrast to findings from other quantitative studies on direct speech in novels. For example, in the study of British 19th century novels by Menon (2019), the overall fraction of dialogue across her entire corpus compares roughly to ours (36%), but she finds no significant change over time.

Furthermore, our findings challenge the widely held critical assumption within literary historiography that the use of direct speech increased with the rise of the realist novel in the late 19th century, as argued by Kristensen (1955), Allison (2018), and Cohen and Green (2019). Instead, our analysis seems more consistent with the argument presented by Cohn (1978) that the French naturalist aesthetic favored free indirect speech over direct speech, leading to a decline of the latter. This perspective aligns more closely with the downward trend we observe in 19th century Scandinavian literature than with the stable levels of direct speech that Menon (2019) reports in British novels from the same period. In other words, based on these quantitative analyses of direct speech, late 19th century Scandinavian novels appear to align more closely with conventional ideas of naturalist narrative techniques than with those of more conventional realist aesthetics.

6 Conclusion

We presented a dedicated dataset and methodology for annotating direct speech in Danish and Norwegian novels from the late 19th century, useful for not only literary studies but also for lin-

guistics by providing access to representations of 19th spoken language. By building on the MeMo corpus, we systematically annotated typographic markers, speech tags, and direct speech segments, addressing the significant variation and inconsistencies in typographic conventions within historical Scandinavian literature. Through our experiments with multiple language models, including Danish Foundation Models and MeMo-BERT, we found that DFM (Large) performed best. Using it to quantify the proportion of speech in the main corpus, we observed a decreasing trend over time.

Future work will extend our analysis to include other variations of speech, namely indirect discourse, i.e. reporting of character speech, and free indirect discourse, namely the incorporation of a character’s speech within the narrator’s language (Cohen and Green, 2019). Literary-historical research will examine the lexical variations of the speech tags within the corpus to address a hypothesis (Allison, 2018) that a narrative development from “telling” to “showing” in 19th century literature is manifested in a movement towards greater nuance and lexical variation in the speech tags. While ‘telling’ is a narrative style, where events are explained explicitly (e.g., ‘He was angry’), ‘showing’ uses a more detailed narrative style to implicitly convey what is at stake in the event, as in ‘He slammed his fist on the table and shouted “Enough”’. The hypothesis is that this shift is reflected in more nuanced speech tags, moving from simple terms like ‘said’ to varied ones like ‘muttered’ or ‘snarled.’ Our code and data are in this Github repository: <https://github.com/mime-memo/DirectSpeech>.

References

Ali Al-Laith, Alexander Conroy, Jens Bjerring-Hansen, and Daniel Hershcovich. 2024a. Development and evaluation of pre-trained language models for historical Danish and Norwegian literary texts. In

⁵These numbers may not be perfectly accurate as they are a result of an accurate-but-not-perfect classifier, as shown in §4.3. Moreover, they may be more reliable for some years than for others, but we are unable to quantify this with our current dataset, since our testing set consists only of segments from 1896 to 1899.

- Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4811–4819, Torino, Italia. ELRA and ICCL.
- Ali Al-Laith, Daniel Hershcovich, Jens Bjerring-Hansen, Jakob Ingemann Parby, Alexander Conroy, and Timothy R Tangherlini. 2024b. Noise, novels, numbers. a framework for detecting and categorizing noise in Danish and Norwegian literature. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.
- Ali Allaith, Kirstine Degn, Alexander Conroy, Bolette Pedersen, Jens Bjerring-Hansen, and Daniel Hershcovich. 2023. Sentiment classification of historical Danish and Norwegian literary texts. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 324–334, Tórshavn, Faroe Islands. University of Tartu Library.
- Sarah Allison. 2018. *Reductive reading: A syntax of Victorian Moralizing*. Johns Hopkins University Press, Baltimore, Maryland.
- Jens Bjerring-Hansen, Ali Al-Laith, Daniel Hershcovich, Alexander Conroy, and Sebastian Ørtoft Rasmussen. 2024. Literary time travel: Distinguishing past and contemporary worlds in Danish and Norwegian fiction. In *Computational Humanities Research 2024*.
- Jens Bjerring-Hansen, Ross Deans Kristensen-McLachlan, Philip Diderichsen, and Dorte Haltrup Hansen. 2022. Mending fractured texts. a heuristic procedure for correcting OCR data.
- William A. Cohen and Laura Green. 2019. Introduction: Revisiting dialogue. *Narrative*, 27(2):1013–1019.
- Dorrit Cohn. 1978. *Transparent Minds: Narrative Modes for Presenting Consciousness in Fiction*. Princeton University Press, Princeton, United States.
- Jonathan Culpeper and Merja Kytö. 2010. *Early Modern English dialogues: Spoken interaction as writing*. Cambridge University Press, Cambridge, England.
- Kenneth Enevoldsen, Lasse Hansen, Dan S. Nielsen, Rasmus A. F. Egebæk, Søren V. Holm, Martin C. Nielsen, Martin Bernstorff, Rasmus Larsen, Peter B. Jørgensen, Malte Højmark-Bertelsen, Peter B. Vahlstrup, Per Møldrup-Dalum, and Kristoffer Nielbo. 2023. Danish foundation models.
- Pascale Feldkamp, Jan Kostkan, Ea Overgaard, Mia Jacobsen, and Yuri Bizzoni. 2024. Comparing tools for sentiment analysis of Danish literature from hymns to fairy tales: Low-resource language and domain challenges. In *Proceedings of the 14th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, pages 186–199, Bangkok, Thailand. Association for Computational Linguistics.
- Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. The INCEpTION platform: Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 5–9, Santa Fe, New Mexico. Association for Computational Linguistics.
- Sven Møller Kristensen. 1955. *Impressionismen i dansk prosa 1870-1900*. Gyldendal, Copenhagen, DK.
- Per E Kummervold, Javier De la Rosa, Freddy Wetjen, and Svein Arne Brygfeld. 2021. Operationalizing a national digital library: The case for a Norwegian transformer model. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 20–29, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- Ea Lindhardt Overgaard, Pascale Feldkamp, and Yuri Bizzoni. 2024. Towards a GoldenHymns dataset for studying diachronic trends in 19th century Danish religious hymns. In *Proceedings of the 5th Workshop on Computational Approaches to Historical Language Change*, pages 55–61, Bangkok, Thailand. Association for Computational Linguistics.
- Tara Menon. 2019. Keeping count: Direct speech in the nineteenth-century british novel. *Narrative*, 27(2):160–181.
- Dan Nielsen. 2023. ScandEval: A benchmark for Scandinavian natural language processing. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 185–201, Tórshavn, Faroe Islands. University of Tartu Library.
- Vésteinn Snæbjarnarson, Annika Simonsen, Goran Glavaš, and Ivan Vulić. 2023. Transfer to a low-resource language via close relatives: The case study on Faroese. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 728–737, Tórshavn, Faroe Islands. University of Tartu Library.
- Leon Strømberg-Derczynski, Manuel Ciosici, Rebekah Baglini, Morten H. Christiansen, Jacob Aarup Dalsgaard, Riccardo Fusaroli, Peter Juel Henriksen, Rasmus Hvingelby, Andreas Kirkedal, Alex Speed Kjeldsen, Claus Ladefoged, Finn Årup Nielsen, Jens Madsen, Malte Lau Petersen, Jonathan Hvithamar Ryrstrøm, and Daniel Varab. 2021. The Danish Gigaword corpus. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 413–421, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- Sara Stymne. 2024. Direct speech identification in Swedish literature and an exploration of training data type, typographical markers, and evaluation granularity. In *Proceedings of the 8th Joint SIGHUM Workshop on Computational Linguistics*

for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2024), pages 253–263, St. Julians, Malta. Association for Computational Linguistics.

Sara Stymne and Carin Östman. 2020. SLäNDa: An annotated corpus of narrative and dialogue in Swedish literary fiction. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 826–834, Marseille, France. European Language Resources Association.

Sara Stymne and Carin Östman. 2022. SLäNDa version 2.0: Improved and extended annotation of narrative and dialogue in Swedish literature. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5324–5333, Marseille, France. European Language Resources Association.

Enrica Troiano and Piek T.J.M. Vossen. 2024. CLAUSE-ATLAS: A corpus of narrative information to scale up computational literary analysis. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 3283–3296, Torino, Italia. ELRA and ICCL.

Lars S. Vikør. 2022. Rettskrivingsreform i store norske leksikon på snl.no. In <https://snl.no/rettskrivingsreform>.

Diachronic Analysis of Phrasal Verbs in English Scientific Writing

Diego Alves

Saarland University / Saarbrücken, Germany

diego.alves@uni-saarland.de

Abstract

Phrasal verbs (PVs) are a specific type of multi-word expression and a specific feature of the English language. However, their usage in scientific prose is limited. Our study focuses on the analysis of phrasal verbs in the scientific domain using information theory methods to describe diachronic phenomena such as conventionalization and diversification regarding the usage of PVs. Thus, we analysed their developmental trajectory over time from the mid-17th century to the end of the 20th century by measuring the relative entropy (Kullback-Leibler divergence), predictability in the context of the phrasal verbs particles (surprisal), and the paradigmatic variability using word embedding spaces. We were able to identify interesting phenomena such as the process of conventionalization over the 20th century and the peaks of diversification throughout the centuries.

1 Introduction

Multi-word Expressions (MWEs) are sequences composed of two or more words that have a degree of conventionality among speakers of the language community, holding a strong relationship in communicating meaning (Siyanova-Chanturia and Sittis, 2018). MWEs encompass idioms that are formally fixed and have a figurative meaning (e.g., *kick the bucket*), compounds (*bus ticket*), phrasal verbs (*take a ride*), and other formulaic expressions that are typically compositional and often lexically fairly productive (cf. Avgustinova and Iomdin (2019)).

MWEs contribute to language efficiency due to the highly predictable transitions from one word to the next and/or because of their high degree of

conventionalization (i.e., convergence in linguistic usage over time). Also, MWEs have a strong influence on register formation, providing conventional encodings of context-specific meanings.

We are principally interested in MWEs in scientific English from a diachronic perspective (mid-17th century to today). Scientific English developed into a recognizable register during the late modern period and became highly conventionalized in modern times (cf. Degaetano-Ortlieb and Teich (2022)).

However, phrasal verbs (PVs), despite being a specific type of multi-word expression and one of the most distinctive features of the English language, are less common in academic prose, when compared to other registers. In the scientific register, usually more specialized verbs are preferred (cf. Biber et al. (2021) and Brown et al. (2015)).

The usage of PVs in English scientific writing indicates specific lexical choices influenced by contextual configurations and communicative constraints. Thus, our aim is to investigate, using information theory measures, whether phrasal verbs contribute to standardization in scientific English as other types of MWEs and grammatical constructions do. Our idea is to analyse if the effects of conventionalization of phrasal verbs can be observed over time with three different approaches: 1) analysis of PVs temporal dynamics using relative entropy, 2) study of the predictability in the context of the PVs particles using surprisal measure, and 3) examination of the paradigmatic variability of PVs using embeddings.

The remainder of the paper is organized as follows. In Section 2 we discuss related work on PVs in scientific English. Sections 3 and 4 present our methods and results, followed by a discussion of the main findings in Section 5. We conclude with a summary and outlook (Section 6).

2 Related Work

As previously mentioned, PVs are known for being less common in scientific texts when compared to other registers. (Biber et al., 2000) shows that PVs are mostly used in speech and fiction. News texts tend to use less than these two genres, but academic prose is where PVs have the least overall frequency per million words.

Regarding diachronic analysis of PVs in scientific English, Alves et al. (2024) showed that compared to other types of MWEs, PVs are the only ones presenting a decrease in its relative frequency over time (mid-17th century to end of 20th century). Moreover, PVs present a specific behaviour regarding dispersion and association measures. In terms of dispersion, most PVs are not homogeneously distributed over time, only very specific ones commonly used in academic texts such as *carried out*, *pointed out*, and *depend on*. Regarding the association measures, as the verbs and particles are also found in other contexts, in most cases, the values were quite low, except for specific cases where the verb is mostly used with its particle (e.g., *churned up*, *smoothes out*, *budded off*). Although the authors present a preliminary diachronic analysis of the evolution of the association measure, no conventionalization study was presented.

The diachronic changes of the paradigmatic variability of different parts-of-speech in scientific English using word embedding space were analysed by Teich et al. (2021). Overall, there is a reduction of paradigmatic variability over time for the different grammatical classes. However, PVs were not analysed separately to see if their behaviour is similar or discrepant when compared to other verbs.

Moreover, there are numerous corpus-based studies of MWEs in different registers, including the scientific one (e.g. Biber and Barbieri (2007); Hyland (2008); Liu (2012)). Some of these descriptions include lists of MWEs used in academic texts that are freely available as part of English for Academic Purposes (EAP). However, since PVs are not commonly used in scientific texts, they are usually not considered in the analysis.

Regarding computational methods for identifying PVs, the PARSEME initiative¹ clearly identifies PVs or verb-particle constructions (VPCs) as one category of verbal MWEs. Multilingual cor-

pora annotated following PARSEME guidelines are available, however, without any diachronic data.

Finally, in terms of studies regarding the cognitive processing of PVs, most studies concern L2 learners and the difficulties of learning these specific MWEs (cf. Alejo-González (2010); Mohammed (2019); Alisoy (2023)). In their study, Perdomo and Kaan (2023) looked at surprisal measures to analyse the effects in priming of phrasal verb construction alternations, comparing native speakers and L2 learners, thus, focusing on learning difficulties, not in conventionalization processes as this paper.

3 Methods

3.1 Dataset

As our objective is to investigate the conventionalization processes of PVs in the development of English scientific writing, we used the Royal Society Corpus (RSC) 6.0, which is a diachronic corpus of scientific English covering the period from 1665 until 1996.

It comprises 47,837 texts (295,895,749 tokens), which are mainly scientific articles covering a wide range of areas from mathematical, physical, and biological sciences, and is based on the Philosophical Transactions and Proceedings of the Royal Society of London (Fischer et al., 2020).

The RSC 6.0 was parsed using Stanza tool (Qi et al., 2020) and the combined model for English, provided by the developers, which was trained with different Universal Dependencies² (UD) corpora. To extract the PVs from the RSC, we developed a Python script using pyconll library³ to identify and count the PVs⁴ in the RSC texts per year. A manual evaluation of 140 sentences (20 per 50-year period of the RSC) showed that the accuracy of the Stanza parser is 90 regarding PVs.

3.2 Information Theory Measures

To analyse the diachronic phenomena regarding PVs in Scientific English, we applied three different methods to measure the relative entropy (Kullback-Leibler divergence), the surprisal of the particles, and the paradigmatic variability. The

²<https://universaldependencies.org/>

³<https://github.com/pyconll/pyconll>

⁴Phrasal verbs are easily identified in texts parsed with UD corpora as the dependency label of the PV particle is *compound:prt* and its head is the verb.

¹<https://gitlab.com/parseme/corpora/-/wikis/home>

workflow is schematized in Figure 4 and further described in the following sub-sections.

3.2.1 Kullback-Leibler Divergence

To identify evolutionary trends in the use of phrasal verbs (PVs) within the RSC, we applied relative entropy, specifically the Kullback-Leibler Divergence (KLD; Kullback and Leibler (1951)). This method compares probability distributions by measuring the additional bits required to encode dataset A when using a (non-optimal) model based on dataset B for a given set of elements X, as described in Equation 1. In this study, A and B correspond to sub-sets of the RSC (e.g. time slices) and X, i.e. the ensemble of PVs.

$$D_{KL}(A\|B) = \sum_{x \in X} A(x) \log \left(\frac{A(x)}{B(x)} \right) \quad (1)$$

The KLD measure provides an indication of the degree of divergence between corpora and identifies the features that are primarily associated with a difference. Possible discrepancies regarding the vocabulary size of the subcorpora are controlled by using Jelinek-Mercer smoothing and lambda 0.05 (cf. Zhai and Lafferty (2004) and Fankhauser et al. (2014)).

To detect periods of change in the use of PVs using KLD, we adopt the methodology described in Degaetano-Ortlieb and Teich (2018)⁵. We compare 20-year windows of past and present language use sliding with a 5-year gap over the timeline (e.g. t1=1665-1685, t2=1671-1691). Then, by plotting the divergence for each comparison on the timeline, we can inspect peaks or troughs which indicate a change: a peak is an indication that the divergence of the analysed feature increases, and is thus *typical* of the future 20 years in comparison to the past 20 years.

Due to the asymmetric characteristic of the KLD, we are only interested in the direction from subsequent periods to the preceding ones as we aim to determine periodization from past to present in the development of PVs usage in English scientific writing.

In this study, we examined the KLD at two different levels: a) all PVs combined, to verify if a conventionalization process can be identified and b) each PV individually, to identify individual diachronic phenomena.

⁵Degaetano-Ortlieb and Teich (2018) make the code available at: <https://stefaniadegaetano.com/code/>

3.2.2 Surprisal

Surprisal is formalized as the negative log probability of a unit in context which results in bits of information (Shannon, 1948), as defined in Equation 2.

$$Surprisal(unit_i) = \log_2(unit_i|Context) \quad (2)$$

A decrease in the surprisal of a specific term can indicate a conventionalization phenomenon as showed by Degaetano-Ortlieb and Teich (2022) regarding scientific English using a four-gram language model. N-grams surprisal models have limitations, thus, Steuer et al. (2024) propose a transformer-based surprisal model trained with the RSC corpus.

Our analysis concerns the diachronic changes of the surprisal values of the PVs particles. Thus, using the transformer-based model cited above trained over the RSC divided into 10-year periods, we extracted, per year of the RSC, the surprisal values of the particles identified in the parsed version of the dataset.

3.2.3 Paradigmatic Variability

To analyse diachronic changes in the paradigmatic context of PVs, we apply a context-aware version of entropy, paradigmatic variability, based on word embeddings and the close neighbours of a word in the vector space within a given radius (Teich et al., 2021). As previously mentioned, a drop in paradigmatic variability indicates a conventionalization phenomenon.

Regarding the word embeddings model, we used structured skip-grams (Ling et al., 2015) as it presents the advantage of representing each position in the left and right context separately, not as a mere bag of words as in simple skip-gram models.

The paradigmatic variability of a word over time is calculated by comparing period-specific word embedding models (i.e., per decade, from the 1660s to the 1990s). We followed the same procedure as presented in Teich et al. (2021), with the initialisation of the first decade being done with an atemporal embeddings model trained on the complete corpus as proposed by Fankhauser and Kupietz (2017). All following decades were then initialised with the embeddings of the previous decade. In our study, as our objective is to analyse PVs, the verbs and particles were joined

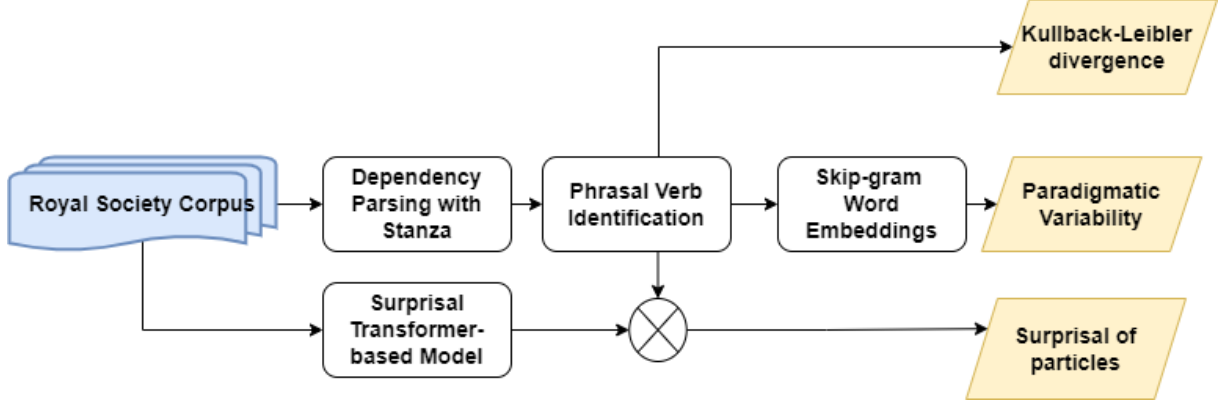


Figure 1: Experimental workflow.

with $||$ in the generation of the embedding space, thus allowing the differentiation between PVs and the other verbs.

This choice concerning the initialization process has the advantage of better representing low-frequency words in the embedding space, avoiding low-frequency words appearing in the centre of the space in the first few periods, and reducing bias regarding the movement in the embedding space over time. The subsequent statistical analysis of the vector space models only considers words with a frequency higher than 50^6 .

Once the word embeddings for each decade are obtained, the paradigmatic variability of a word x , $pvar(x)$, can be calculated as the entropy over a probability distribution, which is based on the probability $p(x_i|C_x)$ of a word x_i from the neighbourhood C_x being chosen instead of word x .

This is calculated using both the cosine similarity in the vector space between x_i and x and the frequency of x_i ($freq(x_i)$).

Thus:

$$\begin{aligned} pvar(x) &= H(P(\cdot|C_x)) \\ &= - \sum_{\cos(x_i, x) > \theta} p(x_i|C_x) \log(p(x_i|C_x)) \end{aligned} \quad (3)$$

$$\text{with } p(x_i|C_x) = \frac{\cos(x_i, x) \text{freq}(x_i)}{\sum_{x_j} \cos(x_j, x) \text{freq}(x_j)} \quad (4)$$

The θ threshold was set to 0.6 and we considered a maximum of 30 neighbours. Thus, a

⁶The other parameters used to generate the embeddings were: type 3; size 100; negative 10; hs 0; sample $1e-4$; threads 4; binary 0; and iter 5.

word with a homogeneous distribution of neighbours has a high value of $pvar(x)$.

4 Results

4.1 Kullback-Leibler Divergence

Figure 2 presents the relative entropy values (i.e., Kullback-Leibler divergence) of PVs in the RSC corpus over time as described in Section 3.2.1.

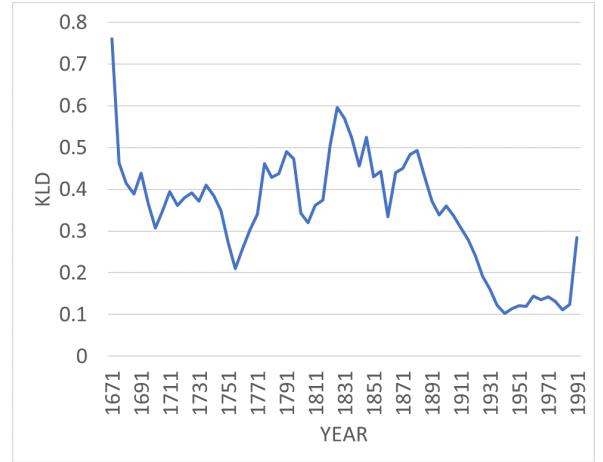


Figure 2: KLD measures for phrasal verbs in the Royal Society Corpus.

It is possible to observe peaks and troughs around the value of 0.4 from the seventeenth century to the end of the nineteenth century. On the other hand, at the beginning of the twentieth century, we clearly see a declining tendency of KLD, indicating, thus, a conventionalization in the usage of this feature, with a stabilization around 0.1 in the second half of this century.

To better understand the diachronic usage of PVs in scientific English, we also looked at the point-wise KLD, checking the relative entropy

shifts for each PV type. For each 20-year period used for the KLD calculation, we examined the number of PVs with positive values of divergence (i.e., PVs that became more typical), and the number with negative KLD (i.e., PVs that became less distinctive). Figure 3 shows the results of this analysis.

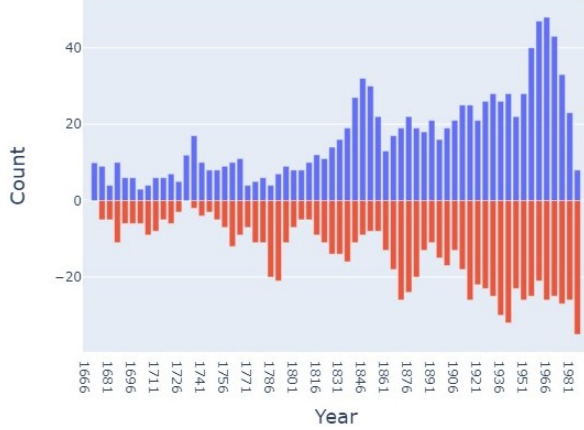


Figure 3: Number of phrasal verb types having positive (blue) or negative (red) KLD values per period of the RSC.

Besides the overall increasing trend in the number of PV types with positive KLD, it is possible to notice periods with higher increase, probably due to the specific textual needs of each period. Moreover, we can observe that the number of PVs with negative KLD also increases over time. Furthermore, the periods with more PVs having negative values of KLD, usually succeed periods where there is a peak in the number of PVs with positive values.

The increase in the number of PVs with positive KLD indicates a cyclical process of diversification (i.e., linguistic items acquiring different, more specific usages/meanings). Even though the overall relative frequency of PVs reduces over time, more different types are being used in specific periods. However, due to peaks regarding PVs with negative KLD, it seems that the usage of the new types does not become conventionalized.

4.2 Surprisal

As described in Section 3.2.2, another way of identifying possible conventionalization processes is using surprisal measures. PVs being MWEs, the surprisal of the particle is expected to be lower than the measure for the correspondent verbs. A decrease in time of the mean surprisal value of the

particles indicates a conventionalization regarding the usage of these grammatical constructions.

Figure 4 presents the plot of the mean surprisal values of the phrasal verbs present in the RSC per year.

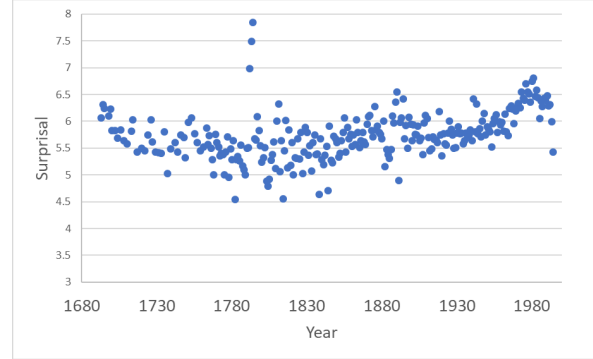


Figure 4: Mean surprisal value of phrasal verbs particles per year of the RSC.

Applying the Mann-Kendall trend test (Hussain and Mahmud, 2019), we observe that there is an overall statistically valid (i.e., a p-value below 0.00001) increasing tendency regarding the surprisal values of the particles.

This result can be correlated with the KLD observations. We observe that the conventionalization of the usage of PVs only happened in the twentieth century, moreover, throughout the centuries, we notice an increase in the usage of different PV types. Moreover, as shown by Alves et al. (2024), the relative frequency of PVs decreases over time in the RSC. All these factors contribute to an increase in the surprisal values.

In addition, it is also possible to notice that, even though there is an overall increasing tendency, there are periods with a decrease in the surprisal values and others with a more accentuated increase. When comparing Figures 3 and 4, we observe that periods with a high increase in the number of PV types (i.e., 1836-1856 and 1956-1976) also correspond to periods of accentuated increase regarding surprisal values.

Another factor that may influence the surprisal value is the distance between the verb and the particle. In the RSC, we find examples such as:

1. It suggested that development could be *broken down* into series of gene controlled chemical reactions. ($d = 1$, 1995)
2. ... which it gently touched with little or no damage, *blowing only off* a few tiles. ($d = 2$,

1695)

3. ... but BICHAT was continually *holding* a thing *up* by the wrong end ... ($d = 3$, 1823)
4. ... that his assistance should be sought to *bring* the new edition *up* to the existing state ... ($d = 4$, 1908)
5. ... and as I *wrote* many of the 336 them *down* from his own dictation ... ($d = 5$, 1840)

Thus, we decided to conduct an analysis of the diachronic evolution of the mean distance between verbs and particles in the RSC. Figure 5 shows the plot of the mean distance per 50-year period.

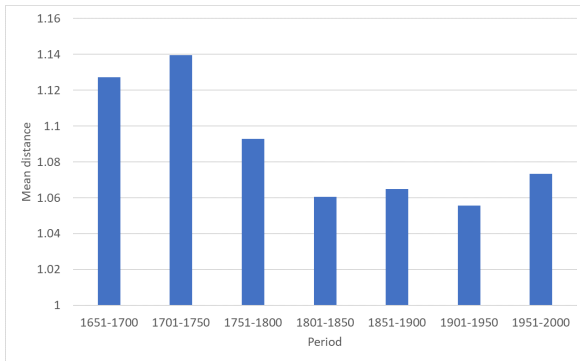


Figure 5: Mean distance between verbs and particles per 50-year of the RSC.

A statistical analysis of these results showed that p-value is below 0.001 for the following comparisons:

- 1701-1750 and 1751-1800
- 1751-1800 and 1801-1850
- 1901-1950 and 1951-2000

Thus, it is possible to notice a clear decrease in the mean value from the eighteenth century to the mid-nineteenth century, followed by a stabilization until the mid-twentieth century when a new increase is observed. The decreasing period regarding the mean distance between verbs and particles corresponds to a period with also a decrease in surprisal values (4). Moreover, the peak of surprisal (around 1970) is observed when there is also an increase in the mean distance.

4.2.1 Paradigmatic Variability

As previously explained in Section 3.2.3, using word embedding spaces, we calculated the paradigmatic variability of PVs per decade of the RSC. Figure 6 shows the results and the comparison with the variability of other verbs and all parts-of-speech in the dataset.

As shown by Teich et al. (2021), the paradigmatic variability of all words (i.e., all parts-of-speech) decreases over time as a general trend. This is due to two main mechanisms: conventionalization — a word becoming the dominant choice within its neighbourhood by frequency, possibly replacing other, alternative words — and diversification, i.e., words within a neighbourhood becoming more distant, leading to a split into two or more neighbourhoods.

Regarding non-phrasal verbs, they begin with a slightly higher paradigmatic variability than all POS, but end up with a tendency of lower variability. On the other hand, PVs start out with similar values as other verbs, but the paradigmatic variability decrease is much more accentuated, especially in the twentieth century, where the KLD measures already showed signs of conventionalization, as shown in Figure 2, and diversification (Figure 3).

Thus, it is possible to assume that the PVs have overcome a more accentuated process of conventionalization and diversification over time than other types of verbs in scientific English.

5 Discussion

In this study, our main objective was to analyse the contribution of PVs regarding the conventionalization processes happening in scientific English.

By analysing three different methods to measure linguistic shifts over time, we were able to notice that, although PVs are less common in scientific prose, they have undergone interesting diachronic phenomena.

Regarding the relative entropy measures (i.e., KLD), it was possible to notice that a conventionalization process occurred only throughout the twentieth century (Figure 2).

From the seventeenth to the twentieth century, although some small peaks and troughs can be observed, the KLD values did not change considerably. This tendency is different from what was observed by Degaetano-Ortlieb and Teich (2018) who analysed the whole lexicon (i.e., lemmas). In

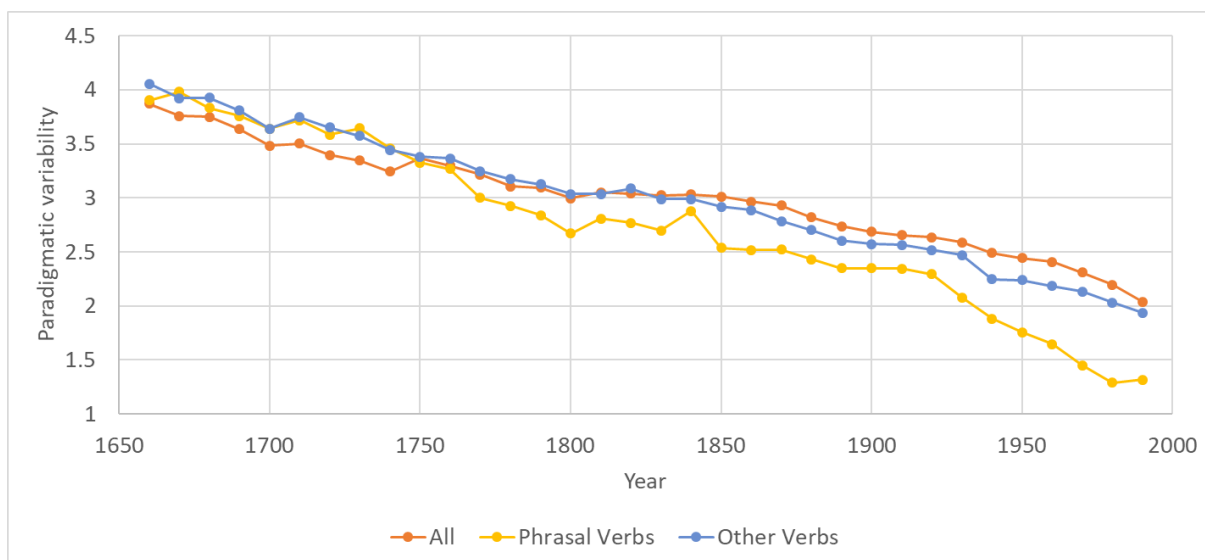


Figure 6: Paradigmatic Variability over time of phrasal Verbs compared to other verbs and all parts-of-speech in the RSC.

their study, the decreasing tendency is relatively constant throughout time.

Moreover, it was possible to verify that the conventionalization process occurs in parallel with a peak of diversification, as shown in Figure 3. This diversification has an impact on the surprisal measures, increasing the surprisal of the particles.

Both conventionalization and diversification processes are also confirmed with the paradigmatic variability analysis (Figure 6). PVs undergo a more accentuated decrease in their paradigmatic variability over time when compared to other verbs, principally during the twentieth century.

To better understand the peaks regarding the diversification of PVs, we analysed in detail the results of the KLD for each PV, per 20-year slice, present in the RSC.

What is possible to observe is that, throughout time, there are shifts regarding the PVs with peaks of KLD, i.e., verbs becoming more distinctive of specific periods.

Regarding the two main peaks of diversification identified in Figure 3, around 1846 and 1971, we can see that the PVs with the highest values of KLD in these periods differ considerably.

- 1846: *carry out, break up, filter off, bring out, split up, build up, map out, sum up, spread out, shut off.*
- 1971: *turn out, point out, rule out, end up, make up, go on, open up, break down, take on, bring together.*

A clear distinction can be made when analysing these two periods. In the nineteenth century, most PVs are linked to the description of experimental design, while in the twentieth century, there is a shift towards verbs focusing on the presentation of results, i.e., on the outcome of the research.

In Figure 3, we showed that the diversification is cyclic, new phrasal verbs are used in specific periods due to particular textual needs and, then, become less typical in future ones, however, in a more conventionalized way throughout the twentieth century.

Regarding the surprisal analysis, it is possible to notice that the shifts in the surprisal of the particles are a complex phenomenon. It is influenced not only by the peaks regarding the diversification process, but also by the changes regarding the distance between the verb and the particle, and, by the decrease regarding the relative frequency (i.e., with the verbs and particles appearing in more varied contexts, not as phrasal verbs).

6 Conclusions and Future Work

In this paper, we presented a multifaceted approach to characterize diachronic shifts regarding the usage of PVs in scientific English from the mid-seventeenth century to the end of the twentieth century by applying different information theory methods to the Royal Society Corpus.

By measuring the Kullback-Leibler divergence, we showed that the process of conventionalization

of PVs occurred mostly throughout the twentieth century. Moreover, we observed that peaks of diversification (i.e., increase in the number of PV types) happened in specific periods, followed by periods with a high number of PVs becoming less typical.

In terms of surprisal measure regarding the particles, we identified an overall tendency of increase, however, it was also possible to notice periods of accentuated increase, and some periods of decrease. These phenomena are probably correlated to the decrease regarding the relative frequency, to the peaks of diversification, and, to the distance between the verb and particles.

The analysis of the paradigmatic variability showed that PVs have a more accentuated decrease over time when compared to other verbs. This is probably due to the usage of PVs in specific contexts, where they cannot be replaced by similar terms. Moreover, the highest decrease regarding this measure was observed during the twentieth century, when a conventionalization phenomenon was detected using KLD.

Our findings not only enhance understanding of PVs in scientific English but also pave the way for future linguistic research, particularly in language evolution and specialized registers. In future work, we intend to proceed with the analysis by conducting a semantic analysis of the PVs with peaks of divergence to better understand their usage throughout time. Moreover, as part of a larger study, these results will be integrated with other types of MWEs (e.g., compounds, fixed expressions) to better understand the impact of the usage of these formulaic expressions in scientific texts throughout time.

Acknowledgements

This research is funded by *Deutsche Forschungsgemeinschaft* (DFG, German Research Foundation) – Project-ID 232722074 – SFB 1102.

References

- Rafael Alejo-González. 2010. Making sense of phrasal verbs: A cognitive linguistic account of L2 learning. *AILA review*, 23(1):50–71.
- Hasan Alisoy. 2023. Enhancing understanding of english phrasal verbs in first-year elt students through cognitive-linguistic methods.
- Diego Alves, Stefan Fischer, Stefania Degaetano-Ortlieb, and Elke Teich. 2024. Multi-word expressions in english scientific writing. In *Proceedings of the 8th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2024)*, pages 67–76.
- Tanya Avgustinova and Leonid Iomdin. 2019. https://link.springer.com/chapter/10.1007/978-3-030-30135-4_2 Towards a typology of microsyntactic constructions. In *Proceedings of the International Conference on Computational and Corpus-Based Phraseology*, pages 15–30.
- Douglas Biber and Federica Barbieri. 2007. Lexical bundles in university spoken and written registers. *English for specific purposes*, 26(3):263–286.
- Douglas Biber, Stig Johansson, Geoffrey Leech, Susan Conrad, and Edward Finegan. 2000. Longman grammar of spoken and written english.
- Douglas Biber, Stig Johansson, Geoffrey N Leech, Susan Conrad, and Edward Finegan. 2021. *Grammar of spoken and written English*. John Benjamins.
- David West Brown, Chris C Palmer, Michael Adams, Laurel J Brinton, and Roger D Fulk. 2015. The phrasal verb in american english: Using corpora to track down historical trends in particle distribution, register variation, and noun collocations. *Studies in the history of the English language VI: Evidence and method in histories of English*, 85:71–97.
- Stefania Degaetano-Ortlieb and Elke Teich. 2018. Using relative entropy for detection and analysis of periods of diachronic linguistic change. In *Proceedings of the 2nd SIGHUM LaTeCH-CLfL workshop*, pages 22–33.
- Stefania Degaetano-Ortlieb and Elke Teich. 2022. Toward an optimal code for communication: The case of scientific english. *Corpus Linguistics and Linguistic Theory*, 18(1):175–207.
- Peter Fankhauser, Jörg Knappen, and Elke Teich. 2014. Exploring and Visualizing Variation in Language Resources. In *LREC*, pages 4125–4128.
- Peter Fankhauser and Marc Kupietz. 2017. Visual correlation for detecting patterns in language change. In *Visualisierungsprozesse in den Humanities. Linguistische Perspektiven auf Prägungen, Praktiken, Positionen (VisuHu 2017). Tagung vom 17. bis 19. Juli 2017, Universität Zürich*. Universität Zürich.
- Stefan Fischer, Jörg Knappen, Katrin Menzel, and Elke Teich. 2020. The royal society corpus 6.0: Providing 300+ years of scientific writing for humanistic study. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 794–802.
- Md. Manjurul Hussain and Ishtiaq Mahmud. 2019. pyMannKendall: a python package for non parametric Mann Kendall family of trend tests. *Journal of Open Source Software*, 4(39):1556.

- Ken Hyland. 2008. As can be seen: Lexical bundles and disciplinary variation. *English for specific purposes*, 27(1):4–21.
- Solomon Kullback and Richard A Leibler. 1951. On Information and Sufficiency. *The annals of mathematical statistics*, 22(1):79–86.
- Wang Ling, Chris Dyer, Alan W Black, and Isabel Trancoso. 2015. Two/too simple adaptations of word2vec for syntax problems. In *Proceedings of the 2015 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1299–1304.
- Dilin Liu. 2012. The most frequently-used multi-word constructions in academic written english: A multi-corpus study. *English for Specific Purposes*, 31(1):25–35.
- Al-Otaibi Ghuzayyil Mohammed. 2019. A cognitive approach to the instruction of phrasal verbs: Rudzka-ostyn’s model. *Journal of Language and Education*, 5(2 (18)):10–25.
- Michelle Perdomo and Edith Kaan. 2023. Surprisal effects in priming of phrasal verb construction alternations in native speakers and l2 learners.
- Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. <https://nlp.stanford.edu/pubs/qi2020stanza.pdf> Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.
- Anna Siyanova-Chanturia and Diana Van Lancker Sittis. 2018. What online processing tells us about formulaic language. *Understanding formulaic language*, pages 38–61.
- Julius Steuer, Marie-Pauline Krielke, Stefan Fischer, Stefania Degaetano-Ortlieb, Marius Mosbach, and Dietrich Klakow. 2024. Modeling diachronic change in english scientific writing over 300+ years with transformer-based language model surprisal. In *Proceedings of the 17th Workshop on Building and Using Comparable Corpora (BUCC)@ LREC-COLING 2024*, pages 12–23.
- Elke Teich, Peter Fankhauser, Stefania Degaetano-Ortlieb, and Yuri Bizzoni. 2021. Less is more/more diverse: on the communicative utility of linguistic conventionalization. *Frontiers in Communication*, 5:620275.
- Chengxiang Zhai and John Lafferty. 2004. A Study of Smoothing Methods for Language Models Applied to Information Retrieval. *ACM Transactions on Information Systems (TOIS)*, 22(2):179–214.

Applying and Optimising a Multi-Scale Probit Model for Cross-Source Text Complexity Classification and Ranking in Swedish

Elsa Andersson, Johan Falkenjack, Arne Jönsson

Department of Computer and Information Science

Linköping University

Linköping, Sweden

elsa@anderssondito.se, {johan.falkenjack, arne.jonsson}@liu.se

Abstract

We present results from using Probit models to classify and rank texts of varying complexity from multiple sources. We use multiple linguistic sources including Swedish easy-to-read books and investigate data augmentation and feature regularisation as optimisation methods for text complexity assessment. Multi-Scale and Single Scale Probit models are implemented using different ratios of training data, and then compared. Overall, the findings suggest that the Multi-Scale Probit model is an effective method for classifying text complexity and ranking new texts and could be used to improve the performance on small datasets as well as normalise datasets labelled using different scales.

1 Introduction

Measuring or estimating text complexity is essential in various fields, including readability research and the adaptation and recommendation of texts for different audiences. In this paper, text complexity refers only to the linguistic characteristics that affect how easy or difficult a text is to read, without considering the interaction between the text and any particular reader.

Any comprehensive evaluation of text complexity must include three key components. First, linguistic features must be quantified, such as calculating the average sentence length. Second, relevant linguistic features need to be selected for evaluation. Third, the impact of each linguistic feature on text complexity must be assessed, for example, determining whether longer sentences increase or decrease complexity and to what extent. The distinction between effective and ineffective evaluations lies in the execution of these

components. The selection of features and the methods employed to measure them significantly affect the quality of the evaluation (Bailin and Grafstein, 2001).

Moreover, text complexity is not defined by a single superficial quality; rather, it results from an interplay of various features, each influencing complexity in distinct ways (Santini and Jönsson, 2020). Understanding how and to what extent each linguistic feature contributes to overall text complexity poses an additional challenge. The approaches for identifying and selecting linguistic features vary, ranging from employing theoretical linguistic frameworks and reasoning about feature impacts (Ellis, 2020) to training machine learning models on specific features and assessing their performance (Falkenjack et al., 2013), or even employing a combination of these methods.

Another aspect of the assessment of text complexity is the type of output that is produced. Depending on the purpose of the evaluation, the results may be in the shape of a single binary classification of "easy to read" or "not easy to read". This type of evaluation is traditionally realised through simple linear functions, or more recently using machine learning models like the Support Vector Machine (SVM) that splits texts into two classes (Benjamin, 2012). Another common evaluation method is to use one or a few linguistic features in a simple equation (often referred to as readability formulas) and computing a score to measure the complexity (e.g. the *Flesch Reading Ease formula* (Flesch, 1948)). These methods are beneficial in several ways, but all share a common downside. When using a few simple features or classifying texts in a binary manner, much nuance of text complexity is lost, and comparisons between texts are less informative (Bailin and Grafstein, 2001).

To solve these problems, we propose creating a model that uses many complex linguistic features

and classifies or ranks texts into non-binary levels. This approach would, however, usually require data that are already labelled according to class or rank. The more features used in the model to increase the complexity of the evaluation, the more data is required in each class or rank (Bengio et al., 2000).

One method that has the potential to resolve many of the issues mentioned above is the Multi-Scale Probit model, proposed and first implemented in Falkenjack (2018). The Probit model is a well established statistical model, introduced in the 1930s (Bliss, 1934b) and used primarily for classification. It is closely related to the younger but somewhat more well known *Logit model*, or *Logistic regression* as it is often called in psychometric contexts, but it has some properties which make it especially suitable for Bayesian modelling (McCulloch et al., 2000).

The Multi-Scale Probit model is a generalisation of the Bayesian Ordered Probit model and is capable of training on data labelled into ordered levels, such as how hard a text is to read, from multiple text sources. These sources may use completely different scales, meaning that the levels need not correspond in any sense between sources apart from indicating text complexity. There is no requirement for a minimum amount of texts per level, which enables the use of data that would have to be discarded in other approaches. The key idea behind the model is the presence of a *latent variable* that is shared among all labelling schemes. In this context, that latent variable is text complexity, with the assumption that the different labelling schemes used across different data sources all represent measures of that latent variable. Information about the latent variable is captured in the features, and the model learns how the latent variable is affected by the features, making it able to classify and even rank the text complexity of new texts.

We explore how the Multi-Scale Probit model performs when trained and evaluated on novel data, consisting of easy-to-read literature for children, teenagers, and adults¹.

2 Text complexity analysis

Text complexity generally refers to characteristics of a text that make it more or less cognitively engaging during reading (Vega et al., 2013). Quan-

titative and qualitative assessments of text complexity are of great value, as they can be used in many fields such as education (e.g. determining the appropriate material (Fitzgerald et al., 2015) or automatic essay grading (Valenti et al., 2003)), customisable text simplification (e.g. determining which texts to simplify (Štajner et al., 2012)), or customising texts based on cognitive requirements (e.g. for readers with dyslexia (Santini and Jönsson, 2020)).

Pinpointing the properties of a text that tells us about its complexity has been proven to be a difficult and confusing task. The factors that make up the complexity of a text can themselves create a hyperplane that spans across a highly multidimensional space.

Classification is a simplified version of this with the purpose of assigning texts into one or more classes such as "easy to read". Classification approaches consist of machine learning algorithms, statistical methods, and other NLP techniques.

Such approaches need to be trained on different text features or combinations of features and then evaluated on their performance in classifying texts accurately. As model performance becomes an indirect measurement of the relevance of the feature(s) to text complexity analysis, the features used to train models with better performance are chosen over the features of models with poorer performance (Falkenjack, 2018). Another category of classification algorithms is logistic regression and its variants. Compared to SVMs and similar methods, cf. Schwarm and Ostendorf (2005); Pitler and Nenkova (2008); Falkenjack (2018), the binary outcome is modelled as a probability between 0 and 1. For instance, a book could be classified as "easy to read" with a probability of 0.6, meaning that there is a 60% probability (according to the model) that the book is "easy to read". A common approach for such probability estimation is Logistic regression (Hosmer Jr et al., 2013), or the Logit model, and in this paper we apply a version of the closely related Probit model.

3 Text complexity features

The most commonly used method for the analysis of text complexity is automatic evaluation using quantifiable features of texts, which are then used to compute one or more ratings of text complexity. These features measure different aspects of the text and can be categorised into four ordered levels

¹<https://www.nyponochviljaforlag.se/om-oss/>

of increasing analytical depth, as outlined below.

Shallow features: The features in the first category do not contain information about the content of the text. They simply consist of letter and word counts; very little or no knowledge of their meaning is necessary to measure or understand shallow features. Nevertheless, they have been proven to be useful for measuring text complexity and are very simple to extract. The text is processed through tokenisation to create tokens out of words (and other components, e.g. delimiters). The tokens can be counted either as they are or by tallying the characters they contain. Several traditional metrics are based on one or more of these features or variants thereof, cf. Flesch (1948); Björnsson (1968).

Lexical composition: The lexical composition of a text targets frequencies of words based on the lexical category they belong to. The categorisation process includes lemmatising all words using a large vocabulary. For Swedish text, a vocabulary called *SweVoc* was developed in 2012 for this purpose by Mühlenbock and Kokkinakis (2012). In *SweVoc*, each word is represented as a lemma with some additional information depending on how it is used, including which category (or categories) it belongs to. In this research, the following categories will be used: *SweVocD* (words related to every-day matters), *SweVocH* (high-frequency words), and *SweVocTotal* (the total ratio of words in the text that are part of *SweVoc*). Because the *SweVoc* vocabulary is a subset of the Swedish language which excludes some complex or specialised words, it could be assumed that easy-to-read texts have a higher ratio of *SweVoc* words than more complex texts.

Morpho-syntactic features: Morpho-syntactic features include tagging words and tokens according to their part-of-speech (POS). The POS tags can then be used in a number of text features. In this research, *UnigramPOS* features will be used. The *UnigramPOS* features are the probabilities of a unigram occurring in a text, expressed as the ratio of each POS tag per token. Calculating the unigram probabilities of a text is a type of language modelling that can be effective in measuring the readability of a text (Heilman et al., 2007).

Syntactic features: Although unigram language models are effective in capturing content information and variations in word usage, they lack the ability to capture syntactic information. The

analysis of syntactic complexity requires parsing of the text, which involves mapping words and phrases and their dependencies based on grammatical structures of a sentence. For this research, the syntactic text features consist of a subset of features extracted through dependency parsing on each sentence. These features are: *UnigramDep* (probabilities for each dependency relation type), *RightDep* (ratio of total dependencies where the headword occurs after the dependent word), *UVA* (unigram probabilities for verbs with a specific number of dependants), and *Lexical density* (ratio of content words).

4 Text complexity as a latent variable

A key idea behind the use of statistical models for text complexity assessment is the assumption of a latent variable. As established, text complexity cannot be measured directly. Instead, it is estimated using one or more linguistic features. Furthermore, text complexity is assessed and labelled in various ways. For example, texts may be labelled as "easy to read" (with the implication that regular texts are less "easy to read"), rated on a scale of 1 to 7, or categorised into age groups, among other methods. Although texts from different sources may use varying labels and methods to measure readability, we assume that they share the underlying *latent variable* of text complexity. In other words, variations in text complexity may be expressed differently, but the concept is consistently modelled across all sources. If data can be processed appropriately, it enables the latent variable to be statistically modelled and subsequently used to classify or rank texts.

4.1 The Probit model

The Multi-Scale Probit model we use is a generalisation of the Ordered Probit model which itself is a generalisation of the Probit model. The Probit model can be viewed as a linear binary classifier. It can also be considered a *Generalized Linear Model* with the inverse of the cumulative distribution function (CDF) of the Standard Normal distribution, the *Probit* function (Bliss, 1934a), as link function. In essence, the Probit model takes a vector of covariates \mathbf{x}_i of the i th observation and uses it to predict the outcome, or label, y_i . It does so by estimating a coefficient vector β that represents the effects of \mathbf{x}_i on the value of y_i . In simple terms, the model can be expressed as "what is the

probability that y_i is 1, given the information in \mathbf{x}_i ?”. Mathematically, the Probit model can be expressed as

$$P(y_i = 1 | \mathbf{x}_i) = \Phi(\alpha + \mathbf{x}_i^T \boldsymbol{\beta}), \quad (1)$$

where Φ is the CDF of the Standard Normal distribution and α is the *intercept*, defined as a constant value that represents the baseline probability of y_i being 1 even if all covariates are 0.

In the context of text complexity, \mathbf{x}_i would consist of measurements of linguistic features and y_i would represent a certain label, for example “easy to read”. Furthermore, the Probit model can generally be conceptualised as a latent variable model, the latent variable y^* in our application being text complexity. By setting a threshold $\gamma = -\alpha$ and denoting the two binary outcomes as 1 and 2, the Probit model can instead be expressed as

$$y_i = \begin{cases} 2 & \text{if } y_i^* > \gamma \\ 1 & \text{otherwise} \end{cases} \quad \text{where} \quad y_i^* = \mathbf{x}_i^T \boldsymbol{\beta} + \epsilon_i \quad (2)$$

where y_i^* represents the value of the latent variable and ϵ_i is the error term for the i th observation. Under this interpretation, we can view the Probit model as a linear regression over an unobserved, or latent, real-valued variable which underlies the assigned labels in the classification problem. If class 1 represents “easy to read” and class 2 “not easy to read”, this can be expressed as “if the complexity of a text is above a certain threshold, it should be classified as ‘not easy to read’, otherwise it should be classified as ‘easy to read’”.

This latent variable formulation can be generalised to the case of an ordinal response variable with possible outcomes $C_1 \dots C_m$ by introducing further thresholds $\gamma_1 \dots \gamma_{m-1}$ giving rise to the Ordered Probit model:

$$y_i = \begin{cases} C_1 & \text{if } y_i^* \leq \gamma_1, \\ C_2 & \text{if } \gamma_1 < y_i^* \leq \gamma_2, \\ \vdots & \\ C_m & \text{if } y_i^* > \gamma_{m-1} \end{cases} \quad (3)$$

where y_i^* is the same as in Equation 2.

The latent variable interpretation of Probit models lends itself especially well to a Bayesian approach. Essentially, a Bayesian approach entails declaring a prior belief, which is then updated using Bayes Theorem as new evidence is gathered,

generating a posterior belief based on that evidence. These beliefs are commonly referred to as simply the prior and the posterior. Bayes’ theorem can be applied for inference of the posterior probability distribution of the coefficients vector $\boldsymbol{\beta}$ and thresholds γ according to the following formulation

$$P(\boldsymbol{\beta}, \gamma | \mathbf{y}, \mathbf{X}) \propto P(\mathbf{y} | \boldsymbol{\beta}, \gamma, \mathbf{X}) P(\boldsymbol{\beta}, \gamma), \quad (4)$$

where $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)^T$, $\mathbf{y} = (y_1, \dots, y_n)^T$, $P(\boldsymbol{\beta}), \gamma$ is the prior and $P(\mathbf{y} | \boldsymbol{\beta}, \mathbf{X})$ is the likelihood function. Although this posterior distribution is mathematically intractable, the *Markov Chain Monte Carlo* (MCMC) simulation can be used to estimate the posterior. Gibbs samplers for both the binary (Albert and Chib, 1993) and ordinal (Cowles, 1996) versions are well established.

The goal of the sampling process for the formulation in Equation 4 is to approximate the joint posterior distribution of $\boldsymbol{\beta}$ by estimating marginal distributions of individual variables.

4.2 The Multi-Scale Probit model

The formulation for the Probit model as a model for the latent variable in Equation 3 can be extended further to fit binary and non-binary data labelled on different scales. Let us take a practical example to demonstrate these characteristics. Say we have books sourced from two publishers, *A* and *B*. Publisher *A* labels its books on a scale from ‘easy’, ‘medium’ to ‘hard’ based on readability. Publisher *B* labels its books on a scale from 1 to 5, also based on readability. The publishers use unknown and possibly different methods for measuring readability, the difference in complexity between each level within either scale is unknown, and there is no known function translating between the scales. The only assumption we make is that the labels are ordered and that they constitute measures of the same phenomenon, i.e. text complexity. The Multi-Scale Probit model uses one set of thresholds to discriminate between levels for each scale such that $\gamma^{(s)}$ is the set of thresholds for scale s . Using our example, the two sets would be $\gamma^{(A)} = \{\gamma^{(A_{easy})}, \gamma^{(A_{medium})}, \gamma^{(A_{hard})}\}$ and $\gamma^{(B)} = \{\gamma^{(B_1)}, \gamma^{(B_2)}, \gamma^{(B_3)}, \gamma^{(B_4)}, \gamma^{(B_5)}\}$. Furthermore, the model fits a single latent variable y^* to all data, meaning that only a single coefficient vector $\boldsymbol{\beta}$ is estimated. The Multi-Scale Probit model

can therefore be expressed as

$$y_i = \begin{cases} C_1^{(s_i)} & \text{if } y_i^* \leq \gamma_1^{(s_i)}, \\ C_2^{(s_i)} & \text{if } \gamma_1^{(s_i)} < y_i^* \leq \gamma_2^{(s_i)}, \\ \vdots & \\ C_m^{(s_i)} & \text{if } \gamma_{m-1}^{(s_i)} < y_i^* \end{cases} \quad (5)$$

for observation $i = 1, \dots, n$, where y_i^* is the same as in Equations 2 and 3, the response label y_i is measured on scale s_i , and $C_1^{(s_i)} \dots (C_m^{(s_i)})$ denotes the labels for scale s_i . The complete posterior distribution of the joint is estimated using a variation of the Gibbs sampling algorithm proposed by Cowles (1996) for the Ordinal Probit model. The conditional posteriors for all sets of $\gamma^{(s)}$, β and the latent variable y^* can be sampled through the process described above. The latent variable estimated by the Multi-Scale Probit model can be used to order data samples, enabling total ranking of all data samples. Essentially, the Multi-Scale Probit allows us to, from some number of disjunct and partially ordered sets, estimate a total order on the union of all sets.

The applicability of the Multi-Scale Probit to our domain has previously been investigated in Falkenjack et al. (2018).

4.3 Measures for evaluation

Because the Multi-Scale Probit model can be used for both classification and ranking, we want to evaluate it using appropriate measures for each purpose.

As the data we use are not balanced, i.e. there is not a consistent number of observations per class, straight *accuracy* would not be a suitable metric if we consider the performance as equally important for all classes. In such cases, it is common to use the *macro-averaged* F_1 -score (Murphy, 2012, p. 185). The F_1 -score of a single class is the harmonic mean of the *precision* and the *recall* for that class. The macro-averaged F_1 -score is the average of the F_1 -scores for all classes. This value can be used as an overall measurement of how well the model performs in regards to classification.

The Multi-Scale Probit estimates a numeric latent variable and can thus be viewed as a model for ranking in addition to classification. We evaluate this performance by computing the Kendall rank correlation coefficient, τ , between the estimated latent variable and the known observed variable.

Kendall’s τ assesses the ordinal association between two variables and gives a score between -1 and 1 depending on the correlation. Since the observed variable is an ordinal class, giving rise to a large number of ties, we use a modified version called Kendall’s τ_B specifically made to handle such situations (Kendall, 1945).

Just as the F -measure uses the harmonic mean between *Precision* and *Recall*, we can combine the classification performance F_1 and ranking performance Kendall’s τ_B using the harmonic mean. We use this as a combined performance metric for both classification and ranking in our figures in Section 7.

5 Data

The majority of data used in this research consisted of books from a corpus called Nypon-Vilja, consisting of books from *Nypon och Vilja*, the largest Swedish publisher of easy-to-read literature for children, teenagers, and adults. Swedish easy-to-read literature is catered to people with reading difficulties, beginner readers, or non-native readers learning Swedish.

Books from *Nypon* and *Vilja* are (generally) aimed at two different target groups; *Nypon* at ‘children and young’ and *Vilja* at ‘young adults and adults’. The publisher uses separate scales (with their own naming schemes), each consisting of 6 levels, to indicate how easy or difficult a book is, where the first level (1 and X-Small) is the easiest and the last level (6 and XX-Large) the most difficult.

Before processing, all books were manually annotated based on their alignment with one of two narrativity dimensions: *informational* and *narrative* (McNamara, 2013). *Informational* text tends to be non-fictional, written to inform about or explain a specific topic. *Narrative* text on the other hand is typically fictional and story-driven. In order to minimise the effects of variations in language use that affect text complexity between dimensions of narrativity, only books classified as *narrative* are used. Finally, as level 6 from *Nypon* contained only 2 books, they were merged with the books in level 5. This resulted in a dataset of 356 books with 5 levels in *Nypon* and 6 levels in *Vilja*, summarised in Table 1.

The Stockholm-Umeå Corpus (Ejerhed et al., 2006) (*SUC*) is a large collection of annotated Swedish texts written in the 90’s. It contains

Nypon		Vilja	
Level	N samples	Level	N samples
1	48	X-Small	4
2	59	Small	14
3	68	Medium	20
4	46	Large	42
5	13	X-Large	38
		XX-Large	4
Sum	234		122

Table 1: Number of data samples from each level.

texts in 10 categories including newspaper reportage, popular lore, and imaginative prose, written for different audiences and with varying writing styles. The annotations contain information about linguistic, structural, and functional information. In this research, we used a free-for-use bag-of-sentences version (*SUCX 3.0*) publicly available from *Språkbanken*². Thus, no text features dependent on sentence order are included in our analysis. Furthermore, in order to minimise the effects of variations in language use that affect text complexity between genres (Štajner et al., 2012; Hiebert, 2012; Dell’Orletta et al., 2014), only texts from the category ‘imaginative prose’ were extracted, giving a total of 127 texts from *SUC*. This category was assumed to contain texts in a style the most similar to those extracted from *Nypon och Vilja*, being non-informational. The purpose of using *SUC* is to obtain a composition of data at a level of text complexity above all books from *Nypon och Vilja*. This is a key assumption and is based on the rationale that texts from *SUC* are written for typical adult readers and not with the express purpose of being especially easy to read, meaning text complexity can be assumed to be higher compared to the books from *Nypon och Vilja*.

To extract all necessary linguistic features, all texts were processed using the StilLett API Service (SAPIS) (Fahlborg and Rennes, 2016). The API service allows for the tokenization, lemmatisation, part-of-speech tagging, and dependency parsing of any text input. It also allows for text complexity analysis through the SCREAM module (Falkenjack et al., 2013) which computes related metrics.

²<https://spraakbanken.gu.se/>

6 Model implementation and evaluation

The Multi-Scale model was implemented using a modified version of the framework developed by Falkenjack (2018) and executed using R (version 3.6.3) with *RStudio* (RStudio Team, 2022). The model uses a set of covariates as input. These covariates are the values of metrics extracted through the data processing step resulting in a total of 47 features, c.f. (Falkenjack, 2018)

Data containing values for all covariates in the feature set were split into 5 classes for the *Nypon* scale and 6 classes for the *Vilja* scale ordered according to their levels. The data were then first split into training and test sets 500 times, using different ratios for training and test data, creating 500 models. The training data were used to estimate the full joint posterior distribution described in Section 4.2 through sampling according to the scheme described in Falkenjack (2018). This step was completed to evaluate the performance of the models. Then, instead of splitting the data into training and test sets, all data were used to run 20 chains of the Gibbs sampler resulting in a combined set of samples of a full posterior distribution of the entire dataset. The number of chains was based on the number of CPU cores available, using one core per chain to speed up the sampling process.

Furthermore, the Multi-Scale model by definition uses *multiple scales*, meaning the posterior distribution is sampled using data from both *Nypon* and *Vilja*. However, since the Multi-Scale Probit is a generalised version of an Ordered Probit model, which uses only one scale, its performance on either scale can be compared with a traditional Ordered Probit sampled using data from only that dataset. Such models used the same implementation as the Multi-Scale model, but using one scale at a time. These models will be referred to as *Single Scale* models.

After completing the sampling processes, parts of the resulting posterior distributions were visualised. The posterior was also used for classification and ranking, where the performance was evaluated by computing values for several evaluation metrics. For all evaluation metrics above, a higher positive value indicates better performance. For visualisation purposes and to enable easier comparison between model distributions, mode values are also plotted. The mode corresponds to the point with the highest probability density of a dis-

tribution, i.e. equivalent to the value that appears most frequently in a discrete probability distribution.

7 Model performance

The data was randomly separated into 500 different permutations of training and test data, and models were trained and evaluated for each such permutation. To assess whether one model generally outperforms the other we compute the difference between the posterior mean performance (F_1 -scores and *Kendall's* τ_B correlations, respectively, as well as the harmonic mean of them) between the models for each permutation. Finally, to further examine how well the models perform given varying amounts of training data, two different split ratios were used: 2/3 training and 1/3 test data, and vice versa.

With 2/3 of the data used for training, the mean posterior modes of the performance metrics F_1 and *Kendall's* τ_B can be seen in Table 2.

	$F_1^{(M)}$	$F_1^{(S)}$	$\tau_B^{(M)}$	$\tau_B^{(S)}$
Nypon	0.35	0.35	0.45	0.45
Vilja	0.27	0.4	0.25	0.27

Table 2: Model performance using 2/3 of the data for training.

We can see that the choice of model makes little difference to the performance on the *Nypon* dataset but has a noticeable impact for the *Vilja* dataset.

Direct comparison of the models is done by computing the difference of the posterior mean F_1 and *Kendall's* τ_B for each model over the 500 data permutations. This shows that the Multi-Scale model outperforms the Single Scale model with respect to the F_1 -score 54.4% of the time on the *Nypon* dataset and 73.4% of the time on the *Vilja* dataset. The same comparison of *Kendall's* τ_B show that the Multi-Scale model is better 62% of the time on the *Nypon* dataset and 89.2% of the time on the *Vilja* dataset. Figure 1 plots the distribution of differences in the harmonic mean of F_1 -score and *Kendall's* τ_B between the models over all 500 data permutations, showing that the Multi-Scale model is better in 58.4% and 85.4% of cases for *Nypon* and *Vilja* respectively when both performance metrics are considered.

When the ratio of training to test data is reversed (i.e. 1/3 of the data used for training, the

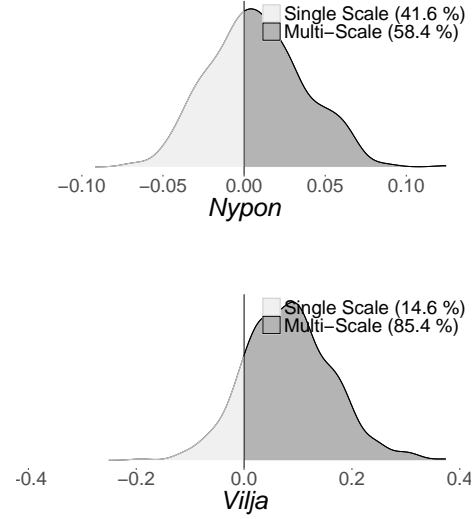


Figure 1: The posterior distribution for the difference in posterior harmonic mean of F_1 -score and *Kendall's* τ_B between the Multi-Scale and Single Scale models. (2/3 of the data used for training.)

rest for testing), we see similar differences in overall performance on the *Vilja* dataset but now, the the difference in overall performance on the *Nypon* also shows a marked difference. Figure 2 illustrates this for the harmonic mean of F_1 -score and *Kendall's* τ_B . However, as expected, the performance of both models is slightly lower with mean posterior modes, as seen in Table 3.

	$F_1^{(M)}$	$F_1^{(S)}$	$\tau_B^{(M)}$	$\tau_B^{(S)}$
Nypon	0.3	0.29	0.33	0.3
Vilja	0.25	0.24	0.3	0.24

Table 3: Model performance using 1/3 of the data for training.

This implies that the Multi-Scale model is especially useful when the availability of training data is limited.

Meanwhile, Figure 2 shows that the Multi-Scale model is better in 78.6% and 84.2% of cases for *Nypon* and *Vilja* respectively when both performance metrics are considered.

To summarise, the results show that the Multi-Scale model generally outperforms the Single Scale model on both datasets, particularly on the *Vilja* texts. Furthermore, this performance difference was greater when using a data split of 1/3 training data and 2/3 test data compared to a 2/3 training and 1/3 test data split. This implies that

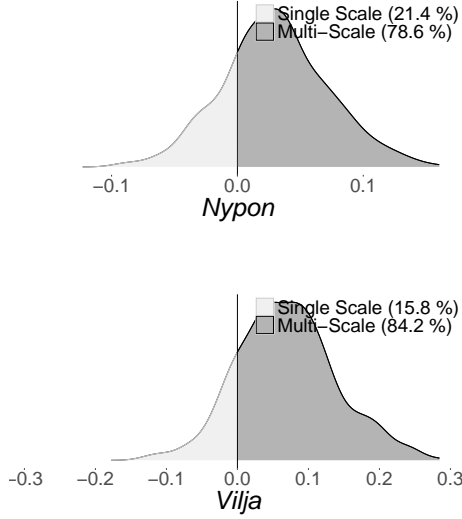


Figure 2: The posterior distribution for the difference in posterior harmonic mean of F_1 -score and Kendall's τ_B between the Multi-Scale and Single Scale models. (1/3 of the data used for training.)

the relative improvement of the Multi-Scale Probit over the single scale Ordered Probit decreases with the size of the dataset available for training a single scale model. In other words, the Multi-Scale Probit model is especially useful in contexts with sparse and diverse data for training.

7.1 Results of data augmentation

Augmenting both scales with the *SUC* corpus added an additional level to both scales above the other levels. The results of models using 500 augmented data sets with a data ratio of 2/3 training and 1/3 testing show that the modes of the posterior F_1 -score and Kendall's τ_B show a marked improvement with an augmented dataset as shown in Table 4.

	$F_1^{(M)}$	$F_1^{(S)}$	$\tau_B^{(M)}$	$\tau_B^{(S)}$
Nypon	0.47	0.47	0.71	0.71
Vilja	0.4	0.32	0.7	0.6

Table 4: Model performance using 2/3 of the augmented data for training.

Figure 3 shows that when considering both F_1 -score and Kendall's τ_B the Multi-Scale model outperforms the Single Scale model most of the time for both datasets.

Using 1/3 of the data for training and 2/3 for testing, as shown in Table 5, reinforces what we

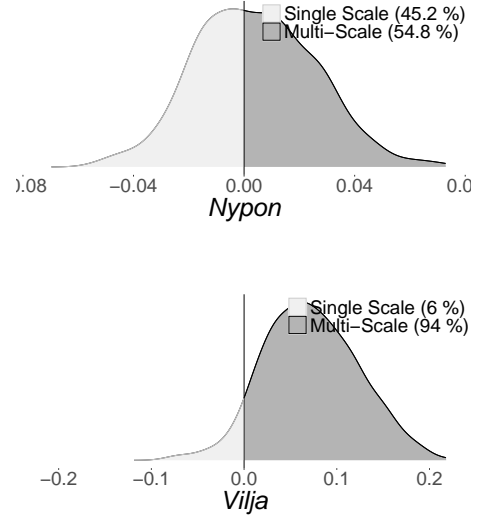


Figure 3: The posterior distribution for the difference in posterior harmonic mean of F_1 -score and Kendall's τ_B between the Multi-Scale and Single Scale models. (2/3 of the *augmented* data used for training.)

saw with the original data and with the 2/3 training ratio with augmented data. There is only a small improvement on the larger *Nypon* dataset but a more noticeable improvement on the smaller *Vilja* dataset.

	$F_1^{(M)}$	$F_1^{(S)}$	$\tau_B^{(M)}$	$\tau_B^{(S)}$
Nypon	0.4	0.39	0.68	0.67
Vilja	0.35	0.33	0.68	0.56

Table 5: Model performance using 1/3 of the augmented data for training.

Figure 4 again shows the Multi-Scale model outperforming the Single Scale model most of the time for both datasets.

The results show that, just as in the previous section, the Multi-Scale model generally outperforms the Single Scale model, particularly when tested on the smaller *Vilja* dataset, and that this improvement is greater when reducing the ratio of training data.

7.2 Results of regularisation

The regularisation process consisted of inspecting the marginal posteriors for each of the original 47 features, removing features with a positive or negative influence certainty % below specific thresholds (25%, 50% and 75%), and then resampling

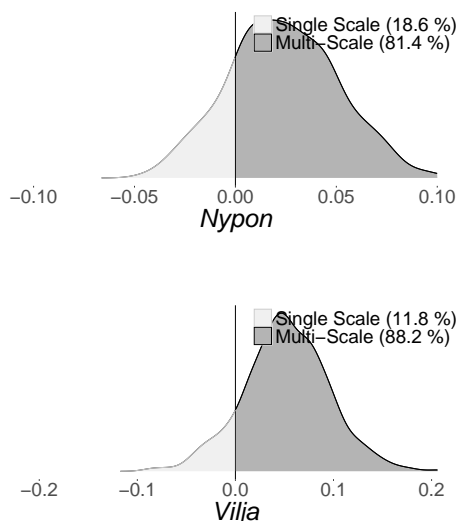


Figure 4: The posterior distribution for the difference in posterior harmonic mean of F_1 -score and Kendall's τ_B between the Multi-Scale and Single Scale models. (1/3 of the *augmented* data used for training.)

the posterior distributions with each of the three reduced feature sets. The purpose of the regularisation process was to examine whether the change of feature set affects the predictive capabilities of the models.

Using 500 data sets with a ratio of 2/3 training data and 1/3 testing data, the modes of the posterior distribution of the harmonic mean align nearly perfectly for both models across all three feature sets when tested on the *Nypon* scale. On the *Vilja* scale, the Multi-Scale model slightly outperforms the Single Scale model across all feature sets. Furthermore, the modes for both models increase slightly between the first and second feature sets when tested on both scales. On the *Nypon* scale, there is an increase between the second and third feature sets, but no noticeable increase when tested on the *Vilja* scale.

The training/test split was again reversed (1/3 training, 2/3 test) on 500 reduced feature sets of data. The performance results of the models show that the modes of the posterior distribution of the harmonic mean are marginally higher for the Multi-Scale model compared to the Single Scale model across all feature sets when tested on the *Nypon* scale. On the *Vilja* scale, the difference in modes between the two models is greater, approximately 0.1 higher for the Multi-Scale model across all feature sets. Furthermore, the modes for both

models increase slightly between the first and second feature set when tested on both scales, and a larger increase between the second and final feature set when tested on the *Nypon* scale, but not the *Vilja* scale.

8 Conclusion

The purpose of this research was to utilise the Multi-Scale Probit model in order to enable a standardised ranking and classification of text complexity, while also exploring how the model can be optimised. The assessment of text complexity can be used for a wide range of purposes, making its development pivotal in the field of natural language processing. The results from applying the Multi-Scale Probit on easy-to-read Swedish books have indicated that the model outperforms the Single Scale model in nearly all cases of classification and ranking, measured by F_1 -scores and Kendall τ_B correlations. Furthermore, the results accentuate how the output from the Multi-Scale Probit model can be used in a simple manner to classify and rank new texts in the same domain, or adapted to other domains by creating new models. Through data augmentation and feature regularisation, the model can be optimised in terms of computational complexity and performance in specific contexts. The ability of the Multi-Scale Probit model to utilise data from different sources, without the necessity of large data quantities per category, enables assessments of text complexity that have previously not been possible. This research has contributed to the goal of developing methods for classifying and ranking text complexity, with the broader aim of creating a more accessible society for readers with varying needs.

Acknowledgments

This research is part of the project Text Adaptation for Increased Reading Comprehension, funded by The Swedish Research Council.

References

- James H Albert and Siddhartha Chib. 1993. Bayesian Analysis of Binary and Polychotomous Response Data. *Source Journal of the American Statistical Association*, 88(422):669–679.
- Alan Bailin and Ann Grafstein. 2001. The linguistic assumptions underlying readability formulae: A critique. *Language & communication*, 21(3):285–301.

- Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. 2000. A neural probabilistic language model. *Advances in neural information processing systems*, 13.
- Rebekah George Benjamin. 2012. Reconstructing readability: Recent developments and recommendations in the analysis of text difficulty. *Educational Psychology Review*, 24:63–88.
- CH Björnsson. 1968. Läsbarhet [readability] stockholm. Sweden: Liber.
- C. I. Bliss. 1934a. The method of probits. *Science*, 79(2037):38–39.
- Chester I. Bliss. 1934b. The Method of Probits. *Science*, 79(2037):38–39.
- Mary Kathryn Cowles. 1996. Accelerating monte carlo markov chain convergence for cumulative-link generalized linear models. *Statistics and Computing*, 6:101–111.
- Felice Dell’Orletta, Simonetta Montemagni, and Giulia Venturi. 2014. Assessing document and sentence readability in less resourced languages and across textual genres. *ITL-International Journal of Applied Linguistics*, 165(2):163–193.
- Eva Ejerhed, Gunnel Källgren, and Benny Brodda. 2006. Stockholm umeå corpus version 3.0.
- Nick C. Ellis. 2020. Theoretical frameworks in 12 acquisition. In Patrick Rebuschat, editor, *The Cambridge Handbook of Language Learning*, chapter 4, pages 143–188. Cambridge University Press.
- Daniel Fahlborg and Evelina Rennes. 2016. Introducing sapis-an api service for text analysis and simplification. In *The second national Swe-Clarin workshop: Research collaborations for the digital age, Umeå, Sweden*.
- Johan Falkenjack. 2018. *Towards a model of general text complexity for swedish*. Ph.D. thesis, Linköping University Electronic Press.
- Johan Falkenjack, Katarina Heimann Mühlenbock, and Arne Jönsson. 2013. Features indicating readability in swedish text. In *Proceedings of the 19th Nordic Conference of Computational Linguistics (NODAL-IDA 2013)*, pages 27–40.
- Johan Falkenjack, Mattias Villani, and Arne Jönsson. 2018. Modeling text complexity using a multi-scale probit. *arXiv preprint arXiv:1811.04653*.
- Jill Fitzgerald, Elfrieda H Hiebert, Kimberly Bowen, E Jackie Relyea-Kim, Melody Kung, and Jeff Elmore. 2015. Text complexity: Primary teachers’ views. *Literacy Research and Instruction*, 54(1):19–44.
- Rudolph Flesch. 1948. A new readability yardstick. *Journal of applied psychology*, 32(3):221.
- Michael Heilman, Kevyn Collins-Thompson, Jamie Callan, and Maxine Eskenazi. 2007. Combining lexical and grammatical features to improve readability measures for first and second language texts. In *Human language technologies 2007: the conference of the North American chapter of the association for computational linguistics; proceedings of the main conference*, pages 460–467.
- Elfrieda H Hiebert. 2012. Readability and the common core’s staircase of text complexity. *Text Matters*, 1.
- David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant. 2013. *Applied logistic regression*. John Wiley & Sons.
- M. G. Kendall. 1945. The treatment of ties in ranking problems. *Biometrika*, 33(3):239–251.
- Robert E. McCulloch, Nicholas G. Polson, and Peter E. Rossi. 2000. A Bayesian analysis of the multinomial probit model with fully identified parameters. *Journal of Econometrics*, 99(1):173–193.
- Danielle S McNamara. 2013. The epistemic stance between the author and reader: A driving force in the cohesion of text and writing. *Discourse Studies*, 15(5):579–595.
- Katarina Heimann Mühlenbock and Sofie Johansson Kokkinakis. 2012. Swevoc-a swedish vocabulary resource for call. In *Proceedings of the SLTC 2012 workshop on NLP for CALL*, pages 28–34. Citeseer.
- Kevin P Murphy. 2012. *Machine learning: a probabilistic perspective*. MIT press.
- Emily Pitler and Ani Nenkova. 2008. Revisiting readability: A unified framework for predicting text quality. In *Proceedings of the 2008 conference on empirical methods in natural language processing*, pages 186–195.
- RStudio Team. 2022. *RStudio: Integrated Development Environment for R*. RStudio, PBC, Boston, MA.
- Marina Santini and Arne Jönsson. 2020. Pinning down text complexity: An exploratory study on the registers of the stockholm-umeå corpus (suc). *Register Studies*, 2(2):306–349.
- Sarah E Schwarm and Mari Ostendorf. 2005. Reading level assessment using support vector machines and statistical language models. In *Proceedings of the 43rd annual meeting of the Association for Computational Linguistics (ACL’05)*, pages 523–530.
- Sanja Štajner, Richard Evans, Constantin Orasan, and Ruslan Mitkov. 2012. What can readability measures really tell us about text complexity. In *Proceedings of workshop on natural language processing for improving textual accessibility*, pages 14–22. Citeseer.

Salvatore Valenti, Francesca Neri, and Alessandro Cucchiarelli. 2003. An overview of current research on automated essay grading. *Journal of Information Technology Education: Research*, 2(1):319–330.

Benjamin Vega, Shi Feng, Blair Lehman, Art Graesser, and Sidney D'Mello. 2013. Reading into the text: Investigating the influence of text complexity on cognitive engagement. In *Educational Data Mining 2013*.

Playing by the Rules: A Benchmark Set for Standardized Icelandic Orthography

**Bjarki Ármannsson, Hinrik Hafsteinsson, Jóhannes B. Sigtryggsson,
Atli Jasonarson, Einar Freyr Sigurðsson, Steinþór Steingrímsson**

The Árni Magnússon Institute for Icelandic Studies

`bjarki.armannsson@arnastofnun.is, hinhaf@hi.is,
{johannes.b.sigtryggsson, atli.jasonarson,
einar.freyr.sigurdsson, steinthor.steingrimsson}@arnastofnun.is`

Abstract

We present the Icelandic Standardization Benchmark Set: Spelling and Punctuation (IceStaBS:SP), a dataset designed to provide standardized text examples for Icelandic orthography. The dataset includes non-standard orthography examples and their standardized counterparts, along with detailed explanations based on the official Icelandic spelling rules. IceStaBS:SP aims to support the development and evaluation of automatic spell and grammar checkers, particularly in an educational setting. We evaluate various spell and grammar checkers using IceStaBS:SP, demonstrating its utility as a benchmarking tool and highlighting areas for future improvement.

1 Introduction

Digital language infrastructure, not least for spell and grammar checking, is a productive and growing field within Icelandic Language Technology. Although various datasets have been produced, which in turn have been used to develop and improve spell and grammar checking software, there is a lack of datasets which provide a 1:1 mapping between spelling errors and formalized rules regarding standard orthography (spelling rules).

In this paper, we present the Icelandic Standardization Benchmark Set: Spelling and Punctuation (IceStaBS:SP, Ármannsson et al. 2024), a dataset of examples of text standardization along with thorough explanations of how and why text has been altered. The dataset is based on the official spelling rules for Icelandic.¹ Our goal is to provide a standardized benchmark for evaluating the performance of spell and grammar checkers, thereby contributing to the improvement of digital language tools for Icelandic.

¹<https://ritreglur.arnastofnun.is/>

The paper is structured as follows: Section 2 provides an overview of related work in the field of Icelandic spell and grammar checking, most importantly existing datasets. Section 3 describes the structure of the IceStaBS:SP dataset and the methodology behind it. Section 4 outlines the evaluation experiment we performed to gauge the efficacy of the dataset as a benchmarking tool for orthography. Section 5 presents the results of the evaluation, Section 6 discusses the limitations of our approach and Section 7 concludes the paper with a discussion of the implications of our findings and suggestions for future work.

2 Related Work

The most comprehensive single dataset in the field of spell and grammar checking for Icelandic is the Icelandic Error Corpus (IEC, Arnardóttir et al. 2021) and its subsidiary corpus for errors made by L2 speakers (Glišić and Ingason, 2021). It uses a fine-grained error categorization system and has been used for training and evaluating spell and grammar checkers, specifically the rule-based GreynirCorrect (Óladóttir et al., 2022).

The Grammatical Error Correction Test Set (GECTS, Arnardóttir et al. 2024b), a hand-annotated dataset of Icelandic text with various spelling and grammatical errors, is annotated on the document level as opposed to the IEC, where each individual error is annotated. This, along with a more general error categorization system, makes it more suitable for evaluating recent sequence-to-sequence error correction models by testing the models' context awareness on larger texts.

3 Suggesting Standardized Orthography

We present the Icelandic Standardization Benchmark Set: Spelling and Punctuation (IceStaBS:SP), a dataset of text examples containing non-

standard orthography and their standardized counterparts. Each item in our set corresponds to an entry in the official spelling rules for Icelandic, published by the Icelandic Language Council and applied in Icelandic schools. Each item contains three examples of standardized text, along with thorough explanations of how and why each text has been altered. The dataset is meant to serve as a key component in the development of automatic spell checking in an educational setting, providing handcrafted explanations which can be expanded or used for instruction tuning.

Both the text examples showing non-standard orthography and the additional explanations are constructed and reviewed by the authors of this paper, all of whom have a background in Icelandic linguistics and one of whom is one of the authors of the most recent version of the official spelling rules. The text examples each show exactly one non-standard text feature in order to clearly demonstrate the applicable standardization and in order to check whether that feature has been correctly captured by a spell-checking system. Depending on the orthographic issue being demonstrated, the text examples range from very short and simple sentences to short paragraphs, e.g. to display the prescribed use of punctuation between whole sentences. They are mostly synthetic (and partly based on the examples included in the publication of the spelling rules themselves) but where possible, we have extracted real-world examples from the IEC using the error codes in that corpus. This authentic approach was, however, limited by the need to include only one example of non-standard orthography in each example, so some of those examples have been slightly altered.

The official spelling rules consist of 33 main chapters and numerous subchapters. Some subchapters, as well as all of chapters 30 and 33, are ignored in our set as they are not applicable in the context of automatic correction of spelling and grammar (e.g. some contain only general discussion of phenomena, rather than concrete examples). In other cases, subchapters had to be split into further subsections for our purposes as they dealt with multiple distinct features. These are marked with “(a)”, “(b)”, etc. in IceStaBS:SP. In this way, we define a total of 247 rules over 31 chapters.

For each of these 247 rules, our dataset contains

an entry labeled with a distinct number and consisting of the following parts:

- **Short suggestion:** A suggested format for displaying a correction made by a spell-checker, containing a brief summary of the applicable spelling rule.
- **Long suggestion:** A more detailed description of the applicable rule, complete with a URL to the relevant chapter of the official spelling rules (in a few cases, links to multiple rules are included).
- **Examples:** Three examples of a short text containing the relevant issue, which show a potential correction in a hypothetical spell-correction interface according to the ‘short suggestion’ format. The proposed changes are shown both in isolation and in the context of the whole text.²
- **Error Code:** The relevant error code in the IEC.
- **URL:** The URL of the relevant section of the official spelling rules.

To illustrate how this information is structured in the IceStaBS:SP dataset, the entry for rule 1.2.1 (a) is shown in Figure 1.

The aim of the suggestions and explanations in our set is to provide further assistance to potential future users of an automatic spellchecker, not least young people and second language learners of Icelandic. Therefore, we try to keep our explanations accessible to the average speaker, with as little linguistic terminology as possible (especially in the short suggestions).

To as great an extent as possible, we also try to include helpful generalizations in the short suggestion format as opposed to only word-specific corrections. An example would be *<villa> á lík-lega að vera með stórum staf, <leiðrétt>, þar sem það er örnefni* ‘<error> should likely be written with a capital initial letter, <correction>, as it is a place name’, rather than simply ‘<error> should likely be written with a capital initial letter’. This is sometimes made difficult, however, by rules that can apply to many different scenarios or are simply too complex to sum up in one short sentence.

²In a few cases, these entries will be identical. This is mostly in the case of punctuation, e.g. where rules on appropriate marking of a subclause need to take into account the whole sentence.

```

"1.2.1 (a)": {
  "short_suggestion": "<villa> á líklega að vera með stórum staf, <leiðrétt>, þar sem það kemur  
á eftir punkti.",
  "long_suggestion": "Stór stafur er alltaf ritaður í upphafi máls og í nýrri málsgrein  
á eftir punkti.  
Sjá ritreglu 1.2.1 (https://ritreglur.arnastofnun.is/#1.2.1).",
  "examples": {
    "1": {
      "original_sentence": "Afi og amma ætla að koma í heimsókn. þau koma bráðum.",
      "standardized_sentence": "Afi og amma ætla að koma í heimsókn. Þau koma bráðum.",
      "suggestion": "<þau> á líklega að vera með stórum staf, <Þau>, þar sem það kemur  
á eftir punkti.",
      "original_part": "þau",
      "standardized_part": "Þau"
    },
    "2": {
      "original_sentence": "Ráðgert er að nýtt hús rísi í vor. vinnan við það er þó ekki hafin.",
      "standardized_sentence": "Ráðgert er að nýtt hús rísi í vor. Vinnan við það er þó ekki hafin.",
      "suggestion": "<vinnan> á líklega að vera með stórum staf, <Vinnan>, þar sem það kemur  
á eftir punkti.",
      "original_part": "vinnan",
      "standardized_part": "Vinnan"
    },
    "3": {
      "original_sentence": "Margt skiptir máli þegar skáldsögur eru skrifaðar. málfar er t.d.  
mikilvægur þáttur.",
      "standardized_sentence": "Margt skiptir máli þegar skáldsögur eru skrifaðar. Málfar er t.d.  
mikilvægur þáttur.",
      "suggestion": "<málfar> á líklega að vera með stórum staf, <Málfar>, þar sem það kemur  
á eftir punkti.",
      "original_part": "málfar",
      "standardized_part": "Málfar"
    }
  },
  "error_code": "lower4upper-initial",
  "ritreglur_url": "https://ritreglur.arnastofnun.is/#/1.2.1 (a)"
}

```

Figure 1: JSON structure of the IceStaBS:SP dataset, showing the entry for rule 1.2.1 (a), which deals with capitalization after a full stop. The text in the ‘short suggestion’ slot says: ‘<error> should probably be capitalized, <correction>, as it follows a full stop.’ The text in the ‘long suggestion’ slot says: ‘A capital letter is always used at the start of a text and the beginning of a new sentence following a full stop. See spelling rule 1.2.1 [...]’ Examples 1–3 then show text in Icelandic where the start of a sentence has not been capitalized, with suggested corrections in the ‘suggestion’ slot presented according to the ‘short suggestion’ format.

4 Applying IceStaBS:SP in Evaluation

To gauge the efficacy of the IceStaBS:SP dataset as a benchmarking tool for orthography, we performed an evaluation experiment, where various spell and grammar checkers for Icelandic were applied on our data and then evaluated statistically. This serves two purposes.

Firstly, it allows us to evaluate the performance of these tools on a standardized dataset, which can be used to compare the tools to each other and, preferably, to other benchmark sets. Secondly, we standardize our methods for evaluating correction tools on our benchmark set. The source code of our evaluation methods is then made available on GitHub³ for others to use on new tools, as well as the output of the tools we use in our evaluation.

³<https://github.com/stofnun-arna-magnussonar/IceStabs-eval>

4.1 Tools Evaluated

We intend our dataset to be applicable to any tool which corrects errors in Icelandic text. With this in mind, we selected 10 tools and models to test. These include commercial and open-source software, with a broad range of effectiveness, from state-of-the-art to baseline tools.

Our first focus are tools which can be run programmatically. These were:

Byte-Level Neural Error Correction Model for Icelandic (Ingólfssdóttir et al., 2023): A fine-tuned ByT5-base Transformer designed for error correction in Icelandic text. It functions similarly to a machine translation model, converting erroneous Icelandic into correct Icelandic. We evaluate three versions of this tool, each representing a successive update: 22-09, 23-12, and 24-03.

GreynirCorrect (Óladóttir et al., 2022): A rule-

based spell and grammar checker for Icelandic. The tool is based on Greynir (Þorsteinsson et al., 2019), a syntactic parser for Icelandic. We evaluate the most recent version of this tool: version 4.0.0.⁴

Icelandic GPT-SW3 for Spell and Grammar Checking (Arnardóttir et al., 2024a): A GPT-SW3 (Ekgren et al., 2022, 2024) model, fine-tuned on Icelandic and particularly on the task of spell and grammar checking. The experimental setup we use is identical to the example given in the model’s HuggingFace repository.⁵

Skrambi: A closed-source rule-based spell checker for Icelandic.⁶

In addition to the tools which can be run programmatically, we evaluated four ‘manual’ tools, i.e., tools which are first and foremost accessible through an end-user platform of some kind. These were:

Hunspell: An open-source spell checker and morphological analyzer. The Icelandic language rules⁷ for Hunspell are accessible via LibreOffice, where Hunspell is the standard spell checker.

Google Docs Spelling and Grammar check: Built-in spell and grammar checker of Google Docs.⁸

Microsoft Editor: Built-in spell and grammar checker of Microsoft Word.⁸

Ritvilluvörnin Púki: Proprietary spell and grammar checker, specifically for Icelandic text.⁹

With this in mind, the total number of tools and individual correction models we evaluate is 10. Half of these (5) are developed by Miðeind,¹⁰ a private language technology company.

In our evaluation experiment, each tool is given a simplified label, which we will use to refer to them in the following sections. An alphabetic overview of these labels is as follows:

⁴<https://github.com/mideind/GreynirCorrect>

⁵<https://huggingface.co/mideind/icelandic-gpt-sw3-6.7b-gec/blob/main/handler.py>

⁶<https://skrambi.arnastofnun.is>

⁷<https://github.com/nifgraup/hunspell-is>

⁸Publically available versions as of October 27, 2024.

⁹<https://puki.is>

¹⁰<https://mideind.is>

1. ByT5 (22-09)
2. ByT5 (23-12)
3. ByT5 (24-03)
4. Google Docs
5. GreynirCorrect
6. Hunspell
7. Ice-GPT-SW3
8. MS Word
9. Púki
10. Skrambi

4.2 Evaluation Metrics

We define three main metrics which can be used to evaluate the performance of a spell and grammar checker on our dataset:

Sentence-level accuracy: Direct comparison between output sentences and standardized versions. A sentence is considered correct if the output is identical to the standardized sentence.

Token-level $F_{0.5}$ score: An F-score metric modified for spell and grammar correction. $F_{0.5}$ is a weighted average of precision and recall, where precision is given twice the weight of the recall. It is included in the ERRANT toolkit (Bryant et al., 2017) and was used in the CoNLL-2014 shared task (Ng et al., 2014).

GLEU score: A modified version of the BLEU score. BLEU is used to evaluate the quality of machine translation, while GLEU is used to evaluate the quality of spell and grammar correction. It is especially well suited for evaluating sequence-to-sequence models, as it does not rely on error categories for evaluation (Napoles et al., 2015, 2016).

As the tools we evaluate are technically and functionally diverse, it may be inferred that a given metric may suit one tool better than another. This is up to analysis, but in our overall evaluation structure, we use all three metrics to evaluate all tools.

5 Results

We evaluate the performance of the tools on the IceStaBS:SP dataset using the three metrics described above. The results are shown respectively in Figures 2, 3, and 4.

5.1 Performance Per Tool

The tool with both the highest proportion of correct sentences, as shown in Figure 2, and the highest $F_{0.5}$ score, as shown in Figure 3, is Miðeind’s ByT5 (23-12) with 46.42% sentence accuracy and a token-level $F_{0.5}$ score of 0.70.

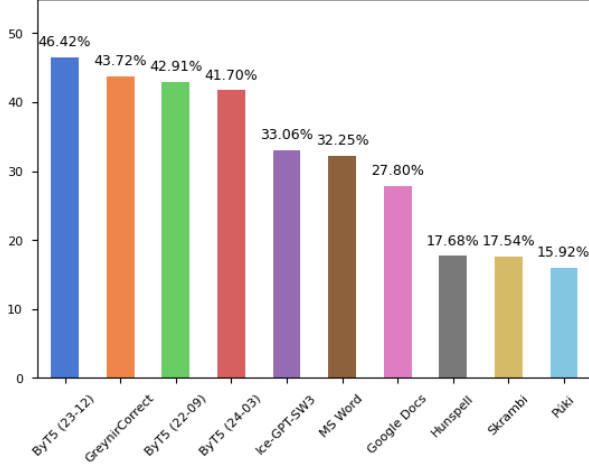


Figure 2: Sentence-level accuracy of the tools evaluated.

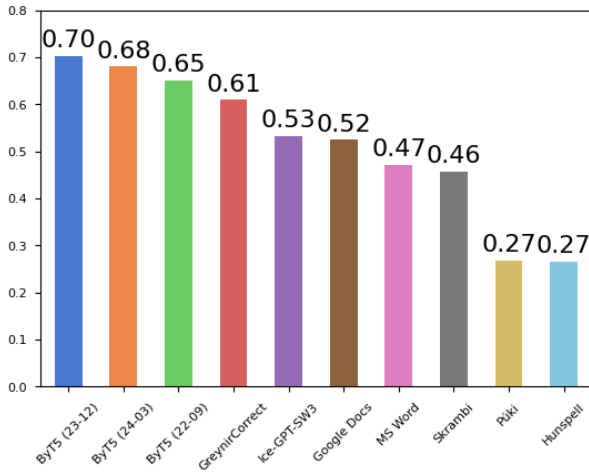


Figure 3: Token-level $F_{0.5}$ scores of the tools evaluated.

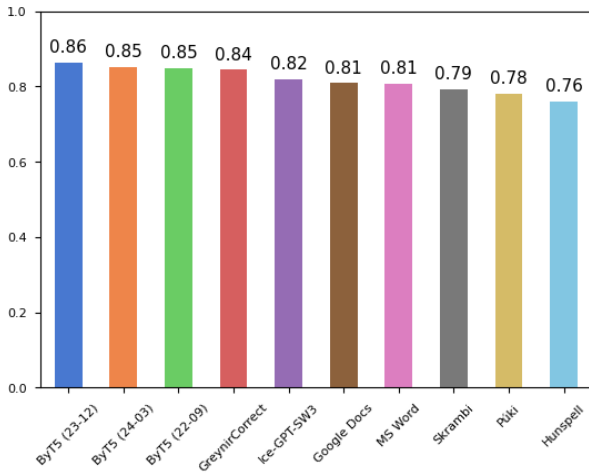


Figure 4: Dataset-level GLEU scores of the tools evaluated.

One possible limitation of the results described here is that not all spell-checking software has equal coverage when it comes to our predefined rule chapters. In the case of MS Word, Google Docs and Púki, various errors are not handled by the spell-checking features of the platform, but the respective autocorrect functionality of the platform. This particular issue is beyond the scope of our current evaluation but will hopefully be controlled for in a future iteration.

We see substantial variance between the highest scoring tools and the lowest. This is especially interesting when real-world integration and use are taken into account. Púki, the widely used spell-checking tool for Icelandic (originally released in 1987 and iterated upon since then), achieves the lowest scores on our sentence correctness and token-level $F_{0.5}$ score metrics.

The leaders of the evaluation metrics are the Miðeind ByT5 models, followed closely by GreynirCorrect. On the one hand, it is interesting that of the three ByT5 models, the newest iteration (24-03) underperforms compared to the previous one (23-12). On the other hand, all the (comparably lightweight) ByT5 models, along with the rule-based GreynirCorrect, outperform the compute-heavy Ice-GPT-SW3 model.

As shown in Figure 4, the three ByT5 models achieve the highest GLEU scores. It should be noted that of the tools we evaluate, the ByT5 (24-03) and Ice-GPT-SW3 models have previously published GLEU scores: ByT5 (24-03) is reported to achieve GLEU scores of 0.90 and 0.91 when evaluated on the GECTS and IEC datasets, respectively (Ingólfssdóttir et al., 2023). The GLEU score for the Ice-GPT-SW3 when evaluated on the GECTS is 0.93 (Arnardóttir et al., 2024a). These numbers are different from our results, which may reflect inherent differences in IceStaBS:SP, compared to the GECTS and IEC datasets. This is not totally unexpected, as the other two datasets are corpora, which IceStaBS:SP is not. Further analysis will shed light on these differences.

5.2 Performance Per Rule Chapter

As is to be expected, as the phenomena dealt with in some chapters are more common or straightforward than in others, there is considerable variance in tool performance across different chapters of the spelling rules. The highest scores recorded for each chapter are shown in Tables 1 and 2, in

terms of $F_{0.5}$ score and sentence accuracy, respectively. In some cases, the best-performing tools correct each example in a chapter exactly as intended. These include chapters 9 and 18, both of which have only three example texts in our dataset and deal with common and fairly straightforward issues (chapter 9 concerns words such as *hvar* (‘where’) that are spelt with *hv* and not *kv*, despite the *h* invariably being pronounced [k^h] and not [h] by most speakers, and chapter 18 deals with double consonant stems).

More pleasantly surprising is the excellent maximal performance achieved on chapters 6 (11 out of 12 examples correct), 16 (11 out of 12 correct) and especially 4 (21 out of 21 examples correct). The top-scoring model for the last of these chapters proved to be the rule-based and computationally light Skrambi, which overall placed 8th out of the ten models in terms of token-level $F_{0.5}$ score, behind all neural models. It is worth noting that chapter 4 concerns the spelling of vowels before the consonants clusters *ng* and *nk*, where letters that typically are used to represent monophthongs are pronounced as diphthongs (e.g. *banki* (‘bank’) instead of **bánki*, even though the relevant sound, [au], is almost always represented with *á* and not *a*) but the opposite can also occur without any cues in pronunciation in some exceptions (e.g. *jánka* (‘agree’), derived from *já* (‘yes’), or *rángirni* (‘greed’), a compound formed by *rán* (‘robbery’) and *girmi* (‘desire’)). This is an example of a scenario that seems to lend itself better to models that are rule-based or include hard-coded exceptions, as opposed to neural models which might possibly be thrown off the trail of the overarching rule by exceptions found in the training data.

On the other hand, for chapters 23 (which covers semicolons) and 27 (which covers parentheses and square brackets), not a single tool corrected a single example in accordance with the spelling rules. Both those rules fall under the punctuation part of the spelling rules, which somewhat predictably yields generally worse results than the spelling portion (chapters 1 through 20). After all, rules on punctuation often depend on some fairly abstract semantic features (e.g. from rule 23.1: ‘A semicolon represents a stronger break in a text than a comma but a lesser break than a full stop’) and deviations from the standard do not result in non-words, as deviations from spelling

rules might.

Ch.	Best Tool	$F_{0.5}$	No. Ex.
1	greynir	0.46	153
2	byt5-23-12	0.60	60
3	skrambi	0.83	12
4	skrambi	1	21
5	word	0.56	75
6	google	0.96	12
7	greynir	0.66	21
8	greynir	0.74	39
9	byt5-22-09	1	3
10	greynir	0.62	21
11	byt5-22-09	0.66	3
12	byt5-23-12	0.85	30
13	google	0.50	12
14	byt5-24-03	0.8	30
15	byt5-22-09	0.57	36
16	greynir	0.91	12
17	hunspell	0.67	9
18	byt5-22-09	1	3
19	google	0.74	21
20	hunspell	1	6
21	ice-gpt-sw3	0.28	42
22	byt5-23-12	0.54	24
23	None	0	3
24	byt5-24-03	0.16	6
25	byt5-22-09	0.66	3
26	ice-gpt-sw3	0.38	33
27	None	0	6
28	byt5-22-09	0.5	6
29	byt5-22-09	0.27	18
30	ice-gpt-sw3	0.17	9
31	byt5-23-12	0.33	12

Table 1: $F_{0.5}$ score Leaderboard for IceStabs:SP Evaluation, for each chapter in the spelling rules.

6 Limitations

There are various aspects of the IceStaBS:SP dataset that could be improved in future iterations. These range from superficial to inherent issues, the solutions to which will need further work and discussion.

As shown in Figure 5, even though the IceStabs:SP data is organized into 31 distinct chapters (reflecting the 33 chapters of the source material), the distribution of examples across these chapters is not uniform. This is due to the fact that some chapters cover more common and straightforward spelling rules, while others deal with more com-

Ch.	Best Tool	Score	Total	Ratio
1	byt5-23-12	68	153	44.44%
2	byt5-22-09	34	60	56.66%
3	skrambi	10	12	83.33%
4	skrambi	21	21	100%
5	word	40	75	53.33%
6	byt5-23-12	11	12	91.66%
7	greynir	14	21	66.66%
8	greynir	28	39	71.79%
9	byt5-22-09	3	3	100%
10	google	12	21	57.14%
11	byt5-22-09	2	3	66.66%
12	byt5-22-09	25	30	83.33%
13	greynir	6	12	50%
14	byt5-23-12	23	30	76.66%
15	byt5-22-09	20	36	55.55%
16	greynir	11	12	91.66%
17	hunspell	5	9	55.55%
18	byt5-22-09	3	3	100%
19	google	15	21	71.42%
20	hunspell	6	6	100%
21	ice-gpt-sw3	12	42	28.57%
22	byt5-23-12	13	24	54.16%
23	None	0	3	N/A
24	byt5-24-03	1	6	16.66%
25	byt5-22-09	2	3	66.66%
26	ice-gpt-sw3	12	33	36.36%
27	None	0	6	N/A
28	byt5-22-09	3	6	50%
29	byt5-22-09	5	18	27.77%
30	greynir	2	9	22.22%
31	byt5-23-12	4	12	33.33%

Table 2: Sentence-level Accuracy Leaderboard for IceStabs:SP Evaluation, for each chapter in the spelling rules.

plex, subjective or less frequent issues. As a result, the dataset contains a larger number of examples for the more common rules, which may skew the evaluation results towards these chapters. In short, not all chapters of the Icelandic spelling rules are created equal.

Chapters 1, 2, and 5 have significantly more entries than the other chapters in the dataset, with chapter 1 (use of upper and lower case letters at the beginnings of words) being particularly prominent. This discrepancy is due to the fact that these chapters cover fundamental and frequently encountered spelling rules in Icelandic orthography.

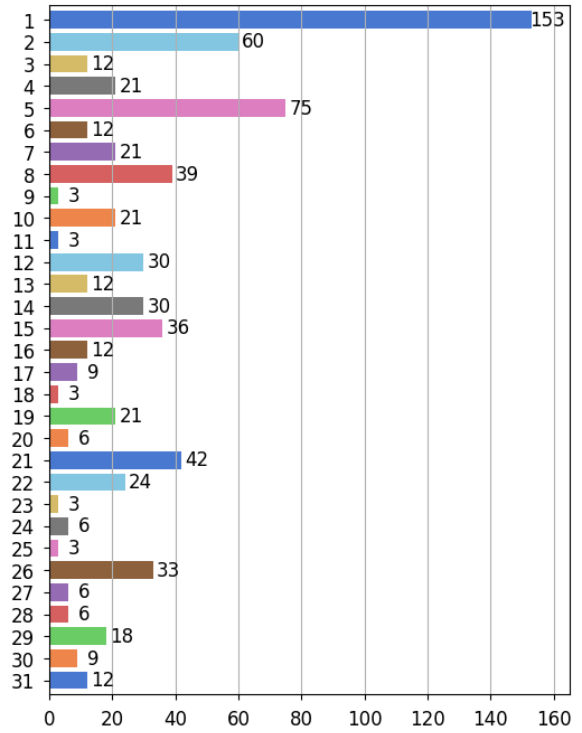


Figure 5: Number of example sentences per main chapter in the IceStaBS:SP dataset.

On the opposite end of this spectrum are chapters 9, 11, 18, 23 and 25, which all have a single rule entry each (giving 3 examples per chapter in the overview in Figure 5). Even though the dataset structure and evaluation procedure treats these chapters as equal to the others, they are not equal in terms of the number of examples.

Currently, the IceStaBS:SP dataset only allows for a single standardized suggestion for each example. This means that the dataset does not account for the possibility of multiple correct solutions to a given spelling or grammar issue. As there are sometimes more than one correct way to write something according to the spelling rules, some entries in our dataset *should* allow for multiple possible correct alterations. An example would be some non-standard way of writing a specific time, which could be corrected to e.g. ‘2.30’ or ‘2:30’ as both a full stop and a colon are possible ways of separating hours from minutes, according to spelling rules 22.5 and 29.5, respectively. Even though the number of these occurrences is low (variation is found in about 20 rules out of 247), this is a limitation that will be addressed in future iterations of the dataset.

7 Conclusions and Future Work

We have presented the IceStaBS:SP dataset, a comprehensive benchmark set for Icelandic spelling and punctuation. The dataset is based on the official spelling rules for Icelandic and provides standardized suggestions for a wide range of spelling and punctuation issues. As such, it is the first of its kind for Icelandic.

We have evaluated the performance of ten spell and grammar checkers on the dataset, using three main metrics: sentence-level accuracy, token-level $F_{0.5}$ score, and GLEU score. The results are broadly in line with expected performance, which is encouraging for the utility of the dataset as a benchmarking tool.

Further work is needed to address limitations in the dataset and explore additional evaluation metrics to provide a more comprehensive assessment of spell and grammar checkers for Icelandic.

Acknowledgments

We would like to thank the NoDaLiDa/Baltic-HLT 2025 organizers for the assistance and communication while working on this submission and three anonymous reviewers for helpful feedback. In addition to the authors of this paper, Finnur Ágúst Ingimundarson is one of the authors of the dataset described here. This work was financed by the Icelandic Ministry of Culture and Business Affairs as part of the Language Technology Programme for Icelandic.

References

- Bjarki Ármannsson, Hinrik Hafsteinsson, Jóhannes B. Sigtryggsson, Atli Jasonarson, Finnur Ágúst Ingimundarson, Einar Freyr Sigurðsson, and Steinþór Steingrímsson. 2024. Icelandic Standardization Benchmark Set: Spelling and Punctuation 24.10. CLARIN-IS.
- Þórunn Arnardóttir, Svanhvít Lilja Ingólfssdóttir, Garðar Ingvarsson Juto, Haukur Barri Símonarson, Hafsteinn Einarsson, Anton Karl Ingason, and Vilhjálmur Þorsteinsson. 2024a. Icelandic GPT-SW3 for spell and grammar checking. CLARIN-IS.
- Þórunn Arnardóttir, Svanhvít Lilja Ingólfssdóttir, Haukur Barri Símonarson, Hafsteinn Einarsson, Anton Karl Ingason, and Vilhjálmur Þorsteinsson. 2024b. Beyond error categories: A contextual approach of evaluating emerging spell and grammar checkers. In *Proceedings of the 3rd Annual Meeting of the Special Interest Group on Under-resourced Languages @ LREC-COLING 2024*, pages 45–52, Torino, Italia. ELRA and ICCL.
- Þórunn Arnardóttir, Xindan Xu, Dagbjört Guðmundsdóttir, Lilja Björk Stefánsdóttir, and Anton Karl Ingason. 2021. Creating an error corpus: Annotation and applicability. In *Proceedings of CLARIN Annual Conference*, pages 59–63. Virtual edition.
- Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. Automatic annotation and evaluation of error types for grammatical error correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 793–805, Vancouver, Canada. Association for Computational Linguistics.
- Ariel Ekgren, Amaru Cuba Gyllensten, Evangelia Gogoulou, Alice Heiman, Severine Verlinden, Joey Öhman, Fredrik Carlsson, and Magnus Sahlgren. 2022. Lessons learned from GPT-SW3: Building the first large-scale generative language model for Swedish. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3509–3518, Marseille, France. European Language Resources Association.
- Ariel Ekgren, Amaru Cuba Gyllensten, Felix Stollenwerk, Joey Öhman, Tim Isbister, Evangelia Gogoulou, Fredrik Carlsson, Alice Heiman, Judit Casademont, and Magnus Sahlgren. 2024. GPT-SW3: An autoregressive language model for the Nordic languages.
- Isidora Glišić and Anton Karl Ingason. 2021. The nature of Icelandic as a second language: An insight from the learner error corpus for Icelandic. In *CLARIN Annual Conference*, pages 26–30. Virtual edition.
- Svanhvít Lilja Ingólfssdóttir, Pétur Orri Ragnarsson, Haukur Páll Jónsson, Haukur Barri Símonarson, Vilhjálmur Þorsteinsson, and Vésteinn Snæbjarnarson. 2023. Byte-level grammatical error correction using synthetic and curated corpora. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7299–7316, Toronto, Canada. Association for Computational Linguistics.
- Courtney Naples, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2015. Ground truth for grammatical error correction metrics. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 588–593, Beijing, China. Association for Computational Linguistics.
- Courtney Naples, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2016. GLEU without tuning.
- Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. 2014. The CoNLL-2014 shared task

on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14, Baltimore, Maryland. Association for Computational Linguistics.

Hulda Óladóttir, Þórunn Arnardóttir, Anton Karl Ingason, and Vilhjálmur Þorsteinsson. 2022. Developing a spell and grammar checker for Icelandic using an error corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4644–4653, Marseille, France. European Language Resources Association.

Vilhjálmur Þorsteinsson, Hulda Óladóttir, and Hrafn Loftsson. 2019. A wide-coverage context-free grammar for Icelandic and an accompanying parsing system. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1397–1404, Varna, Bulgaria. INCOMA Ltd.

An Icelandic Linguistic Benchmark for Large Language Models

Bjarki Ármannsson, Finnur Ágúst Ingimundarson, Einar Freyr Sigurðsson

The Árni Magnússon Institute for Icelandic Studies, Iceland

`bjarki.armannsson@arnastofnun.is`

`fai@hi.is`

`einarr.freyr.sigurdsson@arnastofnun.is`

Abstract

This paper introduces a linguistic benchmark for Icelandic-language LLMs, the first of its kind manually constructed by native speakers. We report on the scores obtained by current state-of-the-art models, which indicate room for improvement, and discuss the theoretical problems involved in creating such a benchmark and scoring a model’s performance.

1 Introduction

Large Language Models (LLMs) have in the last few years become near ubiquitous in the field of Language Technology (LT) and in their wake follows a growing need to test their capabilities on all kinds of tasks, such as language understanding and generation, mathematics, programming etc. As English is the dominant language in the field and the biggest source of training data for these models, it is only natural that the principal benchmarks for the models (translations aside) also focus on English. However, it is vital to also evaluate the capabilities of the models for lower-resource languages.

We introduce a standard benchmarking dataset (Ármannsson et al., 2024) to evaluate LLMs’ grammatical ‘knowledge’ and linguistic accuracy for Icelandic, a lower-resource language. Such benchmarks can help LLM developers to improve their models’ Icelandic proficiency in a measurable way and provide researchers with further insight into these models’ output patterns, limitations and unexpected ‘behaviour’. As far as the authors are aware, this is the first benchmark of its kind specifically constructed for Icelandic by native speakers and experts in linguistics and LT (see Section 2).

Although the models’ capabilities in Icelandic are under scrutiny, we use English for all of

our prompts in order to facilitate future cross-linguistic research. As one reviewer points out, it might be interesting to contrast these results with the same prompts in Icelandic, but we leave that for future work. We do not test for proficiency in standard vs. non-standard Icelandic, for instance the widespread use of dative instead of the standard accusative as the subject case of various psych verbs, like *langa* ‘want’ or *vanta* ‘lack, need’, i.e. *mér* [dat.] *langar* instead of *mig* [acc.] *langar* ‘I want’. We rather aim to focus on features which should be unanimously agreed to be grammatical or ungrammatical by native speakers of Icelandic.¹

The published benchmark set contains 1160 hand-written items over 19 subcategories of syntax, morphology and semantics, tested with five different methods (see Table 1). We also include a small set of 102 translation tasks to test a model’s language understanding and grammatical capabilities in producing Icelandic text.

2 Related Work

In constructing our dataset, we partly look to similar linguistic benchmarks for LLMs that have been constructed for English. Warstadt et al. (2020)’s Benchmark of Linguistic Minimal Pairs for English (BLiMP) is perhaps the most commonly cited example. It is based around 67,000 minimal pairs, where one example is considered grammatical and the other ungrammatical, and models are tasked with ‘judging’ the grammatical acceptability of each sentence. (As this was before instruction-tuned models like ChatGPT-3 and the tendency towards closed black-box models,

¹A comparison study of native human speakers, in order to confirm or challenge some of the assumptions made in the construction of this set, is currently a work in progress. Initial results, focusing only on gender agreement, indicate effectively unanimous native speaker preference for the correct answers in this benchmark and rejection of the incorrect ones.

Method	Category	No. of items
Sentence grammaticality check (yes/no)*	Simple bad/good sentences	40
	Attributive agreement	88
	Predicate agreement	28
	Word order	28
	Verb agreement	28
	Subject case	28
	Island effect sentences	80
	wh-movement	20
	Topicalization	32
	Gapping	120
	Reflexivization	40
Well-formedness check of compound nouns (yes/no)*	Word formation	280
Fill-in-the-blank	Anaphoric reference	20
	Coreference resolution	44
	Wug test (past tense of verbs)	20
Fragment answering	Fragment answers	40
Question answering	Coreference resolution	44
	Attributive agreement	30
	Word sense disambiguation	150
Total		1160

Table 1: The breakdown of items in our main benchmark set. All items were created manually. For the top two method types, marked with an asterisk, we also double the number of items in order to ask the inverse question, i.e. “Is this sentence grammatically **incorrect** (vs. **correct**)?”. For the word sense disambiguation task, we consider pairs of sentences that contain the same lexical form and we double the number of items to ask the same question with the order of the sentence pairs reversed.

the authors simply compared the log probabilities a model assigned to sentences, i.e. making it easy to contrast how likely input sentence A was compared to input sentence B for a given model.) This general blueprint for constructing linguistic benchmarks for LLMs has been widely followed, for instance by the makers of the Zorro test suite (Huebner et al., 2021) and the ScaLa linguistic acceptability dataset for Scandinavian languages (including Icelandic) (Nielsen, 2023).

These test sets all use automatically constructed examples, which makes it possible for the BLiMP dataset, for example, to have 1,000 sentence pairs for each of the 67 grammatical tasks tested. In terms of size, our benchmark certainly pales in comparison. On the other hand, it is possible for a human to have an overview of it, whereas BLiMP is simply too large and lower-quality pairs get lost in the masses (see Vázquez Martínez et al. (2023) for more detailed criticism). In this case, we find our approach preferable, but we are also aware of its drawbacks (see Bowman and Dahl (2021) for arguments that “expert authorship” can be counterproductive, when researchers have direct, fine-grained control over the data, as it may intentionally or unintentionally lead to data “that is oriented toward linguistic phenomena that are widely studied and widely known to be important to the task at hand”).

As far as interesting theoretical work on the linguistic capabilities and limitations of LLMs is

concerned, there has been an ongoing and interesting debate between researchers that have used two different approaches to evaluate models in this regard. One group is represented by Dentella et al. (2023), who use acceptability judgments, widely used in traditional linguistic research, that are elicited with prompts. The other group is represented by Hu and Levy (2023), who argue that prompting is not a substitute for probability measurements in LLMs and that such metalinguistic judgments of acceptability presuppose a model’s understanding of grammatical acceptability. Their approach is to compare the log probabilities of a model’s output on the grounds that this gives a better idea of that model’s “linguistic generalization”. As much as we would have liked to imitate this approach, it was simply not possible in our one-size-fits-all setup, as closed models such as the ones provided by OpenAI and Anthropic offer limited or no access to their log probabilities.²

We take some inspiration from Weissweiler et al. (2023), who test the morphological capabilities of ChatGPT via a ‘Wug test’, where a model is tasked with forming words from non-sense root forms. We also build on the work of Sigurðsson and Nowenstein (2023), who test

²At testing time, OpenAI only provided the option of retrieving the top 5 ‘logprobs’ for a model’s output, i.e. the top 5 most likely tokens, which we tested in a follow-up work to this benchmark along with input log probabilities for models where those probabilities were available (work in progress).

the capabilities of GPT-4 in Icelandic, partly using methods we include in our benchmark set. Lastly, the Icelandic LT company Miðeind maintains an LLM leaderboard on HuggingFace, where a selection of LLMs are evaluated across six tasks for Icelandic: a reduced Icelandic version of Winogrande, grammatical error detection, inflection, Belebele (multiple-choice reading comprehension), machine-translated ARC-Challenge (multiple-choice question answering) and an Icelandic WikiQA dataset.³

3 Benchmark Composition

The benchmark was created by the authors of this paper, who have an academic background in the study of Icelandic, theoretical linguistics and LT. As already mentioned in Section 2, the point of departure were similar linguistic benchmarks for English, but we were also inspired by previous work and studies on Icelandic grammar; we point out some references below, where applicable. Some of the tasks can be applicable in a multitude of languages (such as the sentence grammaticality check), whereas others are more specific to Icelandic and languages that have more complex morphology and a richer inflectional system than, for instance, English (word formation, fill-in-the-blank and fragment-answering). See Appendix A for examples of each task.

3.1 Sentence Grammaticality Tasks

We test for acceptability of different syntactic violations, many of which are tested in similar benchmarks for English. We do this by using grammaticality judgments and prompts of the form: “Is the following Icelandic sentence grammatically correct in Icelandic? *<Example sentence in Icelandic.>* Answer only with one word, yes or no.” Others, such as violations of gender agreement, are more tailored towards Icelandic grammar. To try to control for possible yes/no biases, we ask the inverse question (“[...] **incorrect** [...]”) for each item. Grammaticality judgments have frequently been used in Icelandic syntax research – see, e.g., Þráinsson et al. (2013).

3.2 Word-Formation Tasks

Similar to the sentence grammaticality tasks described in Section 3.1, we ask about the well-

formedness of compounds in which the first noun has one of three suffixes, *-un*, *-ing* or *-uð*, all of which are used in the genitive when they are part of the first noun in a compound: “Is the following compound word in Icelandic well-formed? *<compound>* Answer only with one word, yes or no.” As with the task in Section 3.1, we ask an inverse question, trying to control for yes/no biases. For further reading on compounding in Icelandic, see, e.g., Jónsson (1984), Rögnvaldsson (1990) Bjarnadóttir (2005) and Harðarson (2016).

3.3 Fill-in-the-Blank Tasks

We include three different fill-in-the-blank tasks. One tests an LLM’s ability in anaphoric reference: “Fill in the blank in the following Icelandic sentence with the correct pronoun: *<Sentence with a blank.>* Answer only with one pronoun in Icelandic.” Another task looks at coreference resolution in which the context names two individuals. The continuation of each sentence contains a blank that refers to one of these individuals. The third task tests the past-tense inflection of made-up weak verbs in a Wug test (cf. the classic study by Berko 1958) – for recent studies using Wug tests with native speakers of Icelandic, see Björnsdóttir (2023) and Nowenstein (2023).

3.4 Fragment-Answering Tasks

The question *Who took my car?* does not require a whole sentence as a reply as we could answer it with, e.g. a single name, such as *Ann*. This is a fragment answer. The benchmark contains 40 wh-questions with context where the task is to give a single-word answer: “Here is an Icelandic sentence, followed by a question: *<Context sentence.> <Question that refers to the context.>* Answer the question with only one word in Icelandic.” This task partly builds on previous work on fragment-answering in native speakers of Icelandic (e.g. Sigurðsson and Stefánsdóttir 2014, Sigurjónsdóttir and Nowenstein 2016 and Örnólfsdóttir 2017).

3.5 Question-Answering Tasks

The question-answering part includes direct questions on coreference resolution (“Which name does the pronoun *<pronoun>* refer to in the following Icelandic sentence [...]”), attributive agreement (“Which of the slash-separated options in the following question forms part of a sentence

³<https://huggingface.co/spaces/mideind/icelandic-llm-leaderboard>

Provider	Model	Score (%)
Anthropic	claude-3-5-sonnet-20240620	77.24
Anthropic	claude-3-opus-20240229	71.90
OpenAI	gpt-4o-2024-08-06	72.59
OpenAI	gpt-4-turbo	62.33
OpenAI	gpt-4-0613	63.28
OpenAI	gpt-4o-mini-2024-07-18	66.21
Meta	Meta-Llama-3.1-70B-Instruct	61.21
Meta	Meta-Llama-3.1-405B-Instruct	66.47
Google	gemma-2-27b-it	59.57
Mistral AI	Mixtral-8x22B-Instruct-v0.1	48.71
Qwen	Qwen2-72B-Instruct	55.34
AI-Sweden	gpt-sw3-20b-instruct	46.12
AI-Sweden	gpt-sw3-20b-instruct-4bit-gptq	43.02

Table 2: Models tested and their overall scores.

that is grammatical in Icelandic [...]”) and word-sense disambiguation (“Does the word tagged with <i></i> in the following two Icelandic sentences have the same meaning [...]”).

3.6 Translation Tasks

In addition to our main benchmarking set, we also include a set of 102 translation-based tasks, which contains both Icelandic sentences that should be translated into English and vice versa. These tasks are based on the assumptions that: a) Both current and future state-of-the-art models for Icelandic will be primarily trained on English text; and b) A fair way to test understanding of some feature of natural language is to ask the party in question to rephrase it in another language in which they are fluent. Our translation tasks are, as far as we are aware, a novel method of assessing the linguistic capabilities of LLMs (although similar to linguistically-oriented test suites for benchmarking machine translation systems, see e.g. Mackentanz et al. 2022).

For the translation from Icelandic to English, we use new garden path sentences, which can be used to check whether a model has successfully parsed the sentence or not. For example, for the sentence *Birta Líf og Heimir niðurstöðurnar í næstu viku?* (‘Will Líf and Heimir publish the results next week?’), the word *birta* needs to be read as a verb meaning ‘publish’ and not as the woman’s name *Birta* in order for a reader to comprehend the sentence. If the name *Birta* appears in the English translation, we argue the model has not successfully parsed the sentence.

For translation from English to Icelandic, we include sentences that test: a) Gender agreement in the target output (e.g. for the source sentence *María is a good driver*, the translation

of *good* should agree with the masculine *bílstjóri* (‘driver’), rather than the feminine *María* in order to form a grammatical sentence), and b) Anaphoric reference in the target output (e.g. for the source sentence *The child poured the milk into the cup and checked to see whether it had gone sour*, the pronoun *it* should be translated in the feminine, *hún*, to refer to the milk rather than the cup, *bolli*, which is a masculine noun in Icelandic). As far as the gender agreement is concerned, all sentences have the same structure as the example above (i.e. <name> is a <adjective> <noun>) and we try throughout to select adjectives and nouns that should ideally only have one straightforward translation.

We emphasize that these tasks are not meant as machine translation test sets but can serve as an indicator of a model’s NLU performance and grammatical capabilities in producing Icelandic text. The output needs to be manually examined, as we do not include scripts for automatic evaluation, which is why we keep these two tasks separate from the other tasks in our main benchmark. We show the results of an automatic evaluation in Section 4.2.

4 Current Model Performance

4.1 Main Benchmark Set

We show the results on our benchmark set for thirteen currently available LLMs to give an idea of the state of the art for Icelandic.⁴ The models we tested are shown along with overall scores in Table 2; we show a further breakdown of scores across individual tasks in Appendix B. Anthropic and OpenAI models were accessed through their respective APIs; the Meta, Google, Mistral and Qwen models were all accessed through Together AI’s API. We ran the quantized version of AI-Sweden’s GPT-SW3 model locally and the non-quantized variant through a dedicated HuggingFace endpoint.⁵ For the API requests, we used default settings with two exceptions, setting the temperature to 0 and restricting maximum output tokens to 5 to try and keep the models’ output deterministic and brief.

⁴The models were chosen based on their standing according to the Icelandic LLM Leaderboard hosted by Miðeind and with the aim of including models from different providers.

⁵All tests were run on the 10th and 11th of October 2024, except the two models from AI-Sweden which were tested on the 10th and 13th of January 2025.

Category	Claude-3-5-Sonnet	GPT 4-o
Garden path	51.7	56.7
Agreement	68.2	63.6
Anaphora	100.0	95.0
Total score	64.7	65.7

Table 3: The scores on our set of translation tasks.

The top three scorers overall, and the only models that record over 70% accuracy, are Claude-3-5-Sonnet, GPT 4-o and Claude-3-Opus. Other models record between 43.02% and 66.47% accuracy, indicating considerable room for improvement for Icelandic-language LLMs. The scores vary considerably, however, between different tasks, as seen in Appendix B.

When scoring the outputs, we directly compare the answers obtained from the models with our reference answer but remove additional periods, spaces and the like from correct answers. It could therefore be argued that the scores we present show the models’ performance in too favourable a light (see discussion in Section 5). On the other hand, for some tasks it could have been possible to mark a greater variety of answers as correct than we presently do. This is the case for coreference resolution via the ‘Question-answering’ method, where the models are prompted to name the noun to which a particular pronoun refers. Accounting for the complexities provided by the Icelandic case system, we consider both the particular morphological form used in the example sentence *and* the nominative form of the word (in those cases where those two forms are different) to be correct.

4.2 Translation Task Subset

As previously stated, our set of translation tasks calls for manual evaluation of a model’s output. We therefore decide to score and show the results for only two models. We choose the two highest-scoring models according to our results in 4.1 (which gives us one model from each of the two best-performing ‘families’ of models, Anthropic’s Claude and OpenAI’s GPT). As seen in Table 3, the models achieve very similar scores overall, 65.7% for GPT 4-o and 64.7% for Claude-3-5-Sonnet. Both the garden path sentences and gender agreement tasks seem to present a challenge for these models but the anaphora resolution tasks are near-maximum for both.

5 Limitations

The limitations are a few. Firstly, we tried to find a suitable base prompt for each task that would be understood in the same way by different models. Even though we feel that the uniformness of the resulting answers reflects that we succeeded in this respect, we cannot be sure that some “fine-tuning” of the prompts would not have yielded better results.

Secondly, although we tried to include clear instructions in English in the prompts on what the output should be, such as “answer only with yes or no”, there were some deviations in the answers. These include correct answers in Icelandic, correct answers with an additional tail (e.g. “Yes. The correct sentence”), answers that include a full stop or other additional punctuation etc. To reduce these deviations, we cleaned the model answers for scoring. A correct answer in Icelandic, for instance, was therefore considered correct, as well as answers with a “tail” etc. Even though, as one reviewer points out, post-processing methods are fairly common practice and often used by LLM evaluation frameworks such as Gao et al. (2024), for human-alignment comparisons in LLMs, such lenience has been criticized (Leivada et al., 2024), on the grounds that a human would hardly respond in such a way. We acknowledge this, but would again like to stress that this could perhaps have been avoided with more precise prompts.

It remains an open question how best to score output. In our setup, a model’s answer that matches our reference answer gets one point. An answer that does not, gets none. It could be argued that this method does not highlight the differences in performance between different models sufficiently, as two models both get the same score for a wrong answer if one outputs a single pronoun in Icelandic as prompted and the other outputs gibberish. In this regard, our benchmark perhaps is better suited to measure the differences of better-performing models than capturing the differences between lesser models. We would like to encourage the further development of open-source models, which may require an evaluation that can provide information on when one of two wrong answers is more promising than another. Focusing on open-source models would also allow one to compare model output with input log probabilities of the test examples, following the work of Hu and Levy (2023). On the other hand, a more for-

giving scoring metric, based on e.g. Levenshtein distance, would simply not be applicable for our benchmark as the difference between a correct answer and an ungrammatical one is often only one or two letters.

6 Conclusions

We have presented a standard benchmarking dataset for evaluating the linguistic capabilities of LLMs for Icelandic, the first of its kind. We publish the dataset openly and describe its construction in order to hopefully aid further work in this respect for both Icelandic and Nordic NLP in a wider sense. In order to show the current state of the art for Icelandic, we show the results on our set for a variety of currently available models, which indicate considerable room for improvement for some of the tested phenomena. We also discuss some of the still-open questions regarding the best methods for testing the language capabilities of LLMs.

Acknowledgments

We would like to thank the NoDaLiDa/Baltic-HLT 2025 organizers for the assistance and communication while working on this submission and three anonymous reviewers for helpful feedback. We would also like to thank Jennifer Hu, Iris Edda Nowenstein and Þórdís Úlfarsdóttir for helpful input and feedback during the construction of our benchmark set. This work was financed by the Icelandic Ministry of Culture and Business Affairs as part of the Language Technology Programme for Icelandic.

References

Bjarki Ármannsson, Finnur Ágúst Ingimundarson, and Einar Freyr Sigurðsson. 2024. Icelandic Linguistic Benchmark for LLMs 24.10. CLARIN-IS.

Jean Berko. 1958. The child’s learning of English morphology. *WORD*, 14:150–177.

Kristín Bjarnadóttir. 2005. *Afleiðsla og samsetning í generatífri málfræði og greining á íslenskum gögnum*. Orðabók Háskólans, Reykjavík.

Sigríður Mjöll Björnsdóttir. 2023. Predicting ineffability: Grammatical gender and noun pluralization in Icelandic. *Glossa: a journal of general linguistics*, 8(1).

Samuel R. Bowman and George Dahl. 2021. What will it take to fix benchmarking in natural language understanding? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4843–4855, Online. Association for Computational Linguistics.

Vittoria Dentella, Fritz Günther, and Evelina Leivada. 2023. Systematic testing of three language models reveals low language accuracy, absence of response stability, and a yes-response bias. *Proceedings of the National Academy of Sciences*, 120(51):e2309583120.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac’h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024. A framework for few-shot language model evaluation.

Gísli Rúnar Harðarson. 2016. Peeling away the layers of the onion: on layers, inflection and domains in Icelandic compounds. *Journal of Comparative Germanic Linguistics*, 19:1–47.

Jennifer Hu and Roger Levy. 2023. Prompting is not a substitute for probability measurements in large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5040–5060, Singapore. Association for Computational Linguistics.

Philip A. Huebner, Elior Sulem, Fisher Cynthia, and Dan Roth. 2021. BabyBERTa: Learning more grammar with small-scale child-directed language. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 624–646, Online. Association for Computational Linguistics.

Baldur Jónsson. 1984. Samsett nafnorð með samsetta liði. Fáeinir athuganir. In Bernt Fossetøl, Kjell Ivar Vannebo, Kjell Venås, and Finn-Erik Vinje, editors, *Festskrift til Einar Lundeby 3. oktober 1984*, pages 158–174. Novus, Oslo.

Evelina Leivada, Vittoria Dentella, and Fritz Günther. 2024. Evaluating the language abilities of Large Language Models vs. humans: Three caveats. *Biolinguistics*, 18:Article e14391.

Vivien Macketanz, Eleftherios Avramidis, Aljoscha Burchardt, He Wang, Renlong Ai, Shushen Manakhimova, Ursula Strohmriegel, Sebastian Möller, and Hans Uszkoreit. 2022. A linguistically motivated test suite to semi-automatically evaluate German–English machine translation output. In *Proceedings of the Language Resources and Evaluation Conference*, pages 936–947, Marseille, France. European Language Resources Association.

- Dan Nielsen. 2023. ScandEval: A benchmark for Scandinavian natural language processing. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 185–201, Tórshavn, Faroe Islands. University of Tartu Library.
- Iris Edda Nowenstein. 2023. *Building yourself a variable case system: The acquisition of Icelandic datives*. Doctoral dissertation, University of Iceland.
- Eiríkur Rögnvaldsson. 1990. *Íslensk orðhlutafræði*, fourth edition. Málvísindastofnun Háskóla Íslands, Reykjavík.
- Einar Freyr Sigurðsson and Brynhildur Stefánsdóttir. 2014. ‘By’-phrases in the Icelandic New Impersonal Passive. *University of Pennsylvania Working Papers in Linguistics*, 20(1):311–320.
- Sigríður Sigurjónsdóttir and Iris Nowenstein. 2016. Passives and the “New Impersonal” construction in Icelandic language acquisition. In *Proceedings of the 6th Conference on Generative Approaches to Language Acquisition North America (GALANA 2015)*, pages 110–121, Somerville, MA. Cascadilla Proceedings Project.
- Einar Freyr Sigurðsson and Iris Edda Nowenstein. 2023. Nýjasta tækni og málvísindi. *Málfregnir*, 32:28–37.
- Héctor Vázquez Martínez, Annika Lea Heuser, Charles Yang, and Jordan Kodner. 2023. Evaluating neural language models as cognitive models of language acquisition. In *Proceedings of the 1st Gen-Bench Workshop on (Benchmarking) Generalisation in NLP*, pages 48–64, Singapore. Association for Computational Linguistics.
- Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The Benchmark of Linguistic Minimal Pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.
- Leonie Weissweiler, Valentin Hofmann, Anjali Kantharuban, Anna Cai, Ritam Dutt, Amey Hengle, Anubha Kabra, Atharva Kulkarni, Abhishek Vijayakumar, Haofei Yu, Hinrich Schuetze, Kemal Oflazer, and David Mortensen. 2023. Counting the bugs in ChatGPT’s wugs: A multilingual investigation into the morphological capabilities of a large language model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6508–6524, Singapore. Association for Computational Linguistics.
- Höskuldur Þráinsson, Ásgrímur Angantýsson, and Einar Freyr Sigurðsson, editors. 2013. *Tilbrigði í íslenskri setningagerð I. Markmið, aðferðir og efniviður*. Málvísindastofnun Háskóla Íslands, Reykjavík.
- Þórgunnur Anna Örnólfsdóttir. 2017. Hverjum þolmyndin glymur. Umfjöllun um af-liði í nýju setningagerðinni og hefðbundinni þolmynd án nafnliðarfærslu. B.A. thesis, University of Iceland, Reykjavík.

A Main Benchmark Set Task Examples

A.1 Sentence Grammaticality Tasks

All prompts in this section are of the form: “Is the following Icelandic sentence grammatically correct in Icelandic? <Example sentence in Icelandic.> Answer only with one word, yes or no.” Below we show examples for each category in the sentence grammaticality tasks accompanied by English glosses.

A.1.1 Simple Unambiguously Grammatical/Ungrammatical Sentences

(1) A simple ungrammatical sentence

Blístrum þið of mjög?
whisper.1PL you.2PL too very

(2) A simple grammatical sentence

Sólin skín.
sun-the shines

A.1.2 Attributive Agreement

(3) Violation

María er góð
María(female-name) is good.FEM
bílstjóri.
driver.MASC

(4) Correct version

María er góður
María(female-name) is good.MASC
bílstjóri.
driver.MASC

A.1.3 Predicate Agreement

(5) Violation

Þessar kvikmyndir eru mjög
these.FEM.PL films.FEM.PL are very
skemmtileg.
fun.FEM.SG/NEUT.PL

(6) Correct version

Þessar kvikmyndir eru mjög
these.FEM.PL films.FEM.PL are very
skemmtilegar.
fun.FEM.PL

A.1.4 Word Order

(7) Violation

Við ekki sáu þau í garðinum.
we not saw them in garden-the

- (8) **Correct version**
Við sáum þau ekki í garðinum.
we saw them not in garden-the

A.1.5 Verb Agreement

- (9) **Violation**
Af hverju fór þú ekki
for what went.1SG/3SG you.2SG not
heim?
home?
- (10) **Correct version**
Af hverju fórst þú ekki heim?
for what went.2SG you.2SG not home?

A.1.6 Subject Case

- (11) **Violation**
Alexanders Daníels langar oft í
Alexander.GEN Daníel.GEN wants often in
bíó um helgar.
cinema on weekends
- (12) **Correct version**
Alexander
Alexander.NOM/ACC/DAT
Daníel langar oft í bíó
Daníel.NOM/ACC/DAT wants often in cinema
um helgar.
on weekends

A.1.7 Islands

- (13) **Violation**
Hvaða próf gefur kennarinn Evu góða
what exam gives teacher-the Eva good
einkunn ef hún tekur?
grade if she takes
- (14) **Correct version**
Hvaða próf óttast kennarinn að Eva taki
what exam fears teacher-the that Eva takes
ekki?
not

A.1.8 Wh-movement

- (15) **Violation**
Hvern taldir þú rétt að gefa hærri
who.ACC thought you right to give higher
laun?
salary
- (16) **Correct version**
Hverjum taldir þú rétt að gefa hærri
who.DAT thought you right to give higher
laun?
salary

A.1.9 Topicalization

- (17) **Violation**
Þessari bók gætir þú lesið.
this.DAT book could you read

- (18) **Correct version**
Þessa bók gætir þú lesið.
this.ACC book could you read

A.1.10 Gapping

- (19) **Violation**
Þú borðaðir kökuna og ég
you ate cake-the.ACC and I
kleinuhringurinn
donut-the.NOM
- (20) **Correct version**
Þú borðaðir kökuna og ég
you ate cake-the.ACC and I
kleinuhringinn.
donut-the.ACC

A.1.11 Reflexivization

- (21) **Violation**
Hún vonar að ég flýti sér.
she hopes that I hurry REFL.DAT
- (22) **Correct version**
Ég vona að hún flýti sér.
I hope that she hurries REFL.DAT

A.2 Word-Formation Tasks

All prompts in this section are of the form: “Is the following compound word in Icelandic well-formed? *<compound>*. Answer only with one word, yes or no.” The first part of each compound is a noun ending in *-un*, *-ing* or *-uð*, all of which are used in the genitive when they are part of the first noun in a compound.

- (23) **Violation**
Sýkingþreyta.
infection.NOM-fatigue
- (24) **Correct version**
Sýkingarþreyta.
infection.GEN-fatigue

A.3 Fill-in-the-blank Tasks

The prompts for the anaphoric reference task in this section are of the form “Fill in the blank in the following Icelandic sentence with the correct pronoun: *<Example sentence containing a blank>*. Answer only with one pronoun in Icelandic.” The same prompt is used for the coreference resolution task, except the models are prompted for a name or noun instead of a pronoun. The prompts used for the Wug tests were as follows: “Fill in the blank in the following Icelandic sentence with the correct past tense of the verb tagged with *<i></i>*: *<Example text showing a verb in the infinitive, tagged as stated, and then a blank to be filled with the past tense of the verb>*. Answer only with one word.”

A.3.1 Anaphoric Reference

- (25) Hún ætlaði að telja fuglana í
she meant to count birds-the.MASC in
tjörnunum en _ voru á flugi.
ponds-the.FEM but _ were in flight

Incorrect answer

Þær.
they.FEM

Correct answer

Þeir.
they.MASC

A.3.2 Coreference Resolution

- (26) Lína ætlaði að sópa kjallarann með
Lína meant to sweep basement-the with
kústi en _ var ekki á sínum stað.
broom-the but _ was not in its place

Incorrect answer

Kjallarinn.
basement-the

Correct answer

Kústurinn.
broom-the

A.3.3 Wug Verbs

- (27) Okkur langaði að <i>krata</i> fiskinn
we wanted to <i>krata</i> fish-the
örlítið, þannig að við _ hann áður en
little so that we _ it before than
hann fór í ofninn.
it went in oven-the

Correct answer

Krötuðum.
'krated'. 1 PL

A.4 Fragment-Answering Tasks

All prompts in this section are of the form: "Here is an Icelandic sentence, followed by a question: <Context sentence.> <Question that refers to the context.> Answer the question with only one word in Icelandic."

- (28) Hún bað mig um að hjálpa sér
she.NOM asked me to help REFL.DAT
og ég gerði það. Hverjum hjálpaði ég?
and I did that who.DAT helped I

Correct answer

Henni.
her.DAT

A.5 Question-Answering Tasks

The prompts for the coreference resolution task use the same example sentences as in the fill-in-the-blank tasks. The prompts are on the form:

"Which noun does the pronoun <pronoun> refer to in the following Icelandic sentence: <Example sentence in Icelandic.> Answer only with one noun." The prompts for the attributive agreement task are on the form: "Which of the slash-separated options in the following question forms part of a sentence that is grammatical in Icelandic? <Example sentence in Icelandic with the word 'one' displayed in all three genders.> Answer only with one word." Note the attributive agreement task does not use the same sentences as when the same feature is tested via grammaticality judgments. The prompts for the word sense disambiguation task are on the form: "Does the word tagged with <i></i> in the following two Icelandic sentences have the same meaning? <Two example sentences in Icelandic containing the same word form.> Answer only with one word: True or False."

A.5.1 Coreference Resolution

- (29) Lína ætlaði að sópa kjallarann
Lína meant to sweep basement-the.MASC
með kústi en hann var ekki
with broom-the.MASC but it.MASC was not
á sínum stað.
in its place

Incorrect answer

Kjallarinn.
basement-the

Correct answer

Kústurinn.
broom-the

A.5.2 Attributive Agreement

- (30) Einn/Ein/Eitt
one.MASC/one.FEM/one.NEUT
húðflúranna var af stórum dreka.
tattoos-the.GEN.NEUT was of big dragon

Correct answer

Eitt.
one.NEUT

A.5.3 Word Sense Disambiguation

- (31) **Words used in the same sense**
- a. Hún <i>nam</i> lögfræði við
she studied law at
Háskólann.
university-the
- b. Hún <i>nam</i> grísku við
she studied Greek at
Háskólann.
university-the

(32) **Words used in a different sense**

- a. <i>Gosið</i> var kraftlítið.
eruption-the was weak
- b. <i>Gosið</i> var sykurlaust.
soda-the was sugar-free

B Model Scores by Task

We break down the overall scores for each model by task included in our main benchmark set (see final page). Note that we use truncated model names due to space limitations, see Table 2 for full names.

Grammaticality checks

Model	Simple	AA	PA	WO	VA	SC	Islands	wh	Top.	Gapp.	Refl.
Claude-3-5-Sonnet	90.00	55.68	100.0	92.86	71.43	78.57	87.50	45.00	59.38	82.50	77.50
Claude-3-Opus	95.00	39.77	100.0	82.14	71.43	64.29	83.75	45.00	68.75	81.67	85.00
GPT-4o	100.0	39.77	96.43	78.57	85.71	75.00	93.75	40.00	59.38	73.33	80.00
GPT-4-Turbo	95.00	36.36	75.00	82.14	71.43	64.29	78.75	40.00	75.00	81.67	57.50
GPT-4	100.0	38.64	67.86	78.57	57.14	53.57	82.50	60.00	62.50	69.17	75.00
GPT-4o-Mini	90.00	53.41	85.71	85.71	71.43	46.43	86.25	60.00	59.38	75.00	80.00
Llama-3.1-70B	95.00	39.77	60.71	64.29	67.86	60.71	62.50	50.00	50.00	83.33	42.50
Llama-3.1-405B	85.00	30.68	64.29	60.71	60.71	57.14	85.00	50.00	50.00	72.50	80.00
Gemma-2-27B	95.00	37.50	64.29	53.57	64.29	50.00	82.50	30.00	53.13	70.83	77.50
Mixtral-8x22B	90.00	39.77	53.57	64.29	60.71	46.43	80.00	40.00	53.13	68.33	47.50
Qwen2-72B	85.00	45.45	57.14	57.14	57.14	53.57	42.50	60.00	71.88	75.83	62.50
GPT-SW3-20B	58.00	48.86	50.00	46.43	50.00	50.00	50.00	50.00	50.00	50.00	52.50
GPT-SW3-20B-4bit	55.00	51.14	50.00	39.29	50.00	50.00	65.00	40.00	46.88	46.67	45.00

Table 4: A breakdown of the overall scores for the sentence grammaticality tasks: Simple, unambiguously grammatical or ungrammatical sentences (Simple), attributive agreement (AA), predicate agreement (PA), word order (WO), verb agreement (VA), subject case (SC), island effect sentences (Islands), wh-movement (wh), topicalization (Top.), gapping (Gapp.) and reflexivization (Refl.).

Well-formedness check	
Model	Word formation
Claude-3-5-Sonnet	74.29
Claude-3-Opus	67.14
GPT-4o	62.86
GPT-4-Turbo	38.57
GPT-4	59.29
GPT-4o-Mini	68.21
Llama-3.1-70B	57.14
Llama-3.1-405B	65.00
Gemma-2-27B	65.36
Mixtral-8x22B	42.86
Qwen2-72B	60.00
GPT-SW3-20B	50.00
GPT-SW3-20B-4bit	50.36

Table 5: A breakdown of the overall scores for the well-formedness check of compound nouns.

Fragment answering	
Model	Fragment answers
Claude-3-5-Sonnet	100.0
Claude-3-Opus	100.0
GPT-4o	77.50
GPT-4-Turbo	72.50
GPT-4	82.50
GPT-4o-Mini	62.50
Llama-3.1-70B	72.50
Llama-3.1-405B	97.50
Gemma-2-27B	45.00
Mixtral-8x22B	25.00
Qwen2-72B	25.00
GPT-SW3-20B	0.00
GPT-SW3-20B-4bit	2.50

Table 7: A breakdown of the overall scores for the fragment answering tasks.

Fill-in-the-blank			
Model	Anaphor.	Coref.	Wug
Claude-3-5-Sonnet	100.0	61.36	40.00
Claude-3-Opus	90.00	45.45	10.00
GPT-4o	80.00	52.27	40.00
GPT-4-Turbo	85.00	50.00	20.00
GPT-4	75.00	50.00	20.00
GPT-4o-Mini	45.00	27.27	20.00
Llama-3.1-70B	30.00	40.91	20.00
Llama-3.1-405B	65.00	59.09	20.00
Gemma-2-27B	50.00	13.64	10.00
Mixtral-8x22B	0.00	11.36	0.00
Qwen2-72B	25.00	18.18	0.00
GPT-SW3-20B	10.00	20.45	0.00
GPT-SW3-20B-4bit	0.00	20.45	0.00

Table 6: A breakdown of the overall scores for the fill-in-the-blank tasks: Anaphoric reference (Anaphor.), coreference resolution (Coref.) and wug tests (Wug).

Question-answering			
Model	Coref.	AA	WSD
Claude-3-5-Sonnet	81.82	73.33	84.00
Claude-3-Opus	63.64	80.00	81.33
GPT-4o	86.36	80.00	90.00
GPT-4-Turbo	68.18	63.33	84.00
GPT-4	77.27	60.00	56.67
GPT-4o-Mini	59.09	50.00	66.67
Llama-3.1-70B	65.91	46.67	75.33
Llama-3.1-405B	63.64	83.33	74.67
Gemma-2-27B	59.09	36.67	62.67
Mixtral-8x22B	43.18	3.33	57.33
Qwen2-72B	47.73	33.33	65.33
GPT-SW3-20B	25.00	66.67	56.67
GPT-SW3-20B-4bit	29.55	43.33	35.33

Table 8: A breakdown of the overall scores for the question-answering tasks: Coreference resolution (Coref.), attributive agreement (AA) and word sense disambiguation (WSD).

Transfer-Learning German Metaphors Inspired by Second Language Acquisition

Maria Berger

Ruhr University Bochum

maria.berger-a21@rub.de

Abstract

A major part of figurative meaning prediction is based on English language training corpora. One strategy to apply techniques to languages other than English lies in applying transfer learning techniques to correct this imbalance. However, in previous studies, we learned that the bilingual representations of current transformer models are incapable of encoding the deep semantic knowledge necessary for a transfer learning step, especially for metaphor prediction. Hence, inspired by second language acquisition, we attempt to improve German metaphor prediction in transfer learning by modifying the context windows of our input samples to align with lower readability indices achieving up to 13% higher F1 score.

1 Introduction

Figurative language detection is one of the most crucial tasks in the current digital conversational landscape. However, computationally, it remains also one of the most challenging tasks. Comprehensive resources to train computational models for figurative language detection are generally rare. Further, most existing work is performed on English language textual data. Some works investigate metaphor recognition in languages other than English (Sanchez-Bayona and Agerri, 2022; Aghazadeh et al., 2022).

We focus on applying and testing transfer learning techniques to continuously correct for this imbalance in figurative language prediction. We think that, due to the conceptual nature of metaphors (Lakoff and Johnson, 1980), it is possible to transfer metaphoric meaning given a sufficient amount of data that is capable of encoding this conceptual nature.

The study in this paper is designed as follows: First, we address the motivations of this research by presenting the readability indices of the predicted test samples of a prior study (Berger et al., 2024). Then, we modify the test samples according to these insights by trimming the observed contexts. This means, shortening the input. Last, we re-apply the multi-lingually pre-trained transformer models to determine how the sample modification affects the performance of the multilingual classifiers.

2 Related work

Tsvetkov et al. (2013, 2014) use lexical-semantic word features as well as bilingual dictionaries in several languages as input data for transfer learning to recognize metaphorical expressions across languages. Also, using syntactic patterns or abstractness scores is a common technique to identify or analyze metaphoric expressions (Tsvetkov et al., 2013; Clausen and Nastase, 2019).

Clausen and Nastase (2019) investigate the effect of text simplification on linguistic metaphor preservation (Wolska and Clausen, 2017; Clausen and Nastase, 2019). The authors provide an analysis of parallel text data that are simplified for different grade levels identifying whether metaphors are either preserved, rephrased, or dropped. They also investigate which features are capable of discriminating on whether a metaphor is preserved or dropped and determine that age-of-acquisition scores, imagine-ability, and concreteness scores are useful in the tasks.

Berger et al. (2024) perform a comprehensive study on applying transfer learning to German metaphor prediction, framing the problem as both a sequence labeling and sentence classification task. Several pre-trained transformers are fine-tuned on English metaphor-labeled data and tested regarding their capabilities to identify metaphors cross-lingually. However, multilingual

classifiers perform only moderately well, because the cross-lingual semantic knowledge that these models need to be capable of encoding appears to be hidden deep within the semantic representation of a language.

3 Methodology

We already learned that computational approaches work well in semantically “coarse-grained” tasks such as semantic similarity prediction (Kenter and De Rijke, 2015; Moritz and Steding, 2018; Wang et al., 2020) or authorship attribution (Benze-bouchi et al., 2018), because they are well capable to distinguish the meaning of a word in different contexts. In figurative language identification, contextual representation is also a good input for a classifier to predict whether a word is meant figuratively or literally (Bizzoni and Lappin, 2017; Bizzoni and Ghanimifard, 2018; Liu et al., 2020).

Transfer learning typically makes use of a well-resourced source language to train a classifier on, afterwards, the trained model is applied to predict metaphors in a low/less-resourced language. However, there are two major problems to make figurative language prediction work cross-lingually: first, only a few larger (lexicon-dependent) annotated datasets for training in the source language are available¹; second, the translation models of today’s transformers are incapable to encode the deep semantic knowledge required for transfer identification of figurative language (Berger et al., 2024).

As syntax is the structural representation of meaning, one can carefully state that sentences of more complex syntax usually also entail more complex semantics. As such, “adding” tokens to a string also often (not always) means to “add” semantics to the meaning of a phrase or sentence. This can be partially validated by Batiukova and Pustejovsky (2013) who investigate the role of compositionality and lexical semantics in determining informativeness at the phrasal level.

As we understand that the transformer models may need to be presented with “easier” (shorter, less complex) samples because this is the case when learning a new language, we attempt to improve German metaphor prediction in transfer learning by modifying the context windows of

our input samples to align with lower readability indices. In particular, trimming the context of a potential metaphoric expression can aid in the model’s focus on nearby domain-related context while long-distance context may be less relevant, and the preserving of the sentence’s global meaning possibly plays a subordinated role. To backup the latter assumption, we also test a more sophisticated technique by applying Klöser et al. (2024)’s GPT2.0-based text simplifier—to the best of our knowledge the only model, applicable for German language text—to our test data.

3.1 Transformer models and data (re-)used

We first recapture the zero-shot transferred results from a former study (Berger et al., 2024) that applies multilingual transformers mBERT (Devlin et al., 2018), XLM-RoBERTa (Liu et al., 2019), and sentence transformers (SBERT) (Reimers and Gurevych, 2019) to predict German metaphors from a small German language test set.²

The pretrained transformer models were fine-tuned on the established English language VUA metaphor corpus (Steen et al., 2010) and tested on a smaller German metaphor dataset (Berger et al., 2024).³ The task was designed as a sentence classification problem—inspired by Gao et al. (2018)’s embeddings approach—whereas every input was accompanied by the position of a verb in the sentence and the label whether that verb was used metaphorically (1) or literally (0) in the given context.

Note: Typically, metaphoric meaning predication normally is designed by token labeling or a word classification problem, not a sentence classification problem. Linguistically, however, it is common sense to identify a metaphor based on its source (image provider) and target (recipient of an image). The German test data that we use is a derivative of a linguistically annotated English language metaphor corpus that initially was annotated for lexical representatives of a metaphor

¹The VUA Metaphor corpus (Steen et al., 2010), the TroFi corpus (Birke and Sarkar, 2006), and the MOH datasets (Mohammad et al., 2016) are among the larger ones.

²SBERT is an enhancement of the traditional BERT model, but it is specialized for problems of semantic similarity from sequential input (sentence) embeddings. It is only about half the size of the other two models and also trains/tests much faster.

³We use training:validation:testing data splits from the VUA corpus according to Gao et al. (2018) (15,516:1,724:5,873). These do not represent the most recent version of the VUA corpus, but enables us to compare our results with earlier results. As such, Gao et al. (2018) reach 58.9% F1 (acc. 69.1%) and 69.7% F1 (acc. 81.4%) with both their classifiers in a mono-lingual setup.

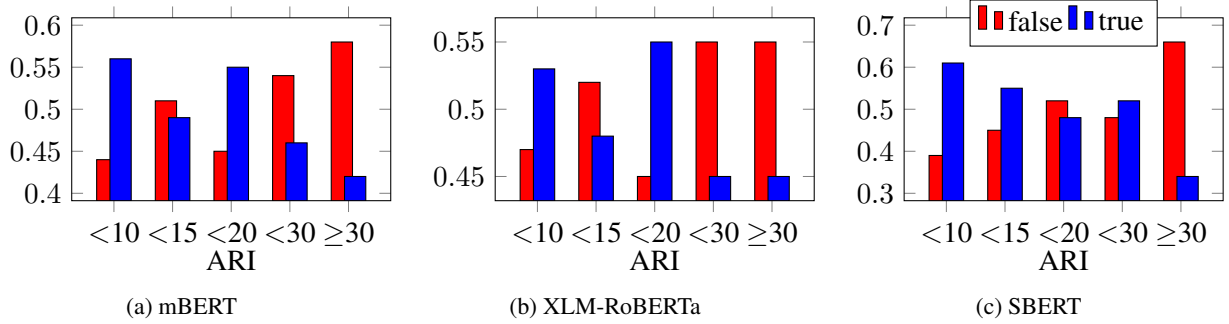


Figure 1: correct (blue) and wrong (red) prediction ratios grouped by 6 different ARI ranges

source and its target. This means, it does not provide labeled metaphoric meaning on the token level. But for trans-lingual metaphor classification (Gao et al., 2018), we can better compare our results with previous results based on this sentence classification set-up.

We use freely available bi-directional encoder representations from transformers instead of the emerging LLMs, because pre-trained BERT models are well-investigated and easily applicable for niche tasks such as transfer-learning for figurative language prediction in German.

3.2 Applying automated readability index

The grade level by Smith and Senter (1967), also known as automated readability index (ARI) is a well-performing measure for text complexity as it considers word and sentence complexity. We start by grouping correct and wrong predictions by the automated readability index (ARI) (Smith and Senter, 1967). We define five groups ranging the ARI below 10, lower than 15, lower than 20, lower than 30 and higher or equal to 30. These ranges approximately align with elementary school students ($ARI < 13$), junior high school students (13-19 ARI), senior high school students ($ARI 20-27$), college students (≤ 28).⁴ Figure 1 shows the predictions of the multilingual transformer models mBERT, XLM-RoBERTa and SBERT according to the groups of ARI scores the classification samples belong to. All of them show a strong correlation between false predictions and ARI scores. SBERT shows the most uniform curves.

3.3 Modifying input representations

As a pre-processing step, we simply trim our testing data to only allow a window of up to 3, 5 and

⁴We refer to Smith and Senter (1967)’s grade level (GL). However, ARI score is more often used as common sense (see also Sec. 5).

10 tokens to both sides of the given verb index, which results in a text snippet of 7, 11, 21 tokens respectively. This way we “simplify” our samples in a computationally easy manner. Because neural models that encode semantics of sentences as input representations cannot “understand” syntax—even though they can cope with it (de Dios-Flores et al., 2023)—it does not matter that our simplification approach ignores the tree-like structure of actual sentences. For typical neural classifiers, important features are mostly given by the contextual, especially sequential representations. Hence, trimming the surrounding longer-distance context makes the model stress close-context relations, which is especially important in figurative meaning prediction. Tab. 1 shows examples to illustrate how trimming modifies input samples.

Running Klöser et al. (2024)’s simplifier on our data removes metaphoric expressions or does not return a representation at all in almost 80% of the samples. Hence, we test metaphor prediction for the remaining 193 samples only.

4 Results and Discussion

Tab. 2 shows that the best performance increase can be reached with trimming the contextual span for the input representation to 11 tokens. Also, a context of 7 enables the models to drastically increase on performance while allowing a window of 10 to each side still results in an increase of up to 6% in F1 (c.f., upper part of Tab. 2). The neural simplification (that also preserves a sentence’s meaning) achieves up to 9% increase in F1 returning the second highest F1 score.

Looking at samples from the SBERT output, we find that limiting the context can help the model to better focus on local meaning. For example in the form of not labeling words as figurative that actually are not used figuratively. While the following

text	label predicted window		
[...] auf der glücklichen Seite des Schweinetrogs stehen, schmeckt Demokratie ziemlich süß.	1	0	orig.
Seite des Schweinetrogs stehen , schmeckt Demokratie ziemlich süß .	1	1	5
[...] on the lucky side of the pork trough, democracy tastes pretty sweet.			

Table 1: Sample sentences next to predictions; label 1: metaphorically meant; 0: literally meant

model	approach	precision	recall	f1-score (+increase)	accuracy
mBERT	original	58	46	52	50
XLM-RoBERTa	sentence	58	44	50	50
SBERT	length	57	65	61	51
mBERT	window 3	67	62	65 (+13)	61
XLM-RoBERTa		65	60	62 (+12)	59
SBERT		67	72	69 (+8)	63
mBERT	window 5	66	62	64 (+12)	60
XLM-RoBERTa		66	59	63 (+13)	59
SBERT		66	80	72 (+11)	65
mBERT	window 10	63	53	58 (+6)	55
XLM-RoBERTa		60	50	55 (+5)	52
SBERT		60	73	66 (+5)	57
mBERT	Klöser et al. (2024) simplified, 193 test samples	70	51	59 (+7)	52
XLM-RoBERTa		71	40	51 (+1)	48
SBERT		70	70	70 (+9)	59
mBERT	fine-tuned on DE metaphor, 98 test samples	91	88	90	88
XLM-RoBERTa		81	86	83	81
SBERT		73	92	82	78
mBERT	fine-tuned on EN metaphor, 908 test samples	82	81	82	79
XLM-RoBERTa		84	83	84	81
SBERT		64	95	76	66

Table 2: precision, recall, f1, accuracy (%) according to a context of 7, 11, 21 tokens; trained on VUA corpus with train:val splits 15,516:1,724 tested on 908 DE language samples; upper part: original setup; mid part: input samples trimmed to window sizes and Klöser et al. (2024)’s simplification approach; lower part: fine-tuned on EN metaphor, splits: 1360:341:908 and fine-tuned on DE metaphor, splits: 720:90:98 (=908)

example was an FP before, it now is classified as TN:

“Der Finanzmanager **erstellt**(TN) Finanzberichte [...]” [The financial Manager **prepares** financial reports [...]]. Some could argue that “erstellt” might take the role of personification in the following context. This borderline example was labeled by SBERT as FP before. With the trimmed context, SBERT labels the examples as TN.

Regarding TPs, the following example shows how SBERT can make better use of unusual relationships learned in the source language it was trained on. Hence, it interpretes the following example correctly in a figurative sense: “[...] auf

der glücklichen Seite des Schweinetrogs stehen, **schmeckt**(TP) Demokratie ziemlich süß.” [...] on the lucky side of the pork trough, democracy **tastes** pretty sweet.]

For comparison, in the lower part of Tab. 2, we also list the results of fine-tuning in German, based on the 908 De language samples, which we split into train, validation and test sets (Berger et al., 2024). We can see that fine-tuning on target data and language after training in the VUA data brings the best results (Berger et al., 2024).

When we test whether fine-tuning on target domain English language data improves the test results in our German language data, we find a positive effect. Especially, XLM-RoBERTa shows the

ability to well generalize to language-independent data points when the source (training) and target (testing) domain remain the same. This can be explained by the dynamic masking process during RoBERTa’s initial training process. However, in semantically challenging set-ups, this flexibly rather prevents RoBERTa from retrieving unknown items, as can be seen in the results of applying RoBERTa to our German metaphor data after only training on the VUA corpus (second line of Tab. 2).

model	window	averaged ARI	
		correct	wrong
mBERT	3	8.1	8.9
XLNet-RoBERTa		7.8	9.4
SBERT		8.1	9.1
mBERT	5	8.6	9.3
XLNet-RoBERTa		8.4	9.6
SBERT		8.1	10.3
mBERT	10	11.1	12.0
XLNet-RoBERTa		11.1	12.0
SBERT		11.1	12.2
mBERT	simplified	13.4	13.5
XLNet-RoBERTa		12.8	14.1
SERT		13.5	13.5

Table 3: Averaged ARI score of the correct and wrong predictions after trimming

Table 3 shows the averaged ARI scores for the correct and wrong predictions of the three models. Almost every set-up shows that the averages of the ARI score are at least one point higher in the wrong predictions class compared to the correct predictions class. This inverse relationship between a model’s ability to predict figurative language and ARI scores leads to the insight that certain lexical and textual properties—independent from the classifier—challenge the prediction of a verb’s meaning in a given context. On the other hand, SBERT—our task-favorite—shows equal ARI scores in the simplification setup. It also is the model that reacts not as drastically to the trimming as the other models do. This hints us to investigate both more deeply, i) a model’s translation representations, and ii) verbalization of metaphor in simpler sentence structures.

5 Remarks

ARI was initially designed for English language text: A possible weakness of this approach may

be that the automated readability index (Smith and Senter, 1967) was originally designed to test students’ capability to understand and comprehend the content of an English language text that also meets certain structural conditions. Because characters per word and words per sentence distribution differ across different languages, the grade-levels defined in 3.2 may not apply to our German language test data. However, Senter & Smith’s score was used before to estimate the complexity reduction of text in languages other than English. For example in Moritz et al. (2016) and Tillman and Hagberg (2014). In the current study, we use the ARI score to obtain an understanding of prediction difficulty, and we think that applying the ARI score in this context is appropriate.

Shortening is not simplification: It is not always the case that a metaphor is difficult to extract because a sentence is syntactically complex, nor is it always true that humans understand shorter sentences better than longer ones. But, sentence simplification usually divides up complex content into many shorter sentences and this also improves metaphor recognition for a computational model. Further, our trimming approach is technically simple and streamlined and shows already good results. We further will elaborate on a quantitative approach that incorporates advanced syntax-tree rules into our window-trimming technique.

6 Conclusion

We demonstrated a computationally simple approach to correct input representation to make them shorter, hence, easier for the model to understand, because—as in second language acquisition, we learned that the translation representations of transformer models have some difficulty in “understanding” the deep semantics required for figurative meaning classification. We also applying a GPT-based simplifier. We achieve an increase of up to 13% (11-token context) and up to 9% (neural simpl.) in F1 and find that the sentence transformer models perform best in metaphor prediction. In future, we plan to apply didactically-informed approaches that utilize linguistic, comparative, and didactic knowledge while being applicable to quantitative methods as well.

References

Ehsan Aghazadeh, Mohsen Fayyaz, and Yadollah Yaghoobzadeh. 2022. Metaphors in pre-trained lan-

- guage models: Probing and generalization across datasets and languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2037–2050.
- Olga Batiukova and James Pustejovsky. 2013. Informativeness constraints and compositionality. In *Proceedings of the 6th International Conference on Generative Approaches to the Lexicon (GL2013)*, pages 92–100.
- Nacer Eddine Benzebouchi, Nabih Azizi, Monther Aldwairi, and Nadir Farah. 2018. Multi-classifier system for authorship verification task using word embeddings. In *2018 2nd International Conference on Natural Language and Speech Processing (IC-NLSP)*, pages 1–6. IEEE.
- Maria Berger, Sebastian Michael Reimann, and Nieke Marie Kiwitt. 2024. Applying transfer learning to german metaphor prediction. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 1383–1392.
- Julia Birke and Anoop Sarkar. 2006. A clustering approach for nearly unsupervised recognition of non-literal language. In *11th Conference of the European chapter of the association for computational linguistics*, pages 329–336.
- Yuri Bizzoni and Mehdi Ghanimifard. 2018. Bigrams and bilstms two neural networks for sequential metaphor detection. In *Proceedings of the Workshop on Figurative Language Processing*, pages 91–101.
- Yuri Bizzoni and Shalom Lappin. 2017. Deep learning of binary and gradient judgements for semantic paraphrase. In *IWCS 2017—12th International Conference on Computational Semantics—Short papers*.
- Yulia Clausen and Vivi Nastase. 2019. Metaphors in text simplification: To change or not to change, that is the question. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 423–434.
- 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*.
- Iria de Dios-Flores, Juan Garcia Amboage, and Marcos García. 2023. Dependency resolution at the syntax-semantics interface: psycholinguistic and computational insights on control dependencies. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 203–222.
- Ge Gao, Eunsol Choi, Yejin Choi, and Luke Zettlemoyer. 2018. Neural metaphor detection in context. *arXiv preprint arXiv:1808.09653*.
- Tom Kenter and Maarten De Rijke. 2015. Short text similarity with word embeddings. In *Proceedings of the 24th ACM international conference on information and knowledge management*, pages 1411–1420.
- Lars Klöser, Mika Beele, Jan-Niklas Schagen, and Bodo Kraft. 2024. German text simplification: Fine-tuning large language models with semi-synthetic data. *arXiv preprint arXiv:2402.10675*.
- George Lakoff and Mark Johnson. 1980. *Metaphors we live by*. University of Chicago Press.
- Jerry Liu, Nathan O’Hara, Alexander Rubin, Rachel Draelos, and Cynthia Rudin. 2020. Metaphor detection using contextual word embeddings from transformers. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 250–255.
- 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*.
- Saif Mohammad, Ekaterina Shutova, and Peter Turney. 2016. Metaphor as a medium for emotion: An empirical study. In *Proceedings of the fifth joint conference on lexical and computational semantics*, pages 23–33.
- Maria Moritz, Barbara Pavlek, Greta Franzini, and Gregory Crane. 2016. Sentence shortening via morpho-syntactic annotated data in historical language learning. *Journal on Computing and Cultural Heritage (JOCCH)*, 9(1):1–9.
- Maria Moritz and David Steding. 2018. Lexical and semantic features for cross-lingual text reuse classification: an experiment in english and latin paraphrases. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.
- Nils Reimers and Iryna Gurevych. 2019. <https://aclanthology.org/D19-1410> Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.
- Elisa Sanchez-Bayona and Rodrigo Agerri. 2022. Leveraging a new Spanish corpus for multilingual and cross-lingual metaphor detection. In *Proceedings of the 26th Conference on Computational Natural Language Learning (CoNLL)*, pages 228–240. Association for Computational Linguistics.
- Edgar A Smith and RJ Senter. 1967. *Automated readability index*, volume 66. Aerospace Medical Research Laboratories (U.s.), Aerospace Medical Division, Wright-Patterson Air Force Base: 1–14. PMID 5302480. AMRL-TR-6620.

- Gerard J Steen, Aletta G Dorst, J Berenike Herrmann, Anna A Kaal, and Tina Krennmayr. 2010. Metaphor in usage. *Cognitive Linguistics* 21–4.
- Robin Tillman and Ludvig Hagberg. 2014. Readability algorithms compability on multiple languages. KTH.
- Yulia Tsvetkov, Leonid Boytsov, Anatole Gershman, Eric Nyberg, and Chris Dyer. 2014. Metaphor detection with cross-lingual model transfer. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 248–258.
- Yulia Tsvetkov, Elena Mukomel, and Anatole Gershman. 2013. Cross-lingual metaphor detection using common semantic features. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 45–51.
- Congcong Wang, Paul Nulty, and David Lillis. 2020. A comparative study on word embeddings in deep learning for text classification. In *Proceedings of the 4th international conference on natural language processing and information retrieval*, pages 37–46.
- Magdalena Wolska and Yulia Clausen. 2017. Simplifying metaphorical language for young readers: A corpus study on news text. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 313–318.

Comparative Concepts or Descriptive Categories: a UD Case study

Matthieu Pierre Boyer

Lattice
DI ENS
Paris, France
matthieu.boyer@ens.fr

Mathieu Dehouck

Lattice, CNRS, ENS-PSL, USN
mathieu.dehouck@cnrs.fr

Abstract

In this paper, we present a series of methods used to quantify the soundness of using the same names to annotate cases in different languages. We follow the idea described by Martin Haspelmath that descriptive categories and comparative concepts are different objects and we look at the necessary simplification taken by the Universal Dependencies project. We thus compare cases in closely related languages as belonging to commensurable descriptive categories. Then we look at the corresponding underlying comparative concepts. We finally looked at the possibility of assigning cases to adpositions.

1 Introduction

There is a fundamental distinction between language-particular categories of languages (which descriptive linguists must describe by descriptive categories of their descriptions) and comparative concepts (which comparative linguists may use to compare languages).

Martin Haspelmath in (Haspelmath, 2018)

Language description and language comparison are two intertwined yet distinct endeavours. Language description is often done in a language different from the one being described (many grammars have been written in English, French, Russian, Spanish and Portuguese for example) and often uses a conventionalised descriptive meta-language associated with a given descriptive school. Language comparison relies on the previous step of language description as it

main data source but also needs a common meta-language to name the various phenomena under study.

In his paper, Haspelmath (2018) warns us against the confusion of the different meta-languages (the descriptive languages used in each individual description and the common comparative meta-language). He advocates for a careful choice of terms when describing similar categories across multiple languages, even when the similarities compel us to use the same term. That is, one should avoid using a single term to describe two categories from two different languages. Even more so, when this term is also used as a comparative concept which then further increases the risk of cross-meta-language confusion.

With all its qualities, the Universal Dependencies (UD) project (Zeman et al., 2024) puts itself exactly in this somewhat uncomfortable situation. One of the main aims of the project is to foster linguistic typological research, and thus it proposes a common annotation scheme for creating treebanks for all natural languages (de Marneffe et al., 2021). Figure 1 depicts the dependency tree of a Turkish sentence as an example. While the scheme has means to accommodate language specific phenomena, its core is language agnostic and treebank creators are compelled to reuse previously defined language specific extensions when annotating similar structures in new languages as a mean to increase the overall consistency and comparability of the corpora. In the dependency tree, the labels of the edge going out of a node is called its *dependency relation* and the target of the edge it the *governor* of the node. However, the annotation also needs to be sound from the point of view of each annotated language (see points 1 and 2 of the presentation page at <https://universaldependencies.org/introduction.html>). Each individual treebank can thus be seen as a kind of description

of its language. Indeed, that is exactly what Herrera et al. (2024) do in their work, where they use sparse representation methods to try to extract a grammar sketch for a language from its annotated treebank. In UD, the same terms are thus used both as comparative concepts and as descriptive categories for all the languages that express that category.

In this study, we investigate the descriptive-comparative confusion arising from UD’s annotation scheme at the morphosyntactic level. We especially focused on the category of case and its different realisations across several languages with the following question in mind: Do cases sharing their name have the same value across different languages? The main reason to focus on the case category, is that it has both strongly syntactic and strongly semantic values. For example, in languages with a case marking the subject of both transitive and intransitive verbs, this case is usually called NOMINATIVE¹ based on its syntactic properties. If the same language has another case marking the “together with” relation, it will usually be called COMITATIVE on semantic ground.

This study should provide insight on the extent to which one can transfer information about a feature from a language to another simply by reusing the same name (using the same descriptive category). In the end, it could help improve cross-lingual learning scenarios where we want to use as much information from other languages as we can, even at the morphological and syntactic levels.

This paper is organised as follows. Section 2 gives an overview of UD’s guidelines on case annotation and how these are realised in practice. Section 3 describes how we assign representations to cases. Section 4 looks at the similarity between cases from different languages as if they were descriptive categories. Section 5 then turns to looking at cases as comparative concepts applied to each individual treebank. Section 6 takes an in between look directly at the cases from all the treebanks. Section 7 investigates the possibility of assigning cases directly to adpositions. Eventually, Section 8 concludes this paper.

¹In this paper, we use faces to distinguish between DESCRIPTIVE CATEGORIES, COMPARATIVE CONCEPTS and UD’s annotation scheme.

1.1 Theoretical Note

In this work, we decided to question the relevance of using the same name to refer to cases in different languages. This assumes the existence of a commensurable case category in each language of interest. There is however no reason to take it for granted.

We decided to take a very pragmatic stance. Universal Dependencies (and indeed, many linguists) assumes a commensurable case category existing across languages. So, we acknowledge this choice. We neither question the existence of a case category in different languages, nor do we question the number of values displayed by said category in each language of interest. We question the relevance of the names given to the different values in different languages.

2 The Case Feature across Treebanks

While realising this study, we stumbled upon a number of incongruities in the way the different corpus use the `Case` feature.

There are essentially three ways the feature `Case` is used in the UD treebanks. The first and by far the most common use is to annotate inflected forms of nouns, pronouns and proper nouns in languages where these words inflect according to their role in a clause, as well as determiners, adjectives and participles in languages where they inflect to match the case of their governor.

The second use that is documented in UD’s guidelines², is to annotate adpositions with the case they give to their nominal phrase, especially so in languages without over case marking on nouns. This annotation principle indicates that UD leans more toward the application of comparative concepts to individual languages. Indeed, if a language does not use the case category, then the “case” represented by an adposition can only be inferred either by comparing its distribution to the distribution of actual cases in languages that possess that category, or by applying formal comparative definitions.

However, this is not always how this feature is used, as in Czech CLTT treebank (Kříž and Hladká, 2018) for example, adpositions are annotated with the `Case` feature and their value always match that of their governing noun. This is all the

²See the page of the `Case` feature: <https://universaldependencies.org/u/feat/Case.html>.

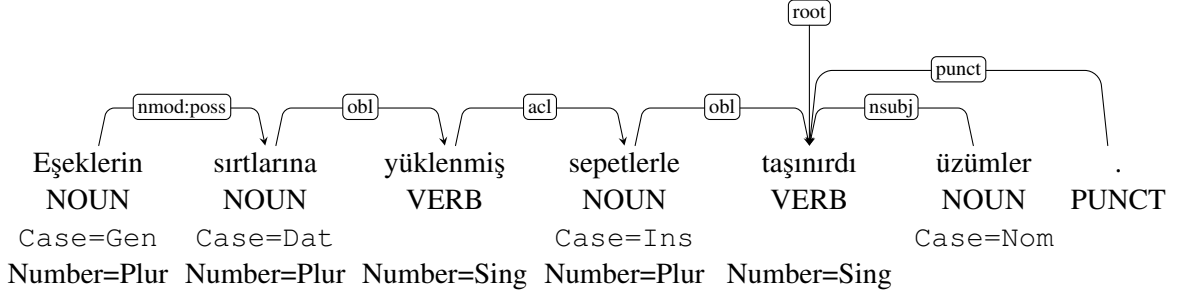


Figure 1: Representation of the dependency graph of the Turkish sentence "Eşeklerin sırtlarına yüklenmiş sepetlerle taşınırdı üzümler." from UD's Turkish BOUN corpus, meaning "Grapes were carried in baskets loaded on donkeys' backs."

more surprising that Czech adpositions are invariable and can license several case values.

This indeed points to another problem with case annotation on adpositions. Like languages exhibiting case syncretism³, adpositions can in principle also be used to mark different syntactic and semantic roles. It becomes then even less clear how one should proceed in assigning cases to adposition.

The third and most divergent use of the *Case* feature can be seen in the Persian Seraji treebank. In this treebank, we only find three case values : *Case=Loc*, *Case=Tem* and *Case=Voc*. The first two values are exclusively used to annotate adverbs of place and adverbs of time respectively. The third value is used to annotate an interjection used to create vocative noun phrase.

3 Case Representation

In order to compare cases from different languages, we need to find a shared representation that should be as language agnostic as possible. We decided to use the syntactic profile of a case defined as the probability distribution⁴ over the dependency relations to its governors. This choice is both theoretical since core cases are usually defined in terms of syntactic relations to the other constituents of a sentence, and practical since UD treebanks are annotated with dependency labels.

In order to make the representations even more language agnostic, we decided to ignore rela-

³A given word form can be ambiguous as to its morphological features. For example, the Latin form *rosae* can be either a genitive or dative singular or a nominative or vocative plural.

⁴This may be better thought of as normalized frequency distributions, since the case of a word is not a random variable but rather the result of its use in context. But mathematically, normalized frequencies can be viewed as probability distributions.

tion sub-types since they are not consistently used across languages and corpora. So, both *flat:foreign* and *flat:name* are counted as *flat*.

We give two representations to each case in a language. The first is the empirical probability distribution of the relation of a word displaying that case to its governor.

However, there are several mechanisms underlying case assignment, and not all are as informative. For example, when determiners inflect for case, they usually inherit their value from their head noun, which therefore does not teach us much about that case since a determiner can in principle take any case that way. Similarly, it would artificially separate cases from languages with articles (a high proportion of *det* relations) from those of languages without.

Furthermore, as mentioned in the previous section, UD also allows annotation of the *Case* feature on adpositions, which is quite different from the way cases are generally assigned to nouns. For all these reasons, we thus decided to have a part-of-speech based representation too.

The second representation is thus the syntactic profile of the nouns (NOUN) which bear the said case. This gets rid of less informative dependency relations such as *case*, *amod* or *det* and we further decided to ignore the *conj* relation for similar reasons.

The relation distributions are computed from the concatenation of the three parts (train, dev and test) of each treebank from UD version 2.14 (Zeman et al., 2024), except when precised otherwise.

4 Sharing Descriptive Categories

With our case representations, we first look at cases used in different treebanks as representing the values of a descriptive category. We want to know how relevant is to apply the same name to values of a similar category in different languages.

First, we compare case labels from two closely related languages, namely Czech and Russian⁵. To do so we compute the euclidean distance between each case in the first language and each case in the second language. Then, we generate a 1-nearest neighbour graph assuming the neighbours of a node must come from the other language. This gives us an idea of the way cases could be mapped in a transfer learning setting for example.

Figure 2 represents the 1-nearest neighbour graph of Czech and Russian cases when representations are computed over all the words marked for case. We see that the Czech and Russian NOMINATIVES are each other’s nearest neighbour and such is the case for the two genitives. However, for the other cases, the picture is less clear. This is likely due to the fact that when we compute the representations using all the parts-of-speech at once, we confuse the different types of case assignment.

Figure 3 which represents the 1-nearest neighbour graph of Czech and Russian cases when representations are computed only on nouns, is clearer. On top of the NOMINATIVES and GENITIVES, the ACCUSATIVES and INSTRUMENTALS are also each other’s nearest neighbours. Only the DATIVES, LOCATIVES and Russian PARTITIVE are still entangled. Looking directly at the data, we realize that the *iobj* relation is never used in the Czech CLLT corpus. The increased probability of seeing a noun in the DATIVE descending from an *obl* relation makes the Czech DATIVE more distinct from the Russian DATIVE and the Czech LOCATIVE is.

The distance matrices for these two graphs can be found in the appendix, along with distance matrices for Czech - Turkish. In the latter, we might for example see that the *equative* behaves erratically on nouns, but that simply comes from the fact that only one noun is annotated with *equative* in the Turkish BOUN corpus.

Note that not all pairs of languages are as well behaved as Czech and Russian, as we shall see in Section 6.

⁵We tried a number of pairs and decided to just present Czech and Russian for space reason.

Case	Description	DepRel
NOMINATIVE	Subject of a clause.	<i>nsubj</i>
ACCUSATIVE	Direct object of transitive verbs.	<i>obj</i>
ABSOLUTIVE	Subject of intransitive verbs and object of transitive verbs.	<i>nsubj</i> <i>obj</i>
ERGATIVE	Subject of transitive verbs.	<i>nsubj</i>
GENITIVE	Noun complement, typically possessor.	<i>nmod</i>
DATIVE	Indirect object of verbs, typically recipient of giving verbs.	<i>iobj</i>

Table 1: Ideal description of a few cases and corresponding UD’s dependency relations.

5 Applying Comparative Concepts

In the previous section we have compared cases from two languages as if they were from a commensurable descriptive category. In this section, we take the other view that Universal Dependencies defines comparative concepts and that the various treebanks are annotated with these concepts. This means that each case has a language agnostic definition and that it is then applied to each language accordingly. Here, the data used is only from the dev part of the treebanks, for computational time reasons.

Since we do not have language agnostic mathematical representations of the various grammatical cases used in UD’s annotations, we need to extract them from the available annotated corpora. Since a case profile depends not only on the choice of a language, but also on the sentences in the corpus (replacing a few sentences will generally slightly affect the frequency distribution), we model each comparative case with a random variable taking values from the probability distributions (or normalized frequency distributions) over the set of dependency relations to a word’s governor.

Formally, let c be a case, d a dependency relation and \mathcal{T} a treebank. We note $f_{\mathcal{T}}(c, d)$ the frequency at which a word inflected in case c is attached to its governor via a relation of type d in corpus \mathcal{T} . Let $\pi_{\mathcal{T}}(c, d) = f_{\mathcal{T}}(c, d) / \sum_{d'} f_{\mathcal{T}}(c, d')$ be the corresponding probability, and $\pi_{\mathcal{T}}(c, \cdot)$ the

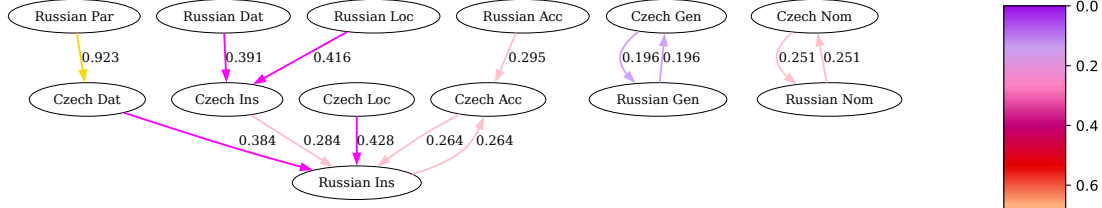


Figure 2: Nearest neighbour graph for Czech CLTT and Russian GSD case profiles. The corresponding distance matrices are Tables 5, 7 and 9 in the appendix.

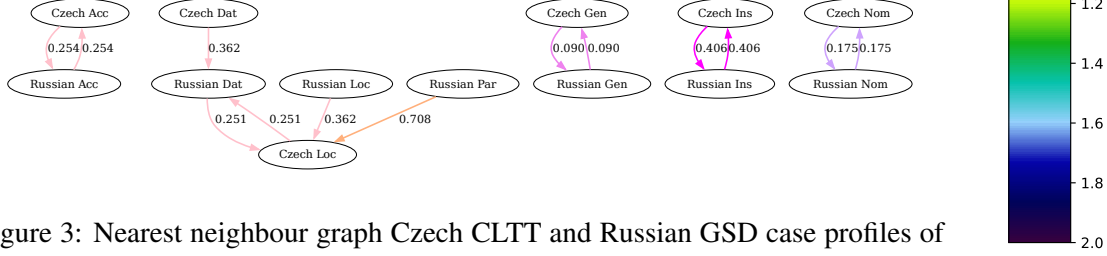


Figure 3: Nearest neighbour graph Czech CLTT and Russian GSD case profiles of nouns. The corresponding distance matrices are Tables 6, 8 and 10 in the appendix.

corresponding probability distribution. We model the case c as a random variable over the probability distributions $\pi.(c, \cdot)$.

We know each of these random variables through a number of realisations: the vector representations of the considered case across all corpora where it is present (which are exactly the probability distributions representations $\pi_{\mathcal{T}}(c, \cdot)$ for each corpus \mathcal{T}). That is, this random variable maps a language/corpus to a probability distribution over the dependency relations reaching words marked with that case.

Then, to compute the profile of the comparative cases, we compute the expected value of the random variables associated to each case. Since the values of the random variables are distributions we also compute the barycenter of the realisations of each variable for the Wasserstein 1-distance (or Earth Mover’s Distance). We will denote the latter by *Wasserstein barycenter*.

Table 2 gives the representations of the expected distributions of a few selected comparative cases. The representations are mostly aligned with our expectations. But we can still notice a few interesting facts. The ERGATIVE is much more strongly associated with being a subject than the NOMINATIVE is. There may be a few different reasons to that. First, some language like Turkish use the nominative/accusative distinction also to mark a definite/indefinite distinction on the object,

with the accusative being kept for definite objects. Another possibility is that when a language has case marking but does not make distinction between subjects and objects such as Irish, it is by default assumed to be nominative-accusative, with the nominative assuming both syntactic roles⁶.

Another interesting fact is that the DATIVE’s main role is not that of indirect object but rather of oblique. This comes from the strong limitations that UD imposes on the use of the `iobj` relation. But still, DATIVE is virtually the only case to assume that role.

However, while this representation allows us to distinguish many cases syntactically, it doesn’t allow to distinguish all cases. More specifically, some cases work in the same syntactic constructions and thus are mostly distinguished through their semantic properties. For example, the Finnish ELLATIVE and ILLATIVE are used to signify that a movement respectively comes from a place or into a place. In the sentence “*I went into his house*”, *house* would be in illative in finnish, while in “*I come back from his house*”, *house* would be in ellative.

This is exactly what we see for non-core cases. LOCATIVE, INSTRUMENTAL and ABLATIVE have very similar profiles, essentially distributed between oblique complements of verbs and nominal

⁶In the eventuality that it would be considered an ergative-absolutive language, the default case would likely be called absolutive rather than ergative anyway.

Case	Average	iobj	nmod	nsubj	obj	obl
ABS	Uniform	0.1	3.3	27.2	36.7	22.4
	Wasserstein	0.0	1.6	28.6	52.2	11.2
ERG	Uniform	0.0	0.7	92.4	0.5	5.9
	Wasserstein	0.0	0.5	97.6	1.4	0.3
NOM	Uniform	0.1	8.0	55.6	7.4	5.0
	Wasserstein	0.0	4.9	65.4	9.3	3.8
ACC	Uniform	0.6	7.8	3.8	62.5	20.5
	Wasserstein	0.0	7.2	1.9	57.6	25.9
GEN	Uniform	0.9	67.4	3.9	5.6	14.9
	Wasserstein	0.0	72.9	3.1	4.5	17.9
DAT	Uniform	14.4	14.9	1.9	0.0	57.2
	Wasserstein	19.0	16.4	0.5	0.0	60.5
LOC	Uniform	0.0	16.6	0.9	1.7	69.6
	Wasserstein	0.0	18.8	0.0	0.0	76.2
INS	Uniform	0.0	17.2	1.4	0.0	66.0
	Wasserstein	0.0	21.3	0.0	0.0	73.8
ABL	Uniform	0.0	16.5	1.3	1.0	70.0
	Wasserstein	0.0	17.2	0.0	0.0	78.5

Table 2: Distributions of the most representative dependency relations for a few cases as computed on nouns. Uniform corresponds to the average profile assuming uniform weighting of each corpus profile. Wasserstein corresponds to barycenters computed with the Wasserstein metric taking into consideration that case profiles are not any vector, but actual probability distributions.

modifiers or nouns.

To check the representativeness of a comparative case P of its realisations across treebanks, we compute also compute its energy E .

$$P = \arg \min_{\mu} E \left(\mu, (\rho_i)_{i \in \llbracket 1, n \rrbracket} \right) \quad (1)$$

$$E \left(\mu, (\rho_i)_{i \in \llbracket 1, n \rrbracket} \right) = \frac{1}{n} \sum_{i=1}^n d(\mu, \rho_i) \quad (2)$$

The energies associated to the two barycenters are of the same magnitude, with the Wasserstein barycenter being more exacerbated as can be seen in Figure 4 for the ACCUSATIVE case. Here, the ρ_i are ℓ^1 -normalized vectors representing cases, and d is the metric used to define the geometry of the space (here, we use the ℓ^2 -metric and the Wasserstein 1-distance).

The x-axis represents the different dependency relations leading to nouns in the accusative, the exact list is given in the appendix for convenience.

It represents in red the uniform mean of distributions (the expectancy of the variable), in yellow the barycenter of the distributions associated to the Wasserstein 1-distance and in purple the (un-normalized for graphical purposes) apparition frequency.

We can notably see that for uniform mean some relations are represented because very present in a few languages while this is not the case for the Wasserstein barycenter, which is more centered on the dependency relations present in a lot of languages.

6 Case Clustering

In this section we apply data visualisation techniques as a mean to look at the general landscape of case across languages. This is a way to explore similarity between cases for many languages at once and without assuming a prototypical representation for each case.

From a practical annotation perspective, this

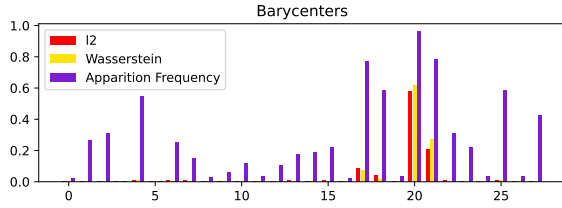


Figure 4: Representation of the uniform barycenter in red and the Wasserstein barycenter in yellow for the comparative ACCUSATIVE case. In purple is represented the proportion of treebanks that associate a given dependency relation with nouns in the accusative from the set of treebanks that inflect their nouns for case.

is interesting since it is more likely too capture the underlying structure of UD’s annotations. Indeed, UD’s guidelines are sometimes underspecified, which is expected from an annotation scheme whose aim is to be applicable to as many languages as possible. Not all use cases and language specific phenomena will have been thought of during the creation of the guidelines. Therefore, when annotators stumble upon a new structure that does not lend itself to a straightforward analysis, they will both turn to the guidelines and to other treebanks in order to see how similar phenomena might have been annotated in other languages.

We first used a *t-SNE* analysis (van der Maaten and Hinton, 2008) with the hope of seeing well defined clusters. However, plotting all the cases at once proved unmanageable and so we resorted to visualising only a pair of cases each time.

The algorithm consists in looking at the probability distribution generated by the high dimensional vectors⁷ representing each instance of the cases and generating a distribution over pairs of those vectors in a way that pairs of *close* vectors are assigned higher probabilities. Then *t-SNE* defines a probability distribution on pairs of 2D points that minimizes the Kullback-Leibler divergence between the two distributions.

Figures 5 and 6 represents the *t-SNE* applied to all the NOMINATIVES and GENITIVES using either the profiles computed on all the words, or just on nouns. It seems that the two cases make for clusters, in the sense they can be grouped along distinct directions. While this is not enough for us to have a classification algorithm, it hints towards

⁷Here the vectors are normalized for the ℓ^1 -norm, but we do not consider them as probability distributions

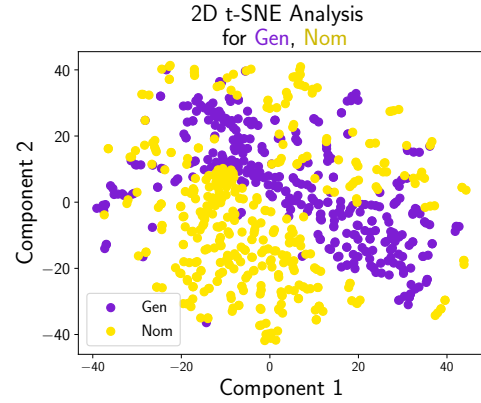


Figure 5: Representation of 2D *t-SNE* analysis of GENITIVE and NOMINATIVE profiles gathered on all the words marked for these cases.

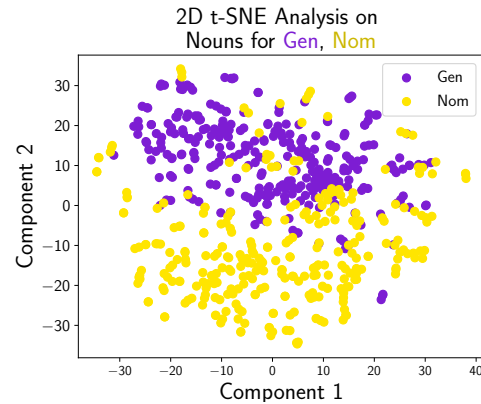


Figure 6: Representation of 2D *t-SNE* analysis of GENITIVE and NOMINATIVE profiles gathered only on the nouns inflected for these cases.

possible ways to visualise the difference between cases.

To confirm this hunch we tried to use *ToMATo* (Chazal et al., 2011), a persistence based clustering algorithm, which uses sub-level sets of a function to design a persistence diagram and derive clusters. The implementation that was used comes from Maria et al. (2014). The idea behind *ToMATo* is to compute the density at each point in the representation space and to cluster points using geodesics: every point above a certain elevation and inside the same geodesic belongs in the same cluster (the same hill) and every point below is ignored.

By repeating the process for different elevations⁸ we can see clusters appear and merge.

⁸*ToMATo* considers the evolution of the topology of superlevel-sets for α of the density function as α decreases and especially their path-connectivity (or 0-persistence in homological terms).

When two clusters merge, the one with the highest elevation absorbs the other and we say that the lowest one dies. One can then represent on a diagram the birth and death time of each cluster. This is depicted in Figure 7 for GENITIVES and NOMINATIVES. The closer a cluster is to the diagonal the shorter its life and therefore the more likely it is to represent random noise rather than an actual cluster

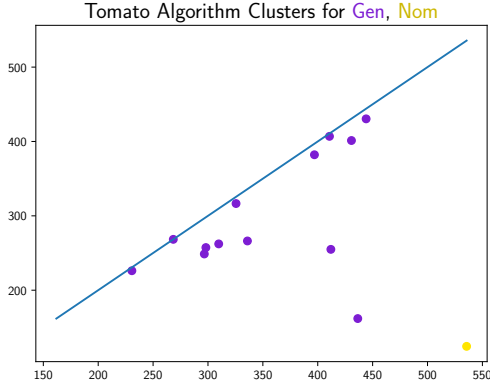


Figure 7: Representation of the ToMATo algorithm for GENITIVE and NOMINATIVE profiles.

In Figure 7 the algorithm proposes multiple clusters, which could not be combined to form better defined clusters. This suggests, as already suggested by Figures 5 and 6 that the possible clusters are not well defined and might overlap with each other. To try and measure the overlap of the clusters, we computed a confusion matrix by the method of the *k*-nearest neighbours.

Pred. \ Target	Target			
	Acc	Gen	Loc	Nom
Acc	130	62	51	34
Gen	69	156	16	42
Loc	35	57	29	34
Nom	29	28	9	227

Table 3: Confusion matrix for *k*-NN with $k = 11$ on Acc, Gen, Loc, Nom. Rows correspond to the prediction and columns to the expected value.

As we can see in Table 3, while cases that are present in many languages (NOMINATIVE, ACCUSATIVE, GENITIVE) are quite recognisable, it is definitely not obvious, especially when throwing on other less common cases such as loca-

tive. In fact, changing the parameter k does not lead to significantly better results. The more common cases are less recognisable with decreasing k , leading to a worse classification, and the less common cases are even more blurred when increasing k , since they are flooded in the total number of samples. Moreover, whatever the parameter, there are always samples from common core cases that are classified as other cases. It appears that the portion of space occupied by each case is neither fully distinct from the others, causing confusion when trying to cluster cases with the same names as well as limiting our ability to distinguish smaller cases from ones that take more space, nor is it well connected, given the fact some samples are always closer to other cases.

7 Adposition Annotation

As discussed in Section 2, some corpora in UD make use of the *Case* feature on adpositions and it is recommended by UD’s guidelines.

Given the postulate according to which all natural languages are equally expressive, one could indeed see case marking and the use of adpositions as two means of achieving the same linguistic goals. Two means that are by no mean exclusive since languages that use case tend to have a rather limited inventory and use adpositions to express a broader range of meanings and relations.

Following Kirov et al. (2017), we have applied the methods described above to represent certain adpositions and to give them a syntactically equivalent case representation. This could partially prove the postulate, as well as help justifying the way some corpora annotate adpositions for case.

To do so, we counted the dependency relations leading to the governors of each adposition. This gave us a distribution on syntactic usage of adpositions similar to a profile, and allowed us to compare adpositions to cases.

Table 4 represents the uniform means of the representations of a few French adpositions across all French corpora. As we can see, and could be predicted by French speakers, most adpositions are used in a similar way in French, mainly as LOCATIVES (*dans, par, sur, sous, vers...*) or INSTRUMENTALS/COMITATIVE (*avec*). For the other adpositions, we see that there is a non-negligible proportion of usage that leads to *advcl*. This comes from infinitive constructions marking goal (*pour*), intent (*à*), avoidance (*sans*) or gerundive construc-

Adpos	advcl	nmod	nsubj	obj	obl
À	16.7	17.3	0.04	0.38	63.4
DANS	0.46	13.8		0.19	78.7
PAR	0.26	13.7	0.10	0.18	74.6
POUR	29.5	15.9		0.02	41.2
EN	8.13	17.1		0.36	54.1
VERS	0.26	35.7			62.1
AVEC	0.61	32.4			62.6
DE	2.10	68.0	0.14	1.31	14.3
SANS	24.4	21.1		0.78	43.8
SOUS	0.21	22.9	0.02	72.8	
SUR	0.47	36.3		0.10	59.4
SAUF	10.7	22.6			38.1

Table 4: Dependency relation profiles of the governors irrespective of its part-of-speech of a few French adpositions.

tions marking manner (*en*).

This justifies the idea of giving a case to adpositions as a reasonable supposition, and confirms our postulate that adpositions replace some cases in language without cases (French actually has cases on personal pronouns; but not for any of the cases *replaced* by adpositions). We believe that this method could be extended to any other part of speech with adequate semantics and syntactic constructions.

8 Conclusion

In this paper, we have investigated the comparative-descriptive confusion that Haspelmath warned us about using Universal Dependency data. We have compared cases between different languages as is it was a commensurable descriptive category and seen that at least for some closely related languages the alignment stands at least for core cases. We then tried to represent archetypal cases as if case was a comparative concept applied onto each treebank, and saw that core cases mostly align with our expectations. However, this asks for a more principled analysis of the use of the term *nominative* for the default case especially so when the nominative-accusative distinction does not exist or when it does not simply mark a syntactic role but also definiteness for example.

References

- Frédéric Chazal, Leonidas Guibas, Steve Oudot, and Primoz Skraba. 2011. Persistence-based clustering in riemannian manifolds. *Journal of the ACM*, 60.
- Martin Haspelmath. 2018. *How comparative concepts and descriptive linguistic categories are different*, pages 83–114.
- Santiago Herrera, Caio Corro, and Sylvain Kahane. 2024. Sparse logistic regression with high-order features for automatic grammar rule extraction from treebanks. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15114–15125, Torino, Italia. ELRA and ICCL.
- Christo Kirov, John Sylak-Glassman, Rebecca Knowles, Ryan Cotterell, and Matt Post. 2017. A rich morphological tagger for english: Exploring the cross-linguistic tradeoff between morphology and syntax. pages 112–117.
- Vincent Kríž and Barbora Hladká. 2018. Czech legal text treebank 2.0. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.
- Clément Maria, Jean-Daniel Boissonnat, Marc Glisse, and Mariette Yvinec. 2014. The gudhi library: Simplicial complexes and persistent homology. In *Mathematical Software – ICMS 2014*, pages 167–174, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.
- Daniel Zeman et al. 2024. Universal dependencies 2.14. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

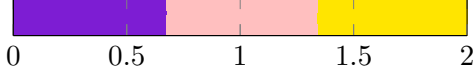
Appendix

List of the dependency relations used for the x -axis in 4:

```

_ ; acl ; advcl ; advmod ; amod ;
appos ; aux ; case ; cc ; ccomp ;
clf ; compound ; conj ; cop ; csubj ;
dep ; det ; discourse ; dislocated ;
expl ; fixed ; flat ; iobj ; list ;
mark ; nmod ; nsubj ; nummod ; obj ;
obl ; orphan ; parataxis ; punct ;
reparandum ; root ; vocative ; xcomp

```



Cs \ Ru							
	Acc	Dat	Gen	Ins	Loc	Nom	Par
Acc	0.3	0.45	0.49	0.26	0.51	0.41	1.03
Dat	0.49	0.44	0.55	0.38	0.46	0.51	0.92
Gen	0.5	0.4	0.2	0.32	0.47	0.52	1.05
Ins	0.43	0.39	0.47	0.28	0.42	0.44	0.95
Loc	0.53	0.48	0.54	0.43	0.5	0.55	0.96
Nom	0.49	0.55	0.59	0.38	0.59	0.25	1.11

Table 5: Distances between Czech CLTT and Russian GSD case profiles.

Cs \ Ru							
	Acc	Dat	Gen	Ins	Loc	Nom	Par
Acc	0.25	0.63	0.7	0.42	0.78	0.78	1.03
Dat	0.67	0.36	0.64	0.46	0.44	0.83	0.72
Gen	0.92	0.63	0.09	0.64	0.82	1.01	1.17
Ins	0.65	0.42	0.63	0.41	0.55	0.75	0.82
Loc	0.66	0.25	0.49	0.41	0.36	0.84	0.71
Nom	0.81	0.82	0.93	0.68	0.96	0.18	1.16

Table 6: Distances between Czech CLTT and Russian GSD noun case profiles.

	Acc	Dat	Gen	Ins	Loc	Nom
Acc	0	0.32	0.37	0.26	0.35	0.38
Dat	0.32	0	0.39	0.17	0.15	0.51
Gen	0.37	0.39	0	0.32	0.38	0.49
Ins	0.26	0.17	0.32	0	0.26	0.41
Loc	0.35	0.15	0.38	0.26	0	0.56
Nom	0.38	0.51	0.49	0.41	0.56	0

Table 7: Distances between Czech CLTT case profiles.

	Acc	Dat	Gen	Ins	Loc	Nom
Acc	0	0.61	0.76	0.55	0.6	0.71
Dat	0.61	0	0.65	0.18	0.28	0.81
Gen	0.76	0.65	0	0.66	0.5	0.98
Ins	0.55	0.18	0.66	0	0.33	0.74
Loc	0.6	0.28	0.5	0.33	0	0.82
Nom	0.71	0.81	0.98	0.74	0.82	0

Table 8: Distances between Czech CLTT noun case profiles.

	Acc	Dat	Gen	Ins	Loc	Nom	Par
Acc	0	0.44	0.55	0.32	0.46	0.51	0.92
Dat	0.44	0	0.44	0.3	0.27	0.53	0.77
Gen	0.55	0.44	0	0.39	0.51	0.58	1.07
Ins	0.32	0.3	0.39	0	0.39	0.39	0.94
Loc	0.46	0.27	0.51	0.39	0	0.59	0.6
Nom	0.51	0.53	0.58	0.39	0.59	0	1.07
Par	0.92	0.77	1.07	0.94	0.6	1.07	0

Table 9: Distances between Russian GSD case profiles.

	Acc	Dat	Gen	Ins	Loc	Nom	Par
Acc	0	0.65	0.87	0.5	0.72	0.87	0.9
Dat	0.65	0	0.62	0.39	0.34	0.84	0.64
Gen	0.87	0.62	0	0.59	0.82	0.96	1.17
Ins	0.5	0.39	0.59	0	0.61	0.72	0.89
Loc	0.72	0.34	0.82	0.61	0	0.97	0.35
Nom	0.87	0.84	0.96	0.72	0.97	0	1.16
Par	0.9	0.64	1.17	0.89	0.35	1.16	0

Table 10: Distances between Russian GSD noun case profiles.

	Abl	Acc	Dat	Equ	Gen	Ins	Loc	Nom
Acc	0.69	0.62	0.66	0.54	0.74	0.74	0.76	0.43
Dat	0.62	0.83	0.6	0.51	0.78	0.67	0.67	0.52
Gen	0.74	0.86	0.71	0.6	0.81	0.78	0.8	0.57
Ins	0.63	0.81	0.61	0.47	0.76	0.69	0.68	0.49
Loc	0.66	0.85	0.64	0.56	0.81	0.71	0.72	0.57
Nom	0.8	0.81	0.77	0.6	0.73	0.84	0.84	0.43

Table 11: Distances between Czech CLTT and Turkish BOUN case profiles.

	Abl	Acc	Dat	Equ	Gen	Ins	Loc	Nom
Acc	0.86	0.49	0.75	1.14	0.88	0.82	0.91	0.53
Dat	0.57	1.06	0.5	1.16	0.92	0.55	0.6	0.64
Gen	1.03	1.22	0.96	1.3	1.09	1	1.07	0.89
Ins	0.66	1.01	0.57	1.1	0.85	0.64	0.69	0.56
Loc	0.56	1.07	0.49	1.16	0.93	0.53	0.6	0.65
Nom	1.01	0.98	0.91	1.17	0.75	0.99	1.04	0.46

Table 12: Distances between Czech CLTT and Turkish BOUN nouns case profiles.

	Abl	Acc	Dat	Equ	Gen	Ins	Loc	Nom
Abl	0	0.93	0.11	0.4	0.87	0.14	0.12	0.65
Acc	0.93	0	0.87	0.88	0.94	0.96	1.01	0.68
Dat	0.11	0.87	0	0.41	0.88	0.14	0.16	0.63
Equ	0.4	0.88	0.41	0	0.83	0.48	0.44	0.57
Gen	0.87	0.94	0.88	0.83	0	0.94	0.96	0.41
Ins	0.14	0.96	0.14	0.48	0.94	0	0.11	0.71
Loc	0.12	1.01	0.16	0.44	0.96	0.11	0	0.73
Nom	0.65	0.68	0.63	0.57	0.41	0.71	0.73	0

Table 13: Distances between Turkish BOUN case profiles.

Investigating the effectiveness of Data Augmentation and Contrastive Learning for Named Entity Recognition

Noel Chia

University of Mannheim
Germany

neraug@noelchia.com

Ines Rehbein

University of Mannheim
Germany

rehbein@uni-mannheim.de

Simone Paolo Ponzetto

University of Mannheim
Germany

ponzetto@uni-mannheim.de

Abstract

Data Augmentation (DA) and Contrastive Learning (CL) are widely used in NLP, but their potential for NER has not yet been investigated in detail. Existing work is mostly limited to zero- and few-shot scenarios where improvements over the baseline are easy to obtain. In this paper, we address this research gap by presenting a systematic evaluation of DA for NER on small, medium-sized and large datasets with coarse and fine-grained labels. We report results for a) DA only, b) DA in combination with supervised contrastive learning, and c) CL with transfer learning. Our results show that DA on its own fails to improve results over the baseline and that supervised CL works better on larger datasets while contrastive transfer learning (CTL) is beneficial if the target dataset is very small. Finally, we investigate how contrastive learning affects the learned representations, based on dimensionality reduction and visualisation techniques, and show that CL mostly helps to separate named entities (NEs) from non-entities.

1 Introduction

Named Entity Recognition (NER) has been widely studied in NLP and has many applications in the computational social sciences and the digital humanities. Many of these applications, however, require the adaptation to new languages or genres for which no or only small amounts of annotated data are available. A major disadvantage of supervised NER systems is their dependence on large and representative datasets for training (Li et al., 2022b). Consequently, the scarcity of labelled data has become one of the major challenges impeding the performance of NER systems, especially in highly specialised domains.

Data Augmentation (DA) seems like a compelling solution to address this problem. By applying transformations to the data, new training instances can be generated, thus reducing the amount of manually annotated data needed to train the model (Perez and Wang, 2017). Many studies have applied DA to text classification tasks, summarisation, or question answering (Li et al., 2022a; Pellicer et al., 2023), with a focus on low-resource scenarios. We are not aware of any studies that report improved results for DA over strong baselines, such as transformers, for medium to large data sizes.

Furthermore, there is a lack of research on DA for token-level tasks such as NER, where the integration of DA presents a unique challenge. Several DA techniques apply transformations directly to tokens, thus changing their contextual information. As a consequence, this process may inadvertently modify the associated entity labels, disrupting the correspondence between tokens and their intended NEs (Dai and Adel, 2020). This challenge underscores the necessity of developing augmentation strategies that preserve the entity labels while enhancing the diversity and robustness of the training data for improved NER model performance.

Another promising approach to improve model performance is contrastive learning (CL), where the model learns to position representations of instances from the same class closer together in the embedding space while representations for data points that belong to different classes are pushed further apart. CL can be used on its own but can also be combined with DA and transfer learning.

In the paper, we address the question of which of the techniques described are effective in improving results for NER on small, medium-sized and large datasets.¹ Our main contributions include:

- a systematic evaluation of DA, CL and transfer learning for NER,

¹Our source code is openly available at <https://codeberg.org/noelchia/NER-Aug>

- an adaptation of supervised contrastive learning for token-level tasks, and
- a visual analysis of the learned representations.

2 Related Work

2.1 Data Augmentation for NER

Only a few studies have applied DA techniques to NER, focussing mostly on low-resource settings. One possible reason for this is that DA reduces overfitting and thus improves the generalisability of the model. Since overfitting is most common and severe for small datasets, we can expect the greatest benefit of DA in this context.

Dai and Adel (2020) explore simple data augmentations such as label-wise token replacement, synonym replacement, mention replacement and shuffle the order of tokens within segments on data from the biomedical and materials science domain. Their transformer-based tagger obtains improvements only for small dataset sizes (≤ 500 instances) but not when training on the full data. Ding et al. (2020) introduce an approach dubbed DAGA where they generate training examples for NER and other token-level NLP tasks using language models. Instead of producing unlabelled text, they generate new labelled training examples.

Zhou et al. (2022) propose Masked Entity Language Modelling (MELM) where they train a language model to generate NEs, conditioned on a masked sentence with NE tags. The main difference between DAGA and MELM is that DAGA generates the entire sentence, while MELM uses pre-existing instances and only replaces existing NEs by masking them and generating a new entity of the same class. Both approaches have been evaluated in low-resource scenarios.

Instead of low-resource NER, Chen et al. (2021) focus on DA for cross-domain NER, using an approach that learns textual patterns and transforms the text from a high-resource to a low-resource domain. based on denoising reconstruction, de-transforming reconstruction and domain classification. Cai et al. (2023) leverage graph propagation to create new data points, based on the relationship between labelled data and unlabelled natural texts, and evaluate their method in low-resource and cross-domain settings.

Zhang et al. (2022) develop two data augmentation methods for a BART based generative NER model. Theirs is the only work we are aware of that

addresses the problem of DA in in-domain settings with medium and large data sizes.

2.2 Contrastive Learning for NER

Contrastive learning (CL) is a discriminative machine learning technique that aims to create similar representations for data points that belong to the same class while pushing samples from different classes further apart in distributional space (Kumar et al., 2022). CL can be used in (semi)-supervised and unsupervised settings and is very popular because it allows the application of self-supervised learning to tasks that were previously only possible in supervised environments (Le-Khac et al., 2020; Liu et al., 2023). However, only few papers apply CL to NER, and most of these focus on few-shot learning.

Huang et al. (2022) introduce COPNER, a method to create prototypical tokens that represent each class. During contrastive training, the token representing the class forms positive pairs with NE tokens from that class while class tokens paired with words from other classes are considered as negative pairs. He et al. (2023) use a similar idea to develop a template-free prompting method for few-shot NER. Using external knowledge like textual descriptions of entity types, they generate anchors to represent the entity type. These anchors are then appended to the end of the input sentence. The authors use CL to train the encoder to produce representations of words that are similar to the corresponding entity type.

Das et al. (2022) use contrastive learning to train a model dubbed CONTAINER, which models the distribution of token classes using Gaussian Embeddings. Tokens from the same class are considered as positive pairs, and all other valid pairs are assumed to be negative. Li et al. (2023) also use Gaussian embeddings, but add a cross-domain attention layer based on HaloNet (Vaswani et al., 2021). Si et al. (2022) propose Span-based Contrastive Learning with Retrieval Augmented Inference (SCL-RAI). Their model focusses on NEs that have been mislabelled as negative instances by the system.

All of the papers above either focus on few-shot scenarios or train their CL method on small data sizes of less than 5,000 instances.

3 Experimental Settings

The last section has shown that there is a severe lack of research regarding the effectiveness of DA and CL for NER in scenarios where ample training data is available. We address this gap by providing a systematic investigation of both techniques in different settings and comparing their impact in isolation and in combination with transfer learning.

Datasets We select three different-sized English datasets with coarse and fine-grained entity type distinctions. CoNLL 2003 (Tjong Kim Sang and De Meulder, 2003) is the smallest dataset with 14 thousand training examples consisting of 301 thousand tokens, encoding four NE types only (Person, Location, Organisation, Miscellaneous). The second dataset, OntoNotes Release 5.0 (Weischedel et al., 2013), is medium sized with 82 thousand instances, over 2 million tokens and encodes 18 different NE types. The largest dataset is Few-NERD (Ding et al., 2021) with more than 131 thousand sentences, 4.6 million tokens and 66 fine-grained NE types. The fine-grained NE types are further grouped into 8 coarse-grained NE types. We use the original train, dev and test splits for CoNLL 2003 and Few-NERD. The authors of OntoNotes Release 5.0 did not release the dataset with predefined train, dev and test splits, so the splits suggested in Pradhan et al. (2013) were used.

Baseline Model We chose RoBERTa (Liu et al., 2019) as our baseline model, as it yields competitive results at reasonable training costs. Our implementation uses the `RobertaForTokenClassification` architecture from the Huggingface Transformers library (Wolf et al., 2020) which adds one additional linear layer on top of RoBERTa.

3.1 Data Augmentation Methods

We adapt three common approaches to data augmentation for NER, namely round-trip translation, paraphrasing and masking.²

Round-Trip Translation Sennrich et al. (2016) proposed to augment monolingual training data with automatic backtranslations to increase the size of the data. Inspired by this, we performed round-trip translation, where we translate a sentence into another language and then back to the original

²More detailed information on the different DA techniques and settings, including the number of augmented instances for each method and dataset, are provided in appendix A.1.

language create a different sentence. We check the round-trip translated output by string matching every NE in the original sample to the augmented sample. If all NEs are found, then the entities are labelled based on the assumption that all string matches represent the same NE, and all other words are not NEs. The neural machine translation model chosen is No Language Left Behind (NLLB) (NLLB Team et al., 2022) and we use translations to/from German. We also experimented with French and Zulu, with very similar results.

For a task like NER that is sensitive to token-level changes, round-trip translation might result in missing or modified NE labels. Hence, checks are performed to ensure that all NE tokens are preserved before adding the augmented data to the training set (for details, see appendix A.1).

Paraphrasing We use T5 (Raffel et al., 2020) to generate paraphrases for our data (also see appendix A.2). The model has been fine-tuned by Vorobev and Kuznetsov (2023b) on the ChatGPT paraphrases dataset, which includes the Quora Question Pairs (QQP) (Iyer et al., 2017), the Stanford Question Answering Dataset (SQuAD) version 2.0 (Rajpurkar et al., 2018) and the CNN / DailyMail Dataset (Hermann et al., 2015). ChatGPT was used to create five paraphrases for each example in the three datasets to train the T5 model.

Masking Inspired by Shen et al. (2020), we randomly mask tokens to produce augmented data. Masking aims to reduce overfitting by forcing the model to learn to predict NEs even when the token or its context is masked. We add a consistency loss to the loss function to encourage the model to make similar predictions for both the original and masked instances (Eq. 1 below).

$$\mathcal{L} = \mathcal{L}_{ce}(\mathbf{x}, \mathbf{y}) + \mathcal{L}_{ce}(\mathbf{x}_{\text{masked}}, \mathbf{y}) + \mathcal{L}_{KL}(\mathbf{x}, \mathbf{x}_{\text{masked}}) \quad (1)$$

\mathcal{L}_{ce} denotes the cross-entropy loss, and \mathcal{L}_{KL} the Kullback-Leibler (KL) divergence loss. For each example \mathbf{x} with target labels \mathbf{y} , an augmented sample $\mathbf{x}_{\text{masked}}$ will be generated, where every token in $\mathbf{x}_{\text{masked}}$ will have a 15% probability of being replaced by a [MASK] token (also see appendix A.3 for more details).

3.2 Supervised Contrastive Learning for NER

Khosla et al. (2020) propose the supervised contrastive (SupCon) loss for computer vision, a supervised variation of contrastive learning that also

makes use of labelled images of the same class as additional positive pairs. This approach allows us to integrate contrastive learning into the downstream task, thus reducing the time requirements for task-specific fine-tuning after the CL step.

We adapt supervised contrastive learning for NER by considering each contextualised token embedding generated by RoBERTa as a training example and add two fully connected layers to the model. The objective of this training step is to maximise the similarity of the contextualised representations for tokens that belong to the same NE type, and to minimise the similarity otherwise. After the contrastive learning step, we add a new fully connected layer to the model and perform task-specific fine-tuning.

Adapting the SupCon loss for NER Tian et al. (2023) show that SupCon is similar to calculating the cross-entropy loss. Let $i \in I := \{1, 2, \dots, N\}$ be the index of a sample, and $a \in A(i) := I \setminus \{i\}$ be the index of a different sample. \mathbf{x}_i is a training example with its corresponding label y_i , and is mapped to projection \mathbf{z}_i by the contrastive model. $\tau \in \mathbb{R}^+$ is a scalar temperature variable. First, a contrastive categorical distribution \mathbf{q}_i is constructed to describe how closely \mathbf{z}_i matches \mathbf{z}_j for $j \in A(i)$ (see Eq. 2).

$$q_{i,j} = \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_j / \tau)}{\sum_{a \in A(i)} \exp(\mathbf{z}_i \cdot \mathbf{z}_a / \tau)} \quad (2)$$

If there is at least one element in $A(i)$, then the weighing term of the contrastive loss can be calculated similarly to the cross entropy ground-truth categorical distribution \mathbf{p}_i as shown in Eq. 3 where the indicator function $\mathbb{1}_{\text{match}}(i, j)$ indicates whether there is a match ($y_i = y_j$).

$$p_{i,j} = \frac{\mathbb{1}_{\text{match}}(i, j)}{\sum_{a \in A(i)} \mathbb{1}_{\text{match}}(i, a)} \quad (3)$$

The supervised contrastive loss is the cross entropy between the ground-truth distribution \mathbf{p}_i and the contrastive distribution \mathbf{q}_i , as shown in Eq. 4.

$$\mathcal{L} = \sum_{i \in I} H(\mathbf{p}_i, \mathbf{q}_i) = - \sum_{i \in I} \sum_{j \in J} p_{i,j} \log q_{i,j} \quad (4)$$

We implement the loss function in Eq. 4 for contrastive learning for NER. The projection head used for supervised learning consists of a hidden layer with ReLU activation before the final linear projection, as Chen et al. (2020) showed that this

performs better than the single linear projection layer used in some contrastive learning models.

3.2.1 Contrastive Learning with DA

We test combinations of DA and CL, using masking (see section 3.1). This augmentation was chosen because of its computational efficiency, requiring only a random number generator to select words for random masking. In contrast, round-trip translation and paraphrasing both require a separate model to generate the input, making it difficult to perform the augmentation during training.

3.2.2 Contrastive Transfer Learning

Experiments on contrastive learning in other domains, such as computer vision (Chen et al., 2020), suggest that the representations produced by CL tend to be highly adaptable across different tasks and domains. We will test the hypothesis that the representations produced by training on one NER dataset can be applied to another NER dataset to improve the model’s performance.

This could be useful for practical applications, especially for cases where only a small set of labelled data is available. By first performing contrastive learning on a larger dataset and then fine-tuning the learned representations on the smaller dataset, better performance could be achieved. This might be an alternative to data augmentation or could be used in combination with data augmentation to further improve results. To assess the effectiveness of CTL for NER and explore how different dataset properties affect the results, we test all six possible combinations of datasets.

4 Results

Data Augmentation We first look at the results for the three DA methods, i.e., round-trip translation, paraphrasing and masking (Table 1). All results are averaged over five runs, with the standard deviation (STDEV) shown in subscript. Statistically significant improvements over the baseline are underlined.

As shown in Table 1, the three data augmentation methods mostly fail to produce statistically significant improvements. Paraphrasing is the worst performer, often producing similar or sometimes even worse results than the baseline. One reason for this lack of improvement might be that the T5 model used for paraphrasing is trained on similar data as RoBERTa, so the paraphrased results represent a distribution of the data that RoBERTa has already seen during pre-training. Hence, the model

Dataset Size (Sentences)	100	500	1000	5000	Full
CoNLL-2003 (4 NE types)					
	Mean F1 Score% \pm STDEV				
Baseline	83.32 \pm 0.36	88.23 \pm 0.60	89.84 \pm 0.46	91.15 \pm 0.23	91.98 \pm 0.43
DA Translate	83.66 \pm 0.81	88.40 \pm 0.45	90.01 \pm 0.36	91.38 \pm 0.36	92.23 \pm 0.39
DA Paraphrase	83.37 \pm 0.65	88.21 \pm 0.47	89.81 \pm 0.52	91.22 \pm 0.32	92.19 \pm 0.61
DA Mask	80.83 \pm 0.38	88.47 \pm 0.33	<u>90.48</u> \pm 0.33	91.34 \pm 0.30	<u>92.47</u> \pm 0.37
CL	82.52 \pm 0.77	88.86 \pm 0.49	90.06 \pm 0.26	<u>91.53</u> \pm 0.30	<u>92.49</u> \pm 0.29
DA Mask + CL	80.90 \pm 0.58	88.12 \pm 0.59	89.95 \pm 0.39	<u>91.56</u> \pm 0.38	<u>92.20</u> \pm 0.26
OntoNotes v5 (18 NE types)					
	Mean F1 Score% \pm STDEV				
Baseline	66.01 \pm 1.58	77.90 \pm 0.55	82.21 \pm 0.32	85.58 \pm 0.42	89.28 \pm 0.25
DA Translate	66.03 \pm 0.73	77.37 \pm 0.50	81.63 \pm 0.53	85.24 \pm 0.41	89.21 \pm 0.33
DA Paraphrase	66.02 \pm 1.16	77.28 \pm 0.66	81.58 \pm 0.55	84.97 \pm 0.30	88.34 \pm 0.41
DA Mask	60.65 \pm 0.99	77.26 \pm 0.51	82.16 \pm 0.33	85.80 \pm 0.45	88.83 \pm 0.74
CL	65.82 \pm 0.79	<u>78.71</u> \pm 0.33	82.42 \pm 0.32	<u>86.51</u> \pm 0.46	<u>89.76</u> \pm 0.25
DA Mask + CL	65.89 \pm 1.52	<u>78.76</u> \pm 0.56	82.12 \pm 0.37	86.02 \pm 0.39	89.65 \pm 0.55
Few-NERD (66 NE types)					
	Mean F1 Score% \pm STDEV				
Baseline	38.77 \pm 0.82	54.07 \pm 0.89	58.17 \pm 0.67	62.09 \pm 0.41	67.90 \pm 0.59
DA Translate	38.22 \pm 1.69	54.27 \pm 0.37	57.93 \pm 0.38	62.42 \pm 0.41	67.95 \pm 0.70
DA Paraphrase	38.87 \pm 0.72	54.16 \pm 0.53	57.95 \pm 0.33	62.36 \pm 0.41	67.42 \pm 0.97
DA Mask	35.85 \pm 0.64	52.89 \pm 0.57	56.66 \pm 0.41	62.01 \pm 0.30	63.72 \pm 0.87
CL	38.46 \pm 0.70	54.93 \pm 0.63	<u>58.85</u> \pm 0.43	<u>63.04</u> \pm 0.34	<u>68.65</u> \pm 0.24
DA Mask + CL	36.84 \pm 0.86	53.15 \pm 0.45	<u>57.36</u> \pm 0.32	<u>62.37</u> \pm 0.56	<u>68.62</u> \pm 0.23

Table 1: Mean F1 scores over five runs for every data augmentation/contrastive training and dataset size combination. Underlined results show statistically significant increases over the baseline (Student’s t-test, $\alpha = 5\%$).

struggles to learn new generalisable information from the examples, and this is reflected in the lack of improvement in the results.

Round-trip translation performs slightly better, but the improvements are also not statistically significant. Both paraphrasing and round-trip translation generate augmentations with tokens that are not NEs as we apply string matching between the NEs in the original data and the augmented examples to ensure that the labels are still valid. This means that our augmentations provide the model with different contexts for known NEs but do not actually show the model new NEs. The lack of improvement raises the question whether a more successful approach would present the model with augmented data that includes new NEs. This, however, is difficult to perform automatically without the risk of changing the NE type.

Masking, on the other hand, can be applied to both NEs and context tokens. However, the results are mixed and do not allow us to draw reliable conclusions. While we see statistically significant improvements for the CoNLL data on the full dataset and on a sample of 1000 sentences, no significant improvements were obtained on the other sample sizes or on the OntoNotes and FewNERD data.

A possible explanation could be that while mask-

ing reduces the chances of overfitting, it also increases the difficulty of the task as the model now needs to guess the NE of the masked tokens. Therefore, the technique might be better suited to easier problems with a high risk of overfitting, such as datasets with fewer NE types like CoNLL with its four coarse NE classes.

Contrastive Learning CL shows the most consistent results. At dataset sizes of above 5,000, we see statistically significant improvements for all three datasets. While data augmentation methods tend to work better on smaller datasets, our results show that contrastive learning needs more data to be beneficial. Instead of providing the model with new instances, contrastive learning improves the representations produced by the model. To learn robust and generalisable representations, large datasets are necessary to avoid overfitting.

Combining DA and CL As both approaches seem complementary, we also test the combination of DA and CL, using masking for data augmentation (Table 1, Mask + CL). While the model occasionally produces statistically significant results, the improvements are rather small. This does not necessarily mean that combining contrastive learning with data augmentation does not work in general.

Dataset Size (Sentences)	100	500	1000	5000	Full
CoNLL-2003 (4 NE types)					
	Mean F1 Score% \pm STDEV				
Baseline	83.32 \pm 0.36	88.23 \pm 0.60	89.84 \pm 0.46	91.15 \pm 0.23	91.98 \pm 0.43
CL only	82.52 \pm 0.77	88.86 \pm 0.49	90.06 \pm 0.26	<u>91.53</u> \pm 0.30	<u>92.49</u> \pm 0.29
CTL + OntoNotes	83.75 \pm 1.23	87.91 \pm 0.44	89.43 \pm 0.21	<u>91.23</u> \pm 0.14	<u>92.19</u> \pm 0.32
CTL + Few-NERD (coarse)	<u>85.46</u> \pm 0.39	<u>89.23</u> \pm 0.39	90.16 \pm 0.20	91.39 \pm 0.21	92.35 \pm 0.36
CTL + Few-NERD (fine)	<u>85.26</u> \pm 0.65	<u>89.02</u> \pm 0.66	<u>90.67</u> \pm 0.21	<u>91.68</u> \pm 0.25	92.15 \pm 0.28
OntoNotes v5 (18 NE types)					
	Mean F1 Score% \pm STDEV				
Baseline	66.01 \pm 1.58	77.90 \pm 0.55	82.21 \pm 0.32	85.58 \pm 0.42	89.28 \pm 0.25
CL only	65.82 \pm 0.79	<u>78.71</u> \pm 0.33	82.42 \pm 0.32	<u>86.51</u> \pm 0.46	<u>89.76</u> \pm 0.25
CTL + CoNLL	65.73 \pm 1.67	<u>78.58</u> \pm 0.42	82.39 \pm 0.81	<u>85.71</u> \pm 0.27	<u>89.28</u> \pm 0.23
CTL + Few-NERD (coarse)	<u>68.47</u> \pm 2.44	<u>79.64</u> \pm 0.16	83.07 \pm 0.01	<u>86.14</u> \pm 0.44	<u>88.87</u> \pm 0.29
CTL + Few-NERD (fine)	<u>67.55</u> \pm 0.88	<u>79.66</u> \pm 0.63	<u>83.17</u> \pm 0.22	<u>86.23</u> \pm 0.55	89.48 \pm 0.27
Few-NERD (66 NE types)					
	Mean F1 Score% \pm STDEV				
Baseline	38.77 \pm 0.82	54.07 \pm 0.89	58.17 \pm 0.67	62.09 \pm 0.41	67.90 \pm 0.59
CL only	38.46 \pm 0.70	54.93 \pm 0.63	<u>58.85</u> \pm 0.43	63.04 \pm 0.34	<u>68.65</u> \pm 0.24
CTL + CoNLL	36.34 \pm 0.61	54.45 \pm 0.30	58.43 \pm 0.41	62.45 \pm 0.20	68.26 \pm 0.28
CTL + OntoNotes	37.79 \pm 1.58	54.79 \pm 0.67	58.32 \pm 0.34	62.57 \pm 0.24	68.33 \pm 0.21

Table 2: Mean F1 scores over five runs with and without contrastive training for different dataset sizes. The underlined results are statistically significant increases over the baseline ($\alpha = 5\%$). Few-NERD (coarse) uses the 8 coarse-grained labels, Few-NERD (fine) refers to the 66 fine-grained NE types.

More work is needed to explore data augmentations for NER to answer that question.

Contrastive Transfer Learning Table 2 shows results for CTL for all possible dataset combinations. We observe two clear trends. First, CTL works better when the *target data* is small. This is not surprising, given that there is more room for improvement when the baseline is low. The second observation is that CL needs sufficiently large *source data* to work well. This also makes sense as a larger transfer learning dataset allows the model to learn more useful representations of the data for the downstream task.

To investigate the impact of the *number of entity types* in the contrastive training set, we report results for two different settings. In the first setting, we use the eight coarse-grained NE types in Few-NERD that have some overlap with the entity inventory in CoNLL and OntoNotes,³ the second setting includes Few-NERD’s 66 fine-grained NE types.

Results show that the coarse-grained entity labels only yield statistically significant improvements when the target training data is small (500 or less sentences) but fail to improve results for larger fine-tuning datasets with 1,000 or more sentences. This indicates that CL has learned more useful representations from the fine-grained information

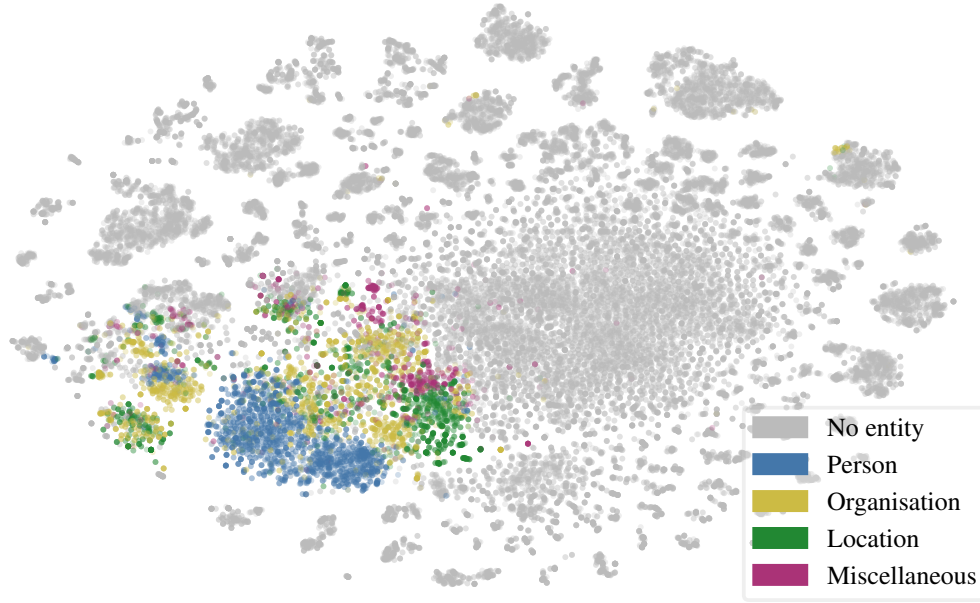
³The coarse-grained entity types are PERSON, LOCATION, ORGANIZATION, ART, BUILDING, PRODUCT, EVENT, MISCELLANEOUS.

in the transfer data which is somewhat surprising, given that the coarse-grained entity types overlap with the labels in the respective target data.

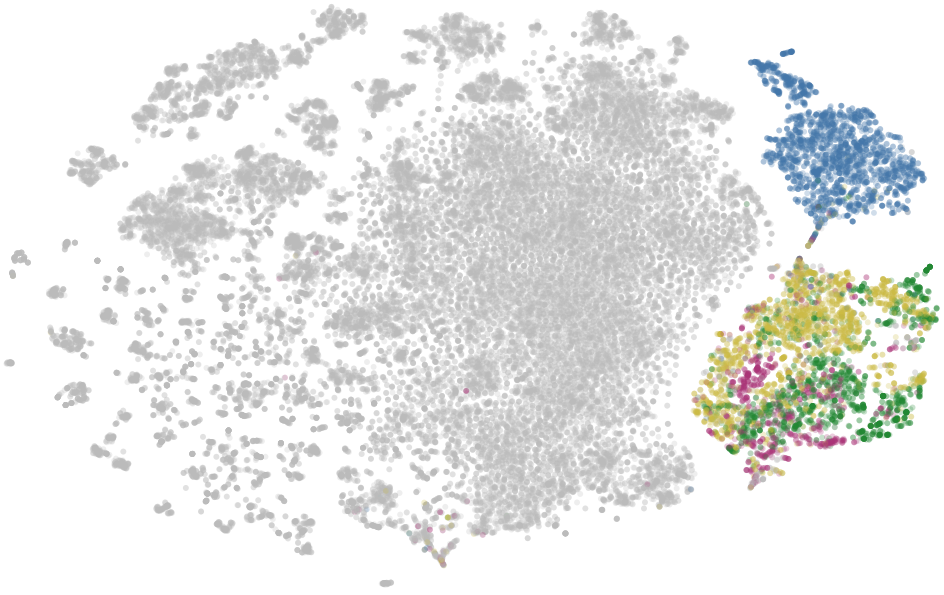
5 Analysis and Visualisation

To better understand the effect of CL, we visualise the learned representations before and after the CL step. As we cannot directly plot the 768-dimensional word embeddings produced by RoBERTa on a two-dimensional graph, we apply dimensionality reduction techniques in order to obtain informative two-dimensional representations.

A popular dimensionality reduction technique is principal component analysis (PCA), which tries to reduce the dimensionality by choosing the linear combination of variables that explain the variance in the data (Jolliffe and Cadima, 2016). While PCA is quite efficient, it can only be applied when all components are linear. A method that can perform non-linear dimensionality reduction is t-SNE (van der Maaten and Hinton, 2008). However, t-SNE still has a high computational cost compared to PCA, especially when dealing with large datasets of high dimensionality. To resolve this problem, we use a combination of PCA and t-SNE for dimensionality reduction. We first apply PCA to reduce the dimensionality of the word embeddings from 768 to 50. Then, t-SNE is used to reduce the dimensions from 50 to two (see appendix B for more details).



(a) Contextual token embeddings before contrastive training.



(b) Contextual token embeddings after contrastive training.

Figure 1: Visualisation of CoNLL 2003 token embeddings using a combination of PCA and t-SNE for dimensionality reduction.

Figure 1 shows the visualisations of the embeddings for the CoNLL dataset, using CL only. Results for OntoNotes and FewNERD are similar, and can be found in appendix C. For all three datasets, the separation between non-entities and NEs is greater than the separation of the representations for neighbouring NE classes. While a possible reason for this could simply be that the difference between

NEs and non-entities is greater and therefore easier to learn, a more likely reason is the distribution of NEs and non-entities in the data where the latter significantly outnumber the former. In CL, this means that the model can minimise the loss by increasing the difference between the non-entities and NEs even if this comes at the expense of decreasing the difference between two different NE classes.

Hence, the lack of separation between the different NE classes can most probably be explained by the class imbalance in the data.

6 Discussion

While we found no evidence that the proposed data augmentations are effective, related work has shown that DA can be beneficial in low-resource scenarios (Dai and Adel, 2020; Ding et al., 2020; Cai et al., 2023). We also observed consistent increases in results for CL for datasets with sizes of at least 5,000 sentences. Our best results for a RoBERTa-base model with CL on OntoNotes (89.75% F1) are only slightly below the ones reported for much larger models (cf., 89.76% F1 for BART-large (Yan et al., 2021) and 90.42% F1 for a T5-base model with DA (Zhang et al., 2022)). These results are promising, given the severe lack of methods for improving the performance on larger datasets, as DA has only been successful when applied in low-resource and few-shot scenarios (Dai and Adel, 2020; Ding et al., 2020; Zhou et al., 2022; Cai et al., 2023), and the same also applies to work on contrastive learning for NER.

Our experiments failed to show that CL works for smaller datasets. However, when combined with transfer learning, the results are improved. CTL works best when the fine-tuning data size is small, making it a good complement to CL without transfer learning. Figure 2 summarises our results, showing which method might work best in different scenarios.

While the improvements we obtained are small, they are still important given that increasing the performance of a model that is already performing quite well tends to be much harder than improving the performance of a poorly performing model. In addition, data augmentation and CL can be combined, as often done in related fields like computer vision (Chen et al., 2020; He et al., 2020). This might be a promising avenue for future work on developing CL methods that work well for smaller datasets. Our experiments demonstrate that combining DA and CL is possible (see Mask + CL in Table 1) but might require more sophisticated data augmentation techniques to improve results.

Addressing Data Imbalance in CL for NER In section 5, we showed that a major problem for applying CL to NER is the data imbalance as the majority of the token labels are non-entities. One approach to address this problem could include a modifica-

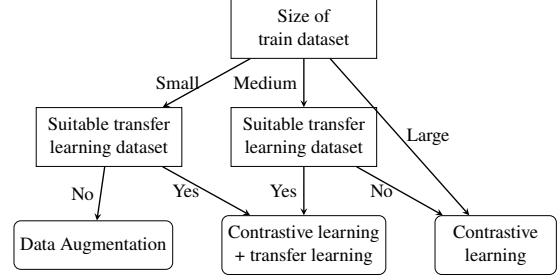


Figure 2: Recommendations for selecting the best approach for different-sized datasets.

tion of the CL loss function to account for the imbalance (Cao et al., 2019; Fernando and Tsokos, 2022; Wang et al., 2020; Rezaei-Dastjerdehei et al., 2020). Assuming that equation (4) is used for the loss function, a modified loss function that includes weights is shown in equation (5), where $w_i \in \mathbb{R}^+$ is the weight for class i .

$$\mathcal{L} = - \sum_{i \in I} w_i \sum_{j \in J} p_{i,j} \log q_{i,j} \quad (5)$$

There are many ways to set w_i , but one possibility is to set it to the ratio of the frequency of non-entities to the frequency of the class. This is shown in equation (6), where n_i is the frequency of class i and n_O is the frequency of the non-entity class.

$$w_i = \frac{n_O}{n_i} \quad (6)$$

This scales the loss function so that different classes can have different weights which might help encourage the model to differentiate between various types of NEs. There are many alternatives for the loss and weight functions, and the functions proposed above might not be optimal. Development and testing of a weighted loss function will be left to future work.

7 Conclusion

We presented a systematic investigation of the effect of DA, CL, and CTL for NER. Our main results can be summarised as follows. First, while DA has been shown to be effective in low-resource scenarios (specifically for pre-transformer-based taggers), we failed to demonstrate an improvement in results in our experiments. CL, on the other hand, can effectively improve results over a strong RoBERTa baseline when medium to large datasets are available for fine-tuning, but has a weaker performance on smaller datasets. For small dataset sizes, contrastive transfer learning is the most promising approach but requires the existence of suitable data for transfer learning.

We hope that the insights from our experiments will foster more work on DA and CL for NER especially for medium and large datasets. To address the problem of data imbalance for NER, where the majority of the labels are non-NEs, we proposed a modification to the loss function, which we plan to explore in future work.

Acknowledgments

The work presented in this paper was funded by the German Research Foundation (DFG) under the UNCOVER project (RE3536/3-1).

References

- Jiong Cai, Shen Huang, Yong Jiang, Zeqi Tan, Pengjun Xie, and Kewei Tu. 2023. Graph Propagation based Data Augmentation for Named Entity Recognition. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 110–118, Toronto, Canada. Association for Computational Linguistics.
- Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. 2019. Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.
- Shuguang Chen, Gustavo Aguilar, Leonardo Neves, and Tamar Solorio. 2021. Data Augmentation for Cross-Domain Named Entity Recognition. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5346–5356, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *ICML’20*, pages 1597–1607. JMLR.org.
- Xiang Dai and Heike Adel. 2020. An analysis of simple data augmentation for named entity recognition. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3861–3867, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Sarkar Snigdha Sarathi Das, Arzoo Katiyar, Rebecca Passonneau, and Rui Zhang. 2022. CONTAINER: Few-Shot Named Entity Recognition via Contrastive Learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6338–6353, Dublin, Ireland. Association for Computational Linguistics.
- Bosheng Ding, Linlin Liu, Lidong Bing, Canasai Kruengkrai, Thien Hai Nguyen, Shafiq Joty, Luo Si, and Chunyan Miao. 2020. DAGA: Data Augmentation with a Generation Approach for Low-resource Tagging Tasks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6045–6057, Online. Association for Computational Linguistics.
- Ning Ding, Guangwei Xu, Yulin Chen, Xiaobin Wang, Xu Han, Pengjun Xie, Haitao Zheng, and Zhiyuan Liu. 2021. Few-NERD: A Few-shot Named Entity Recognition Dataset. 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 3198–3213, Online. Association for Computational Linguistics.
- K. Ruwani M. Fernando and Chris P. Tsokos. 2022. Dynamically Weighted Balanced Loss: Class Imbalanced Learning and Confidence Calibration of Deep Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, 33(7):2940–2951.
- Kai He, Rui Mao, Yucheng Huang, Tieliang Gong, Chen Li, and Erik Cambria. 2023. Template-Free Prompting for Few-Shot Named Entity Recognition via Semantic-Enhanced Contrastive Learning. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–13.
- Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum Contrast for Unsupervised Visual Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738.
- Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching Machines to Read and Comprehend. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.
- Yucheng Huang, Kai He, Yige Wang, Xianli Zhang, Tieliang Gong, Rui Mao, and Chen Li. 2022. COP-NER: Contrastive Learning with Prompt Guiding for Few-shot Named Entity Recognition. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2515–2527, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.
- Shankar Iyer, Nikhil Dandekar, and Kornél Csernai. 2017. First Quora Dataset Release: Question Pairs. <https://quoradata.quora.com/First-Quora-Dataset-Release-Question-Pairs>.
- Ian T. Jolliffe and Jorge Cadima. 2016. Principal component analysis: A review and recent developments. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, 374(2065):20150202.

- Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS'20*, pages 18661–18673, Red Hook, NY, USA. Curran Associates Inc.
- Pranjal Kumar, Piyush Rawat, and Siddhartha Chauhan. 2022. Contrastive self-supervised learning: Review, progress, challenges and future research directions. *International Journal of Multimedia Information Retrieval*, 11(4):461–488.
- Phuc H. Le-Khac, Graham Healy, and Alan F. Smeaton. 2020. Contrastive Representation Learning: A Framework and Review. *IEEE Access*, 8:193907–193934.
- Bohan Li, Yutai Hou, and Wanxiang Che. 2022a. Data augmentation approaches in natural language processing: A survey. *AI Open*, 3:71–90.
- Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2022b. A Survey on Deep Learning for Named Entity Recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):50–70.
- Wei Li, Hui Li, Jingguo Ge, Lei Zhang, Liangxiong Li, and Bingzhen Wu. 2023. CDANER: Contrastive Learning with Cross-domain Attention for Few-shot Named Entity Recognition. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, Gold Coast, Australia. IEEE.
- Xiao Liu, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang. 2023. Self-Supervised Learning: Generative or Contrastive. *IEEE Transactions on Knowledge and Data Engineering*, 35(1):857–876.
- 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.
- NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Hefernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No Language Left Behind: Scaling Human-Centered Machine Translation.
- Lucas Francisco Amaral Orosco Pellicer, Taynan Maier Ferreira, and Anna Helena Real Costa. 2023. Data augmentation techniques in natural language processing. *Applied Soft Computing*, 132:109803.
- Luis Perez and Jason Wang. 2017. The Effectiveness of Data Augmentation in Image Classification using Deep Learning.
- Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards Robust Linguistic Analysis using OntoNotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 143–152, Sofia, Bulgaria. 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. *The Journal of Machine Learning Research*, 21(1):140:5485–140:5551.
- 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.
- Mohammad Reza Rezaei-Dastjerdehei, Amirmohammad Mijani, and Emad Fatemizadeh. 2020. Addressing Imbalance in Multi-Label Classification Using Weighted Cross Entropy Loss Function. In *2020 27th National and 5th International Iranian Conference on Biomedical Engineering (ICBME)*, pages 333–338.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.
- Dinghan Shen, Mingzhi Zheng, Yelong Shen, Yanru Qu, and Weizhu Chen. 2020. A Simple but Tough-to-Beat Data Augmentation Approach for Natural Language Understanding and Generation.
- Shuzheng Si, Shuang Zeng, Jiaxing Lin, and Baobao Chang. 2022. SCL-RAI: Span-based Contrastive Learning with Retrieval Augmented Inference for Unlabeled Entity Problem in NER. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2313–2318, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.
- Yonglong Tian, Lijie Fan, Phillip Isola, Huiwen Chang, and Dilip Krishnan. 2023. StableRep: Synthetic Images from Text-to-Image Models Make Strong Visual Representation Learners. In *Thirty-Seventh Conference on Neural Information Processing Systems*.
- Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition.

- In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.
- Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605.
- Ashish Vaswani, Prajit Ramachandran, Aravind Srinivas, Niki Parmar, Blake Hechtman, and Jonathon Shlens. 2021. Scaling Local Self-Attention for Parameter Efficient Visual Backbones. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12894–12904.
- Vladimir Vorobev and Maxim Kuznetsov. 2023a. ChatGPT paraphrases dataset. <https://huggingface.co/datasets/humarin/chatgpt-paraphrases>.
- Vladimir Vorobev and Maxim Kuznetsov. 2023b. A paraphrasing model based on ChatGPT paraphrases. https://huggingface.co/humarin/chatgpt_paraphraser_on_T5_base.
- Chen Wang, Chengyuan Deng, and Suzhen Wang. 2020. Imbalance-XGBoost: Leveraging weighted and focal losses for binary label-imbalanced classification with XGBoost. *Pattern Recognition Letters*, 136:190–197.
- Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, Mohammed El-Bachouti, Robert Belvin, and Ann Houston. 2013. OntoNotes Release 5.0.
- 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.
- Hang Yan, Tao Gui, Junqi Dai, Qipeng Guo, Zheng Zhang, and Xipeng Qiu. 2021. A unified generative framework for various NER subtasks. 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 5808–5822, Online. Association for Computational Linguistics.
- Shuai Zhang, Yongliang Shen, Zeqi Tan, Yiquan Wu, and Weiming Lu. 2022. De-Bias for Generative Extraction in Unified NER Task. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 808–818, Dublin, Ireland. Association for Computational Linguistics.
- Ran Zhou, Xin Li, Ruidan He, Lidong Bing, Erik Cambria, Luo Si, and Chunyan Miao. 2022. MELM: Data Augmentation with Masked Entity Language Modeling for Low-Resource NER. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2251–2262, Dublin, Ireland. Association for Computational Linguistics.

Appendices

A Details for Data Augmentation

Table 3 shows how the different DA techniques affect the size of the training data in our experiments. Please note that the dataset size for Masking and CL always remains constant.

A.1 Consistency Checks for Round-Trip Translation

We check the round-trip translated output by string matching every named entity in the original sample to the augmented sample. If all named entities are found, then the entities are labelled based on the assumption that all string matches represent the same named entity, and all other words are not named entities. The neural machine translation model chosen is No Language Left Behind (NLLB) (NLLB Team et al., 2022) and we use round-trip translation to/from German. We also experimented with French and Zulu, with very similar results.

A.2 Paraphrasing

The model used for paraphrasing is T5 (Raffel et al., 2020). To generate the augmented sentence, “paraphrase: ” is prepended to each original example and given to the T5 model as input. The model has been fine-tuned by Vorobev and Kuznetsov (Vorobev and Kuznetsov, 2023b) on the ChatGPT paraphrases dataset (Vorobev and Kuznetsov, 2023a), which uses the Quora Question Pairs (QQP) dataset (Iyer et al., 2017), Stanford Question Answering Dataset (SQuAD) version 2.0 (Rajpurkar et al., 2018) and the CNN/DailyMail Dataset (Hermann et al., 2015). ChatGPT was used to create five paraphrases for each example in the three datasets to train the T5 model.

A.3 Masking

The masking rate is selected based on the design of BERT, which uses the same masking rate for its mask language modelling training. However, this masking method is not exactly the same as that performed by BERT, which only replaces the

Dataset Size		Original	100	500	1000	5000	Full
<i>Round-Trip Translation via German</i>							
CoNLL-2003	(4 NE types)	14,041	158	785	1,587	7,966	22,348
OntoNotes	(18 NE types)	82,122	167	872	1,714	8,638	141,314
Few-NERD	(66 NE types)	131,767	165	837	1,689	8,394	219,969
<i>Paraphrasing</i>							
CoNLL-2003	(4 NE types)	14,041	177	898	1,801	9,051	25,310
OntoNotes	(18 NE types)	82,122	177	896	1,782	8,911	146,041
Few-NERD	(66 NE types)	131,767	173	909	1,791	9,056	238,500

Table 3: Number of augmented instances used for training for the different DA techniques (round-trip translation, paraphrasing) and dataset sizes (100, 500, 1,000, 5,000, full dataset).

chosen token with [MASK] 80% of the time. There is a 10% chance the token will be replaced by a random token and a remaining 10% chance the token will remain unchanged. This is not done because the initial tests show that replacing with the [MASK] token is already a complicated enough task, and the addition of random tokens might cause the model to perform slightly worse.

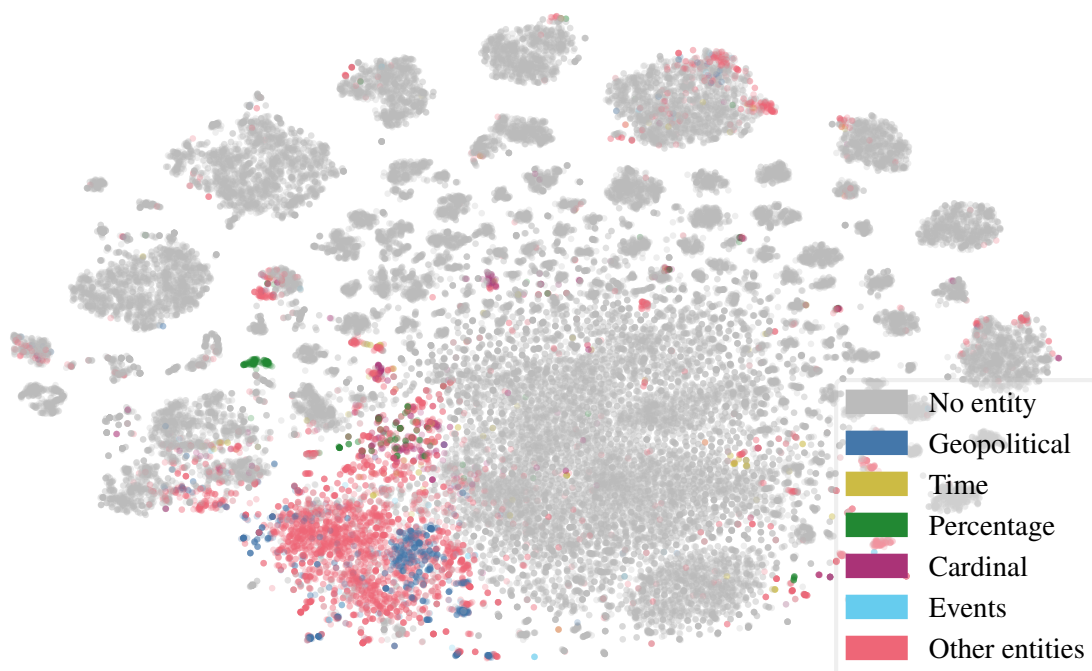
ensure that the best and worst-case scenarios are shown in the plot. The remaining three entity types are randomly selected to give a more representative picture of the rest of the entity types.

B Visualising Word Embeddings with PCA and t-SNE

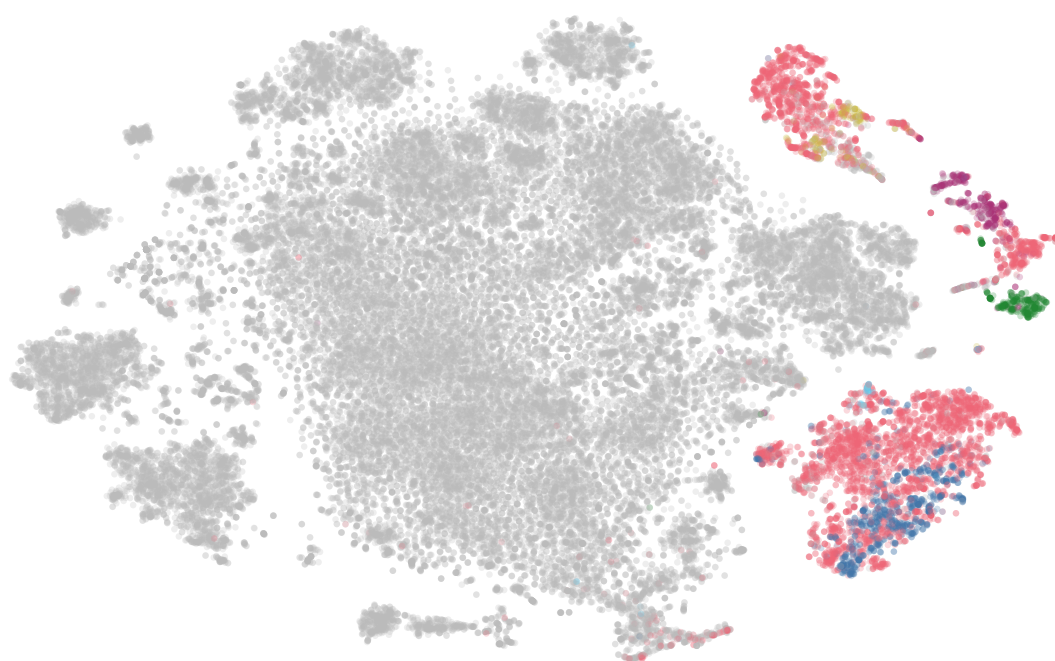
A problem faced when creating the scatter plots after dimensionality reduction is that every word in the test set becomes a point on the plot, so there is a huge number of points found on the plot. This causes the points in the plot to overlap and block each other, making the plot difficult to read. Increasing the transparency of the points and making them slightly smaller was sufficient to make the CoNLL 2003 plot readable. However, a random sample of 50,000 points needed to be taken from the OntoNotes v5 and Few-NERD test set, because these sets were much bigger. The sampling was only done right before plotting to avoid any information loss when performing PCA or t-SNE.

C Visualisations for OntoNotes and FewNERD

The OntoNotes v5 and Few-NERD datasets contain 18 and 66 entity classes respectively. This makes it impossible to find different colours that have good contrast for every entity class, and on the scatter plot, it is difficult to tell so many classes apart. To solve this problem, only five named entity types will have a unique colour, and the rest are grouped together as “other entities”. One of the five selected has a good F1 score after supervised fine-tuning, and another has very poor scores. These two classes

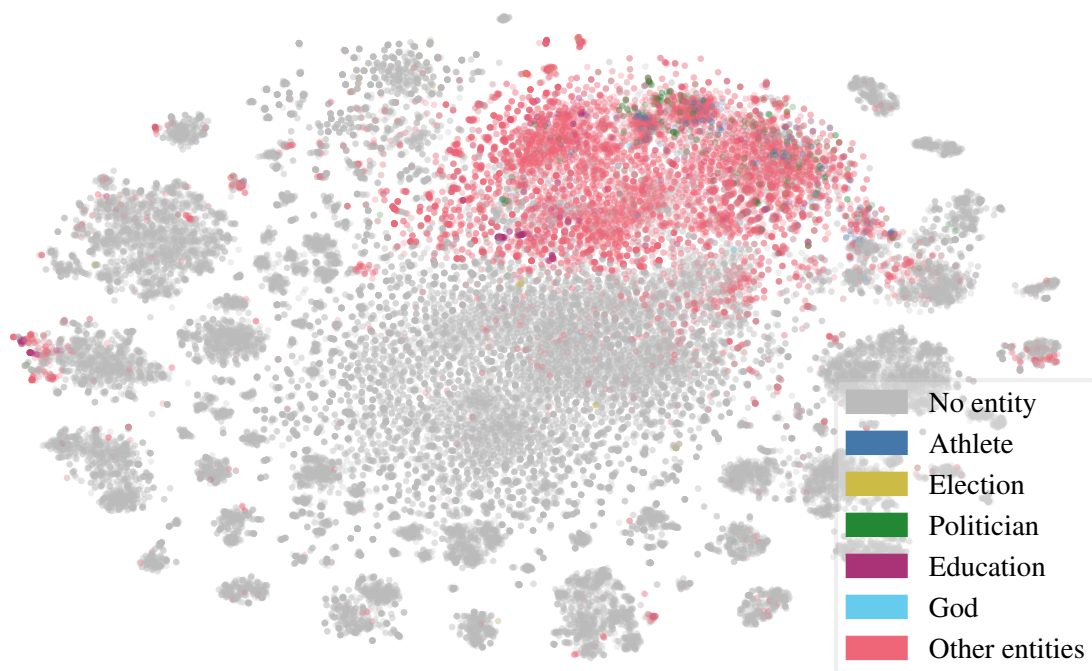


(a) Contextual token embeddings before contrastive training.

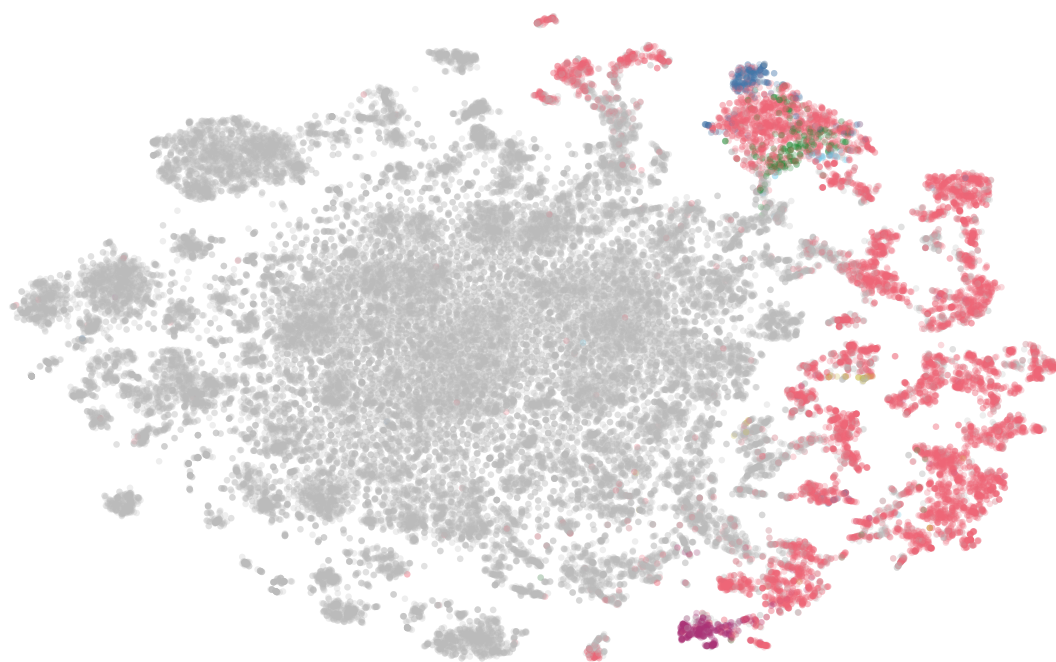


(b) Contextual token embeddings after contrastive training.

Figure 3: OntoNotes v5 token embeddings using a combination of PCA and t-SNE for dimensionality reduction.



(a) Contextual token embeddings before contrastive training.



(b) Contextual token embeddings after contrastive training.

Figure 4: Few-NERD token embeddings using a combination of PCA and t-SNE for dimensionality reduction.

Comparing Human and Machine Translations of Generative Language Model Evaluation Datasets

Sander Bijl de Vroe and **Georgios Stampoulidis** and **Kai Hakala** and **Aku Rouhe**
and **Mark van Heeswijk** and **Jussi Karlgren**

Advanced Micro Devices, Inc.

{Sander.BijldeVroe, Georgios.Stampoulidis, Kai.Hakala,
Aku.Rouhe, Mark.vanHeeswijk, Jussi.Karlgren}@amd.com

Abstract

The evaluation of Large Language Models (LLMs) is one of the crucial current challenges in the field of Natural Language Processing (NLP) and becomes even more challenging in the multilingual setting. Since the majority of the community’s benchmarks exist only in English, test sets are now being machine translated at scale into dozens of languages. This work explores the feasibility of that approach, comparing a Finnish machine translation (MT) of ARC-Challenge with a new human translated version. Our findings suggest that since absolute scores are fairly close and model size rankings are preserved, machine translation is adequate in this case. Surprisingly, however, the datasets reverse the order of base models compared to their chat-finetuned counterparts.

1 Introduction and Background

Generative Large Language Models (LLMs) have made significant progress in the past few years and their usefulness is being explored in many applications. This exploration is occurring world-wide, and as such there are many multilingual models available which have been trained with data in several languages simultaneously. However, a central challenge in building multilingual models is that access to quality data in languages except for the largest ones is limited, and this challenge crucially extends to evaluation datasets.

Our ability to train acceptably performing LLMs in new languages has far outpaced our abilities to create high-quality evaluations for those languages, in part because training can rely on transfer effects, where competence acquired in one language generalises to another language to some

extent (e.g., Gogoulou et al., 2021). Constructing new test sets in the language under consideration allows for controlling the quality as well as cultural validity of test items, but translating existing test sets (usually in English) to a target language involves less effort, less cost, and provides a basis to compare results across languages.

Translating entire test suites involves considerable human effort, so using automatic translation tools is an obviously attractive option. Given the immediate need to evaluate multilingual models, these machine translations of evaluation datasets have started proliferating — for example, Lai et al. (2023) automatically translate the popular benchmarks HellaSwag (Zellers et al., 2019), MMLU (Hendrycks et al., 2020) and ARC (Clark et al., 2018) into 26 languages.

However, these strategies carry a certain risk of systematic bias and introduced error into the test, and very little work has been done to verify that the resulting evaluations can be trusted. Besides actual translation errors, sometimes the objectives of the test are rendered moot by linguistic differences: for instance, tests that exploit structural ambiguities translate poorly from an isolating language to an agglutinative one.

This work investigates how automatically translated tests compare to manually translated tests. We study the case of ARC-Challenge, the challenging subset of ARC, a four-way multiple-choice task that has become a popular English LLM benchmark. We compare the performance of several Finnish LLMs on an automatically translated version (ARC-C-fi-MT) with a new manually translated version (ARC-C-fi-HT), which we release publicly. We find that in this task setup, LLMs perform comparably on the machine- and human-translated versions, so that machine translation may actually suffice in this case. One surprising caveat is that when considering model orderings, base models outperform their chat-tuned

counterparts on human data, while the chat-tuned models are stronger on machine translation data.

2 Datasets and Translation Methods

2.1 ARC-Challenge

Our investigation uses versions of ARC-Challenge, the more challenging portion of the ARC dataset (Clark et al., 2018), one of the most popular evaluation datasets for LLMs. ARC is a four-way multiple-choice Question Answering (QA) dataset, drawn from grade-school science questions designed for human test takers. For example, the question “*Which of these objects is translucent?*”, with choices “*A student’s notebook*”, “*A mirror on the bus*”, “*A brick wall of the school*”, “*A student’s sunglass lenses*”, would have the correct answer D). The Challenge portion of the corpus consists of only those questions that are answered incorrectly by an Information Retrieval (IR) system and a word co-occurrence system¹. We translate the test split, consisting of 1172 samples.

2.2 ARC-C-fi-MT

For the machine translated version of ARC-Challenge, we use the Finnish version released by LumiOpen (2024a), also containing translations into twelve other European languages. Samples were translated using the DeepL API (DeepL, 2025a) through the DeepL Python Library (DeepL, 2025b) using default parameters.

One noteworthy limitation is that answers were translated without the context provided by the question. This carries the drawback that some answers may have an altered meaning without the context or may contain unresolvable ambiguities, although in most cases answers are long enough for correct word sense disambiguation. For example, sample `Mercury_7086520` contains the choice “*be in the same period.*”, which carries a significantly different meaning in the context of the question “*Copper and gold have similar reactive properties. On the Periodic Table of the Elements, these elements are most likely to*”².

Example 1: Incorrect semantics

E: When making observations in nature, what is the best way for students to show respect for the environment?

F1: Miten opiskelijat voivat parhaiten huolehtia ympäristöstä ollessaan maastossa tekemässä havaintoja?

F2: Miten opiskelijat voivat parhaiten kunnioittaa ympäristöä ollessaan maastossa tekemässä havaintoja?

Example 2: Non-idiomatic translation

E: Which action would increase the amount of oxygen in a fish tank?

F1: Mikä toimi lisäisi hapen määrää akvaariossa?

F2: Mikä näistä lisäisi akvaariossa olevan hapen määrää?

Table 1: Examples: original English (E) initial inaccurate translations (*F1*) and revisions (*F2*).

2.3 ARC-C-fi-HT

Human translation data was acquired from a leading translation company that specializes in Nordic languages. The data was produced in two stages. The first version ARC-C-fi-HTv1 underwent a rigorous evaluation process by a native Finnish speaker with experience in translation and localization business. Surprisingly, a significant portion of the initial delivery was found to be of poor quality despite our guidelines. To improve translation quality, we provided detailed feedback and requested revision of the complete dataset, leading to an improved second version that we consider the gold standard, ARC-C-fi-HT.

Our feedback process focused on various difficulties, including sentence structure, semantic misinterpretations, and style. We ensured that cultural references were accurately preserved and additionally requested that a number of literal translations be corrected to more idiomatic Finnish expressions. Some indicative examples are found in Table 1. In Example 1, the first attempt *F1* uses the word *huolehtia*, which translates to *take care of*. The corrected version *F2* uses *kunnioittaa*, a more precise translation of *to show respect*. In Example 2, the inaccurate *F1* translates *action* as *toimi*, a more formal term that usually refers to actions by organizations. The revised version *F2* uses a more idiomatic phrasing (literally *which of these*), with the word for *action* omitted.

To ensure overall quality, we also established standards for capitalization, punctuation, dates, numbers, and names. The complete dataset, along

¹As in sample `VASoL_2009_5_30` mentioned above. Commonsense sentences like “A student’s sunglass lenses are translucent” are unlikely in corpora, so basic strategies are less successful.

²DeepL chooses the translation *samalla ajanjaksolla*, a different sense of *period*.

with ARC-C-fi-HTv1 and an alternate normalized version, is available here.

3 LLM Families

We evaluate three model families, Poro, Viking and Ahma, chosen because they are effectively the only LLM families trained especially for Finnish. Note that although FinGPT (Luukkonen et al., 2023) is absent, it can be viewed as a predecessor of Poro, since Poro uses an extended version of the same training data, and the models use an identical architecture.

3.1 Poro

The Poro base model (Luukkonen et al., 2024) is a 34 billion parameter decoder-only Transformer that uses the BLOOM architecture (Le Scao et al., 2023). It was trained on 1T tokens, of which 54.5% was English, 31.7% program code, 13.0% Finnish, and 0.8% English-Finnish translation pairs.

Poro 34B Chat (Silogen, 2024; LumiOpen, 2024b) is a version of Poro 34B trained to follow instructions in both English and Finnish using full-parameter supervised finetuning. The instruction data consists of roughly 40% English, 40% Finnish, and 20% cross-lingual examples. Because such data is not readily available in Finnish, Poro 34B itself was used to translate English instruction data into Finnish.

3.2 Viking

The Viking family of models (SiloAI, 2024) is another open-source model family that covers Finnish. The models are trained on 2T tokens, which includes further Nordic languages in Danish, Icelandic, Norwegian and Swedish, along with program code. Viking uses a similar architecture as Llama 2 (Touvron et al., 2023b). In this work we experiment with the 7B and 13B variants, for which the finetuned versions are not yet released.

3.3 Ahma

The Ahma model family (Tanskanen and Toivonen, 2024) is the only family of LLMs pre-trained exclusively on Finnish data. They consist of decoder-only transformer models based on Meta’s first Llama architecture (Touvron et al., 2023a). We evaluate both Ahma-7B and Ahma-3B, as well as Ahma-3B-Instruct. Note that the 7B-Instruct

Human Translation		Machine Translation	
1: Poro 34B	.414	Poro 34B-C	.391
2: Poro 34B-C	.397	Poro 34B	.369
3: Viking 13B	.387	Viking 13B	.329
4: Viking 7B	.363	Ahma-7B	.327
5: Ahma-7B	.358	Viking 7B	.326
6: Ahma-3B	.324	Ahma-3B-I	.310
7: Ahma-3B-I	.323	Ahma-3B	.307

Table 2: Model rankings for ARC-C-fi-HT and ARC-C-fi-MT (acc_norm scores).

version is not available at the time of writing. Ahma-3B is trained for 139B tokens, while Ahma-7B was trained for 149B tokens, on a varied collection of deduplicated and detoxified Finnish text sources.

4 Methodology

We use EleutherAI’s LM-evaluation-harness (Gao et al., 2024) to run the evaluations. For each of the datasets, we use the default parameters of English ARC-Challenge. In particular, we evaluate using the `multiple_choice` setting, with `doc_to_text`: “Question: {{question}}\nAnswer:” and `num_fewshot` = 0. This means that answers are obtained using logprobs instead of running inference. For each possible choice, the logprob of the choice text given `doc_to_text` is computed, and the model’s answer is the maximum logprob choice.

We compute both `acc` (accuracy) and `acc_norm`. The latter metric computes accuracy when logprobs are normalized by answer length, so that longer answers are not deemed less likely only due to their length. To guarantee that results are tokenizer-agnostic, normalization is performed using number of characters rather than, for example, number of tokens.

5 Experimental Results and Analysis

One straightforward way of analyzing the translated datasets is through comparing absolute scores per model. However, the datasets can also be compared in terms of whether they preserve the ranking between two sets of evaluated models. The model rankings (Table 2) reveal two main effects. Firstly, the ordering between model sizes remains effectively constant between the different translations: clearly scores follow the expected ordering $34B > 13B > 7B > 3B$. The Ahma and

		HT	MT	EN
Poro 34B-C	acc_norm	.397	.391	.485
	acc	.376	.374	.452
Poro 34B	acc_norm	.414	.369	.462
	acc	.361	.341	.424
Viking 13B	acc_norm	.387	.329	.402
	acc	.346	.312	.359
Viking 7B	acc_norm	.363	.326	.366
	acc	.308	.301	.340
Ahma-7B	acc_norm	.358	.327	.275
	acc	.302	.276	.248
Ahma-3B-I	acc_norm	.323	.310	.250
	acc	.290	.259	.220
Ahma-3B	acc_norm	.324	.307	.255
	acc	.278	.270	.195

Table 3: Results across ARC Challenge variants with Human Translation (HT), Machine Translation (MT) and English (EN)

Viking 7B models show similar performance on Finnish — it is unlikely their change in ranking between datasets is significant.

Secondly, however, for both Poro 34B and Ahma-3B, we see a change in ordering of the base model and chat model variants (note that for other base models the finetuned versions are not yet available). The base completion models rank higher when evaluated using human translation, while the chat models rank higher for the machine translated version.

Table 3 shows full results on the three dataset versions. Here we include the accuracy results next to acc_norm for completeness. A clear initial result is that across the board for every model family, size and training method (as well as for acc and acc_norm) the absolute performance on human translated data is at least slightly higher than on machine translated data.

Still, the size of these differences varies per model. One result worth noting is that the chat models perform similarly between the two datasets in absolute terms (.397 and .391 for Poro 34B Chat; .323 and .310 for Ahma-3B-Instruct). However, and particularly for Poro, there is a larger difference for the base models (.414 and .369 for Poro 34B; .324 and .307 for Ahma 3B). Thus the chat models seem more robust across the datasets, but at a cost to performance (.369 for Poro 34B Base < .391 for Poro 34B Chat on the machine translation condition).

We also find that the Poro and Viking models, trained on both English and Finnish, perform better on the English dataset than on the Finnish datasets. This is unsurprising given that in the case of Poro, there were more than four times as many English tokens in the training distribution. The Ahma models, lacking English training data, reach the expected performance of around 25% on English given 4-way multiple-choice.

6 Discussion

We propose a technique to find particularly interesting examples by 1) filtering for cases where the model is correct on one dataset and incorrect on the other and 2) sorting by the difference in log-probs on the prediction for the correct answer. In this way, we find samples where the difference in translation has the greatest effect on model prediction and performance.

This reveals some clear mistakes in machine translation. For instance, in Mercury_SC_414274 the correct choice is *The Moon is covered with many craters*. Here the human translation is *Kuun pinnalla on paljon kraattereita* whereas machine translation outputs *Kuu on monien kraatterien peitossa*, a more literal and less fluent translation. As a result it is only on the human translation data that Poro-Chat-34B manages to select the correct answer (with a logprob of -10.2 instead of -36.0). Question Mercury_7165218 about geology provides a more egregious error, where the choice *rift* is left untranslated as *rift*.

There are also cases, however, where flaws in machine translation actually increase model scores. In question Mercury_SC_406710 about chameleons, the choice *hunt for food* is translated correctly by humans as *Saalistaa ruokansa*, using the verb reserved for predators, but is machine translated as *metsästää ruokaa*, using the verb for human hunting³. However, perhaps since *metsästää* is more common, Poro-34B-Chat correctly chooses this as the answer, whereas it fails to do so for *saalistaa*. Thus the human translation reveals the incomplete semantics of the model in this case, while the machine translation does not.

A similar case occurs in MCAS_2014_8_6, where Poro-34B-Chat makes the correct choice only for the machine translated answer con-

³Note the unresolvable ambiguity for the MT model in this case, given that it has not seen the context *chameleon*.

taining the phrase *tektonisten laattojen* (tectonic plates). The human translation uses the phrase *litosfäärilaattojen* (lithospheric plates), which is heavily discounted by the model at a logprob of -73.0. Both translations are correct, but *litosfäärilaattojen* is a slightly more scientific and technical term in Finnish. *tektonisten* (of tectonics) is perhaps more common in layman’s language, which would explain both its generation by the MT system and its higher logprob in the LLM’s answer. In such cases, machine translated evaluations assign an inflated accuracy to the models, which should be able to respond positively to both the common and rarer scientific terms. For an agglutinative case-based language such as Finnish, similar cases would be possible when a human translator chooses a more accurate but less common grammatical case.

There are many future research avenues here. One option is to further investigate the chat and completion model reordering. This is possibly explained by an alignment of the fine-tuning training data with machine translation data — in both cases, models are trained using curated sentence pairs (whereas base model pre-training data consists of large chunks of text from massive corpora that tend to be less curated). Perhaps, then, fine-tuning a base model pulls it in the direction of the machine translation model distribution. Future work that compares more pairs of base and chat models, along with extended logprob analyses of both models types, may elucidate the picture.

Future work will also investigate the complex set of benefits and drawbacks of human translation. Human subjectivity and inconsistencies in judgment may introduce bias, and from a practical standpoint manual reviews can be time-consuming and expensive. One concrete direction is to compare the gold standard to ARC-C-fi-HTv1 and versions with alternative choices of normalization. It would also be worthwhile to explore alternate MT solutions, especially ones in which the models have access to the question as context when translating the answers.

7 Conclusion

Following the recent trend to machine translate English evaluation datasets at scale, this work compares a new human translation of ARC-Challenge into Finnish with a machine translated version. Our results indicate that for Finnish ARC-

Challenge, the machine translated dataset rivals the usefulness of the HT dataset for comparative evaluation of LLMs.

This is observed through the small absolute differences between scores (with models performing slightly more favorably on human translations as expected), as well as through the preservation of ordering of model sizes. One interesting caveat is that while chat-finetuned models outperform base models on machine-translated evaluation data, base models actually outperform their chat-finetuned counterparts on the human translated data, warranting further investigation.

Thus although there are drawbacks to using machine translation, especially for literature or other longer-form data, this work reveals that for comparative evaluation of Finnish language models on short multiple-choice questions, MT is sufficient. Future work can continue to reveal distributions of evaluation data, language translation pairs and model classes where this holds. It is clear that the intersection of translation and LLM evaluation provides unique challenges and opportunities that now deserve more attention than ever.

Acknowledgments

We thank Jonathan Burdge, Elaine Zosa, and Teppo Lindberg for their helpful comments. We also thank Sanna Piha for valuable contributions in dataset acquisition and review, and the AMD technical reviewers and copy-editors for their insights and advice. This work has been supported by the European Commission through the DeployAI project (grant number 101146490). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or DeployAI. Neither the European Union nor DeployAI can be held responsible for them. AMD is a trademark of Advanced Micro Devices, Inc. Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

References

- Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge.
- DeepL. 2025a. Deepl. <https://github.com/>

- DeepLcom/deepl-python. Accessed 2nd January 2025.
- DeepL. 2025b. Deepl python library. <https://www.deepl.com/en/translator>. Accessed 2nd January 2025.
- Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac’h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailley Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024. A framework for few-shot language model evaluation.
- Evangelia Gogoulou, Ariel Ekgren, Tim Isbister, and Magnus Sahlgren. 2021. Cross-lingual transfer of monolingual models. *arXiv preprint arXiv:2109.07348*.
- 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*.
- Viet Lai, Chien Nguyen, Nghia Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 318–327, Singapore. Association for Computational Linguistics.
- Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2023. Bloom: A 176b-parameter open-access multilingual language model.
- LumiOpen. 2024a. Lumiopen/arc_challenge_mt. Accessed on September 10th 2024.
- LumiOpen. 2024b. Poro-34b-chat. Accessed on May 28th 2024.
- Risto Luukkonen, Jonathan Burdge, Elaine Zosa, Aarne Talman, Ville Komulainen, Väinö Hatanpää, Peter Sarlin, and Sampo Pyysalo. 2024. Poro 34b and the blessing of multilinguality.
- Risto Luukkonen, Ville Komulainen, Jouni Luoma, Anni Eskelinen, Jenna Kanerva, Hanna-Mari Kupari, Filip Ginter, Veronika Laippala, Niklas Muennighoff, Aleksandra Piktus, Thomas Wang, Nouamane Tazi, Teven Scao, Thomas Wolf, Osmo Suominen, Samuli Sairanen, Mikko Merioksa, Jyrki Heinonen, Aija Vahtola, Samuel Antao, and Sampo Pyysalo. 2023. FinGPT: Large generative models for a small language. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2710–2726, Singapore. Association for Computational Linguistics.
- SiloAI. 2024. Viking 7b: The first open llm for the nordic languages. Accessed on October 7th 2024.
- Silogen. 2024. Poro 34b chat is here. Accessed on May 28th 2024.
- Aapo Tanskanen and Rasmus Toivanen. 2024. Ahma-3b (revision 0b51e96).
- 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*.
- Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

GliLem: Leveraging GliNER for Contextualized Lemmatization in Estonian

Aleksei Dorkin

Institute of Computer Science
University of Tartu
aleksei.dorkin@ut.ee

Kairit Sirts

Institute of Computer Science
University of Tartu
kairit.sirts@ut.ee

Abstract

We present GliLem—a novel hybrid lemmatization system for Estonian that enhances the highly accurate rule-based morphological analyzer Vabamorf with an external disambiguation module based on GliNER—an open vocabulary NER model that is able to match text spans with text labels in natural language. We leverage the flexibility of a pre-trained GliNER model to improve the lemmatization accuracy of Vabamorf by 10% compared to its original disambiguation module and achieve an improvement over the token classification-based baseline. To measure the impact of improvements in lemmatization accuracy on the information retrieval downstream task, we first created an information retrieval dataset for Estonian by automatically translating the DBpedia-Entity dataset from English. We benchmark several token normalization approaches, including lemmatization, on the created dataset using the BM25 algorithm. We observe a substantial improvement in IR metrics when using lemmatization over simplistic stemming. The benefits of improving lemma disambiguation accuracy manifest in small but consistent improvement in the IR recall measure, especially in the setting of high k .¹

1 Introduction

Lemmatization plays an important role in natural language processing by reducing words to their base or dictionary forms, known as lemmas. This process is especially crucial for morphologically rich languages such as Estonian, where words can

exhibit a multitude of inflected forms. Effective lemmatization enhances various downstream NLP tasks, including information retrieval based on lexical search and text analysis. Although dense vector retrieval is gaining traction in information retrieval, lexical search methods remain highly relevant, particularly in modern hybrid systems. Lexical search excels as a first-stage retriever due to its efficiency with inverted indices, and provides reliable exact term matching that dense retrievers may miss (Gao et al., 2021). Recent research demonstrates that lexical and dense retrieval are complementary, lexical matching providing a strong foundation for precise word-level matches, while dense retrieval captures semantic relationships and handles vocabulary mismatches. The complementary nature of these approaches has led to state-of-the-art hybrid systems that outperform either method alone (Lee et al., 2023).

Vabamorf (Kaalep and Vaino, 2001) is a rule-based morphological analyzer for the Estonian language. It provides one or more morphological analysis (including lemma) candidates for each token in a text, where the token can be a word or a punctuation mark. The Vabamorf’s analyzer functionality aims to generate all possible morphological analysis and lemma candidates for each word, regardless of its context. However, in order to find the appropriate analysis together with the lemma in a particular textual context, the analyzer output needs to be disambiguated. Vabamorf employs a built-in Hidden Markov Model (HMM) based disambiguator that can only look at the word’s immediate context to rank the analysis candidates by their likelihood scores. Thus, despite its high precision in generating lemma candidates, Vabamorf’s ability to disambiguate these candidates in context is limited due to its weak representational power.

Previously, Dorkin and Sirts (2023) have shown that, when evaluated on the Estonian Universal

¹A demo of the system is available at <https://huggingface.co/spaces/adorkin/GliLem>

Dependencies corpus, Vabamorf’s disambiguation abilities reach to ca 89% for lemmatization. However, when evaluated in the oracle mode, where a prediction is considered correct if the true lemma appears among the candidates, it achieves an accuracy above 99%.² This significant difference highlights the limitations of the Vabamorf’s current disambiguator and underscores the need for improving its disambiguation component.

Recent methods to neural lemmatization generally follow two approaches: pattern-based token classification (Straka, 2018; Straka et al., 2019) and generative modeling (Kanerva et al., 2018, 2021). The pattern-based approach predicts for each word a transformation pattern that can be used to transform the word token into corresponding lemma. When built on top of contemporary BERT-based encoders, the pattern-based lemmatizer makes use of the contextual token representations directly to make the prediction. The generative approach uses a character-based sequence-to-sequence model to generate the lemma conditioned on the word, relying on disambiguated morphological information as context. While both of these approaches have shown good results on Estonian (Dorkin and Sirts, 2023), neither of them is well suited for developing a new disambiguator for Vabamorf. First, the pattern-based token classification approach operates with a limited pattern vocabulary extracted from a training set and cannot handle previously unseen patterns that may be output by Vabamorf. Secondly, the generative model already assumes the presence of disambiguated morphological analyses making the disambiguation problem circular.

Recently, an open vocabulary model GliNER for Named Entity Recognition (NER) was proposed by Zaratiana et al. (2024) which can be used to match arbitrary text labels with input text spans. In the lemmatizer disambiguation setting, the GliNER approach can be used to match the transformation patterns extracted from Vabamorf analysis candidates to the spans of sub-word tokens making up words in the text, making it suitable for scoring a limited number of lemma candidates for each word.

Our first aim in this paper is to investigate

²For instance, if Vabamorf outputs three distinct lemma candidates for a given token, the oracle considers the prediction correct if one of these candidates is correct. This approach is unusable in a practical scenario, because the predictions have to be disambiguated.

whether GliNER method can be used to disambiguate the Vabamorf’s lemma candidates. For that, we modify the GliNER implementation to predict the transformation patterns of Vabamorf’s generated lemma candidates, using the Estonian Universal Dependencies corpus (Zeman et al., 2023) for training. We find that using this approach boosts the disambiguation accuracy from the HMMs 89% to 97.7%, significantly narrowing the gap between the disambiguator and the oracle.

Our second research question examines the impact of the improved lemma disambiguation accuracy on a downstream information retrieval (IR) task. Due to the lack of suitable Estonian datasets, we first translate the English DBpedia-entity dataset (Hasibi et al., 2017) into Estonian, employing the NLLB translation model (NLLB Team et al., 2022). We compare the performance of stemming, Vabamorf HMM-disambiguated lemmatization, and Vabamorf GliNER-disambiguated lemmatization in a BM25 retrieval setup. The results indicate ca 10% improvement in retrieval metrics when using Vabamorf lemmatization over stemming, with an additional 1% gain achieved through GliNER-enhanced disambiguation.

Overall, our contributions in this paper are threefold:

1. We implement a new neural disambiguator based on an open-vocabulary span-labeling method for the Estonian rule-based morphological analyzer Vabamorf (henceforth referred to as GliLem) and show that it considerably improves the lemmatization results over the existing HMM-based disambiguator.
2. We produce and release the first IR dataset for Estonian by machine translating the English DBpedia-entity dataset.³
3. We demonstrate the efficacy of the proper lemmatization over stemming for the IR task in Estonian, showing also that improved disambiguation translates into up to 1% improvement in the IR metrics.

2 Vabamorf and GliNER

In this section, we first give an overview of both the Estonian morphological analyzer Vabamorf and the open-vocabulary NER model GliNER.

³<https://huggingface.co/datasets/adorkin/dbpedia-entity-est>

2.1 Vabamorf

Vabamorf (Kaalep and Vaino, 2001) is a comprehensive, rule-based morphological analyzer specifically developed for the Estonian language. It leverages extensive morphological rules to generate all possible morphological analyses, including lemma candidates, for each analyzed word token. The analyzer accounts for the rich inflectional patterns of Estonian, which include numerous cases, tenses, and degrees of comparison.

Because many Estonian words can have several morphological analyses, Vabamorf includes a built-in HMM-based disambiguator, which aims to rank these candidates based on the contextual likelihood. However, under the HMM formulation, the disambiguation context is very limited, with the analysis of the current word only being dependent on the analysis of the previous word. Therefore, the performance of the HMM-based disambiguator is more than 10% lower than the oracle accuracy that can be obtained on the Estonian UD dataset (Dorkin and Sirts, 2023). We used Vabamorf via EstNLTK, which is a library that provides an API to various Estonian language technology tools (Orasmaa et al., 2016).

2.2 GliNER

GliNER (Zaratiana et al., 2024) is an open-vocabulary Named Entity Recognition (NER) model that extends traditional NER capabilities by allowing the labels to be specified in natural language (as opposed to nominal labels represented as integer indices in traditional classification models). Unlike conventional NER models that rely on a fixed set of entity types, GliNER can handle an arbitrary number of labels, making it highly adaptable for tasks requiring flexible label sets.

GliNER is based on an encoder-only BERT-like architecture, which is expanded with span representation and entity representation modules (see Figure 1). The modules are used to produce span and entity embeddings, accordingly. Span and entity embeddings are then used to measure pairwise similarity to identify entities in the input text. Entity types are expressed in natural language and separated from each other with the special [ENT] token. The entity types and input text are separated from each other with a [SEP] token, and they are processed in the model simultaneously in a cross-encoder fashion.

To implement GliNER, Zaratiana et al. (2024)

take an existing pre-trained encoder model as a basis for both the span and entity representation modules, and add two blocks of feed-forward layers on top of the encoder to process the spans and entities separately. Finally, entities are assigned to spans by scoring the similarities between the output representations from both the span and entity modules.

GliNER was pretrained on Pile-NER⁴ (Zhou et al., 2023), which is a synthetically annotated large scale NER dataset derived from the Pile corpus (Gao et al., 2020) that has ca 13K distinct entity types. Such pretraining is expected to give the GliNER model an ability to generalize to very different types of labels.

3 Adapting GliNER for Vabamorf Lemma Disambiguator

We observe that the GliNER architecture is flexible enough to be used for essentially any kind of token classification task, including part-of-speech tagging and morphological analysis. To be applicable for lemmatization, we adopt the approach proposed by Straka (2018) that expresses each example of *form* \rightarrow *lemma* as a transformation rule. Each transformation rule comprises the minimal sequence of character-level edits—commonly referred to as a shortest edit script—such as adding, removing, or replacing characters, required to transform a given *form* into its *lemma*. The transformation rules are represented simply as string labels, which are then used in token classification. For specific examples of transformation rules refer to Table 1.

While in theory it would be possible to use lemmas directly as “entities” to be scored instead of transformation rules, that would inflate the number of “entity types” to be learned considerably. Effectively, each token type would have to have its own lemmatization label. Meanwhile, the transformation rules proposed by Straka (2018) are abstract enough to allow for compact representation of similar transformations, and, according to Toporkov and Agerri (2024a), they offer stronger generalization than alternative approaches to shortest edit script generation. For instance, some common rules, such as “do nothing” and “upper case the first character”, are easily applicable to any surface form.

⁴<https://huggingface.co/datasets/Universal-NER/Pile-NER-type>

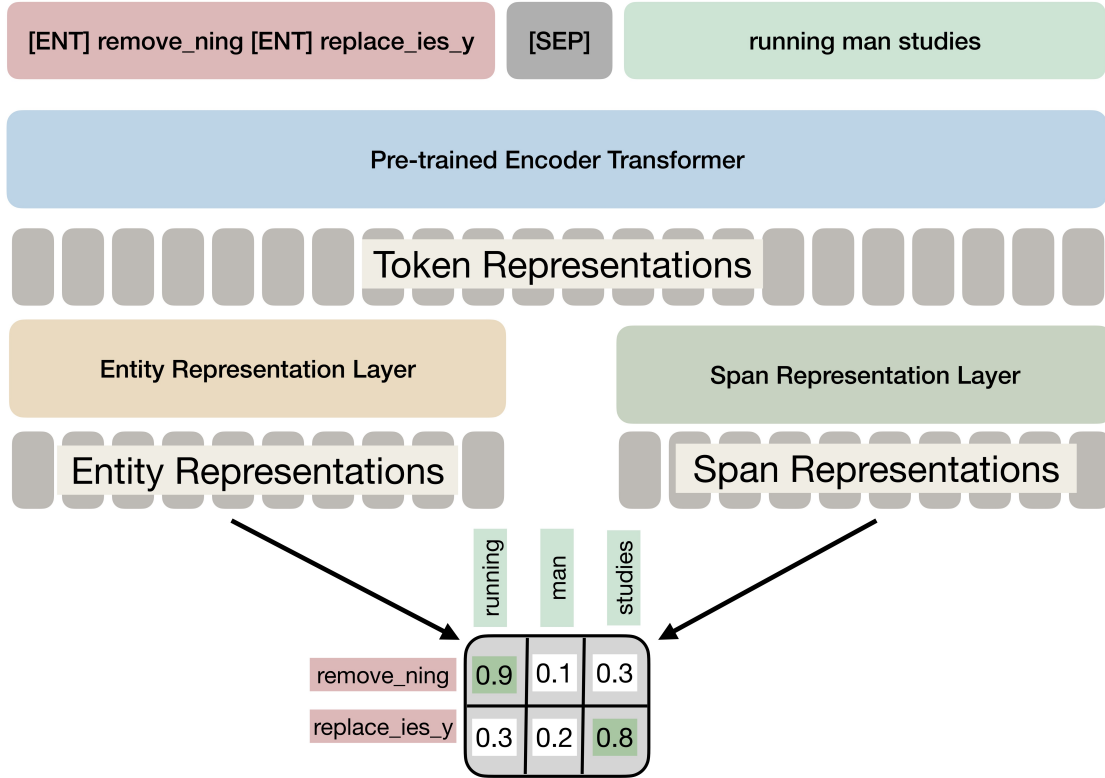


Figure 1: Schematic representation of the GliNER architecture applied to lemmatization.

The total number of unique transformation rules relevant for Estonian is too large to be used as input to GliLem. However, we only aim to score and rank the lemma candidates that the rule-based Vabamorf outputs for each token in the text. This limits the total number of possible “entities” to be scored by the number of tokens in the text, but usually it is much lower than that, mainly because the “do nothing” rule is the most common by far even in morphologically rich Estonian (see Table 1). According to Toporkov and Agerri (2024b), the “do nothing” rule is also the most common rule in diverse languages such as Basque, English, Russian, and Spanish. In Estonian, major contributors to the frequency of this rule, in addition to punctuation marks, include conjunctions, adverbs, some types of adjectives, postpositions, and inflected forms homonymous with the base form.

For each input token Vabamorf outputs at least one morphological analysis (together with lemma). Accordingly, to prepare Vabamorf outputs for disambiguation, a transformation rule is found for each token and all of its lemma candidates. The set of strings representing the obtained unique transformation rules are given as entities in

the GliLem input. GliLem outputs a list of spans, each accompanied by a proposed matching transformation rule and its score. The obtained rules are then applied to the respective spans to get the lemmas. The overall GliLem architecture, i.e., the GliNER architecture applied to lemmatization disambiguation, is schematically represented in Figure 1.

4 Enhancing Disambiguation with GliLem

We implement the GliLem for disambiguating lemma candidates generated by Vabamorf. To assess the effectiveness of the GliLem approach we evaluate the following approaches:

1. Vabamorf lemmatization using the built-in HMM-based disambiguator;
2. Vabamorf lemmatization in the Oracle mode (the prediction is considered correct if the correct lemma is in the proposed non-disambiguated candidates);
3. Pattern-based token classification model for lemmatization;

%	Rule	Description
49.6	$\downarrow 0; d \downarrow$	Do nothing
7.0	$\downarrow 0; d \downarrow -$	Remove the last letter
4.8	$\downarrow 0; d \downarrow --$	Remove two last letters
3.7	$\uparrow 0 \downarrow \downarrow 1; d \downarrow$	Upper case the first letter
3.3	$\downarrow 0; d \downarrow -+m+a$	Replace the last letter with <i>ma</i>
3.2	$\downarrow 0; d \downarrow ---$	Remove three last letters

Table 1: Top 6 most common transformation rules present in the train split of the EDT dataset.

4. Vabamorf lemma candidates disambiguated with GliLem.

4.1 GliLem training

We conducted experiments using the Estonian Universal Dependencies EDT corpus version 2.14, using the pre-defined splits. During training we do not make use of Vabamorf. Instead, we convert the token/lemma pairs provided in the corpus into respective lemmatization labels (transformation rules) and format the data according to the GliNER convention.

GliNER annotation schema differs from the BIO scheme typically used in the NER task. In GliNER, entire token spans with corresponding labels are used as inputs, and more importantly for our case, non-entities, i.e., the most common “default” class, are not labeled. Correspondingly, we do not use the “do nothing” rule as a label, and instead consider it the default state of the token. That means, we only score cases where the lemma is different from the surface form.

For training the GliLem, we use the GliNER training script provided by the authors⁵ using the default parameters and our lemmatization data to train the multilingual version of the pretrained model. The reason for using this model as a base model instead of initializing span and entity modules from scratch is that we expect to benefit from multilingualism of the backbone encoder, and also from the learned span representations of the NER model itself.

4.2 Token Classification Baseline

To contextualize the effect of Vabamorf disambiguation with GliLem, we reproduce the experiments by Dorkin and Sirts (2023) with some differences. We reduce the amount of preprocessing applied to the dataset: we do not lowercase the

data and do not remove the derivational symbols present in some lemmas. We also use the more recent UD version ($2.10 \rightarrow 2.14$).

The token classification model is a simple, efficient, and computationally cheap baseline to offset the complexity of the GliNER-based approach. For that reason we do not directly reproduce the token classification approach of Dorkin and Sirts (2023), but rather use the adapter-based parameter efficient fine-tuning (Houlsby et al., 2019), which reduces the training time down to minutes.

We do not reproduce the results of the generative character-level transformer model (Wu et al., 2021) that Dorkin and Sirts (2023) reported as the highest scoring approach, because it requires additional morphological annotation as input. Essentially, it needs the data to be disambiguated first which is contradictory to our goals in this work.

4.3 Results

The lemmatization results are shown in Table 2. The GliLem model achieves the lemmatization accuracy of 97.7% on the test, which significantly outperforms Vabamorf’s disambiguator that scores only 89.2% on the same set, demonstrating the efficiency of a more advanced disambiguation approach.

The pattern-based token classification model that does not utilize Vabamorf’s candidates reached 96.2% accuracy. While the difference with the GliLem disambiguation is modest (only ca 1.2% in absolute), it suggests that leveraging Vabamorf’s morphological analysis combined with GliLem’s disambiguation capabilities provides a performance advantage. Moreover, lemmatization accuracy scores are skewed towards higher values due to the majority of corpus tokens requiring no changes to transform the initial word form into the lemma, and that is generally not very difficult for any model to learn and predict.

In the Oracle mode, Vabamorf achieves an accuracy over 99%, showing that the disambiguator module has still room for improvement. However, the gap with the GliLem is less than 2% in absolute that can be hard to close.

5 Impact on Information Retrieval

The problem of lemmatization is usually evaluated in isolation, separately from an actual application. While lemmatization can be a useful step in some

⁵<https://github.com/urchade/GliNER/blob/main/train.py>

Method	Dev	Test
Vabamorf	0.878 [0.877, 0.883]	0.892 [0.889, 0.895]
Oracle Vabamorf	0.992 [0.992, 0.993]	0.993 [0.992, 0.994]
Pattern-based Token Classification	0.962 [0.960, 0.964]	0.966 [0.964, 0.968]
GliLem	0.974 [0.973, 0.976]	0.977 [0.975, 0.978]

Table 2: Bootstrap estimates of the lemmatization accuracy on the Estonian UD EDT dev and test sets with 95% confidence intervals. Oracle Vabamorf considers the prediction correct if the correct lemma appears in non-disambiguated Vabamorf predictions, thus making it unusable in a practical scenario where no labels are available.

realistic scenarios, the impact of the improvement in the lemmatization accuracy on the improvement of the downstream task can be difficult to estimate. To emulate the realistic scenario, we evaluate both the original Vabamorf disambiguator and the GliLem disambiguator in an information retrieval (IR) task. While the IR task is nowadays often addressed with dense vector retrieval, hybrid methods that, as a first step, adopt lexical search methods are still highly relevant. Input normalization via lemmatization is also more important in morphologically complex languages that typically have less resources than English. In particular, there is currently no IR benchmark dataset available in Estonian that would allow to evaluate the effect of different text normalization methods to the IR task. The only previous work in Estonian related to information retrieval that we are aware of is by Dorkin and Sirts (2024). However, this work addressed the problem of retrieving dictionary words based on their definitions using dense IR methods and did not deploy hybrid methods necessitating lexical normalization in the first steps. For this reason, we first translate an existing English information retrieval dataset to the Estonian language.

5.1 Dataset Preparation and Translation

DBpedia-Entity v2 (Hasibi et al., 2017) is a test collection for entity search evaluation, consisting of 467 queries with graded relevance judgments for entities from the DBpedia 2015-10 dump. In this work, we refer to the test collection together with the DBpedia dump as DBpedia-Entity. The collection comprises several distinct types of queries:

1. Short, ambiguous queries searching for one particular entity;
2. Information retrieval-style keyword queries;
3. Queries seeking a list of entities;
4. Natural language questions answerable by DBpedia entities.

For each query there is a list of a variable number of documents and their relevance judgments: highly relevant (2), relevant (1), irrelevant (0). Each document in the corpus represents an entity which has an ID, a title in natural language, and a variable length description. The dataset corpus—DBpedia 2015-10 dump—comprises approximately 4.5 million documents. We chose this dataset due to its general domain, the variety of query types it contains, and its focus on retrieving information from a very large collection of documents.

To evaluate the effect of lemmatization accuracy on information retrieval quality in Estonian, we translated the DBpedia-Entity dataset into Estonian using the NLLB (NLLB Team et al., 2022) translation model. We translated both documents and queries using the NLLB 3B,⁶ which is the largest available dense version of the NLLB. We adopted the CTranslate2⁷ library for efficient translation at large scale. Translating the entire dataset took approximately two days on a single A100 GPU on the University’s High Performance Cluster (University of Tartu, 2018).

At this time, we did not perform any quantitative quality evaluation of the resulting translations. Based on the manual examination of a small sample of examples, we note that while the translation quality is far from perfect, it generally pre-

1. Short, ambiguous queries searching for one particular entity;

⁶<https://huggingface.co/facebook/nllb-200-3.3B>

⁷<https://github.com/OpenNMT/CTranslate2>

serves the meaning well enough to be useful for our benchmark.

5.2 Retrieval Experiments

The BM25 algorithm (Robertson et al., 1995) is considered a strong information retrieval baseline to this day even when compared to modern dense retrieval models (Karpukhin et al., 2020; Thakur et al., 2021). BM25 relies on sparse lexical representations of documents and queries, with word-level tokens most commonly used for these representations. The tokens usually undergo additional preprocessing to account for surface form variation. For example, in sparse lexical representation, the present simple and the present participle forms of the word “run” (“run” and “running”) are considered entirely unrelated. That makes it difficult for the user to formulate queries because they have to guess in what form the desired term appears in the documents. For English, applying a stemming algorithm such as PorterStemmer is generally sufficient to deal with this problem.

Meanwhile, stemming algorithms do not perform well for morphologically rich languages like Estonian due to significant variation in stem surface forms in many words. This scenario highlights a practical application of lemmatization—improving the quality of lexical search in such languages. While it is intuitive to expect that lemmatization can help, there are no previous works showing that for the Estonian language. Moreover, it needs to be clarified what effect the additional lemmatization accuracy obtained from better disambiguation of Vabamorf outputs has on information retrieval.

For our experiments we used the recent BM25s library⁸ (Lù, 2024) that provides a fast implementation of BM25. For indexing, we used the default parameters and omitted the preprocessing done by the library—we input the corpus preprocessed by us directly.

We preprocessed the Estonian translation of the DBpedia-Entity corpus by applying the following four preprocessing approaches to the dataset documents:⁹

1. Identity (only tokenization is applied);

⁸<https://github.com/xhluca/bm25s>

⁹We exclude the token-classification baseline because we are interested in gauging the effect of improved lemmatization disambiguation on IR specifically.

2. Stemming using the Estonian Stemmer available in Apache Lucene;¹⁰
3. Vabamorf lemmatization with the built-in HMM disambiguation;
4. Vabamorf lemmatization with the GliLem disambiguation.

The output from each preprocessing resulted in each document being represented as a list of tokens, which were then concatenated with whitespace, the expected input format for BM25. The entire corpus was then passed to the indexer implementation. The indexing process took about three minutes, regardless of the preprocessing type.

Finally, we applied the same preprocessing options to the translated queries and, for each query, retrieved **100** most relevant documents from the corpus. Then, we employed relevance judgments from the original DBpedia-entity dataset to obtain the ground truth documents for each query (we selected only the documents deemed relevant or highly relevant) to calculate several retrieval metrics explained in the next section.

5.3 Evaluation Metrics

Success@k measures whether a user’s information need is satisfied by at least one result in the top **k** retrieved items (Karpukhin et al., 2020; Khatlab et al., 2021). It is a coarse-grained metric that does not distinguish how well the user’s information need was satisfied.

Recall@k measures what percentage of all relevant items for a query appear within the top **k** retrieved results (Buttcher et al., 2016). The metric is suitable for our case, because only a small proportion of the total number of documents is annotated with relevance judgments and therefore the Recall will be upper bounded only with very small **k** values.¹¹

Mean Average Precision (MAP)@k measures both the precision and ranking quality of the results up to position **k**, averaged across all queries.

¹⁰https://lucene.apache.org/core/8_11_0/analyzers-common/org/apache/lucene/analysis/et/EstonianAnalyzer.html

¹¹Consider for instance the case where there are 1000 relevant documents per query. In this case, for instance with **k** of 100, the Recall will be upper bounded by 0.1.

Metric	Baseline	Stemming	Vabamorf	GliLem
Recall@1	0.0269	0.0260	0.0218	0.0278
Recall@5	0.0633	0.0627	0.0702	0.0734
Recall@100	0.2212	0.2167	0.2831	0.2935
MAP@1	0.2077	0.2120	0.2527	0.2591
MAP@5	0.1201	0.1312	0.1596	0.1577
MAP@100	0.0874	0.0856	0.1057	0.1115
Success@1	0.2077	0.2120	0.2527	0.2591
Success@5	0.3704	0.4004	0.4925	0.4797
Success@100	0.6681	0.6767	0.7901	0.7837

Table 3: Retrieval metrics for the proposed token normalization approaches on the translated DBpedia-Entity dataset.

It captures not just whether relevant items were retrieved, but also how high they were ranked, giving more weight to relevant items appearing higher in the results (Buttcher et al., 2016). This metric prioritizes results that group relevant results closer to the top.

5.4 Results and Discussion

The IR performance measures at several k -s are shown in Table 3. First, we observe that the baseline of using word forms is on the same level with stemming on all measures, which is due to the Apache Lucene stemmer, although Estonian-specific, being very weak for Estonian.

When looking at the setting with k equal to 1, the Recall does not change considerably, but both MAP and Success rate (that are by definition equal in this setting) improve more than 4% when using lemmatization over stemming, although enhanced disambiguation with GliLem gives only a minor improvement over the default Vabamorf disambiguation.

In the k equal to 5 setting, Recall improves about 1%, MAP about 3%, and the Success rate, which is the most lenient measure, improves about 9%, when comparing Stemming to lemmatization with Vabamorf. In this setting, only the Recall measure shows a small positive impact of the more complex disambiguation with GliLem over the default Vabamorf disambiguation, while for the MAP and Success rate, the baseline Vabamorf gives better results.

Finally, in the k equal to 100 setting, when comparing lemmatization to stemming, Recall improves ca 7%, MAP about 2% and Success rate improves about 11%, with GliLem disambigua-

tion showing an additional improvement of ca 1% in both Recall and MAP over the Vabamorf default disambiguation.

We conclude that proper lemmatization can considerably improve IR results compared to stemming. At the same time, even large improvements in lemmatization accuracy, obtained by replacing the simple HMM-based disambiguation component with a more complex GliNER-based disambiguation do not easily translate into significantly better IR results. However, when comparing the baseline Vabamorf with the GliLem disambiguation, we observe a small but consistent improvements in Recall for all values of k , with the improvement being the most pronounced in the highest k setting. Using a high k is typical in hybrid IR systems, where the lexical retrieval is the first step to reduce the number of potentially relevant documents. Thus, the relatively small lemmatization improvement can have a positive effect in the downstream IR task.

Upon manual inspection of the original DBpedia-Entity corpus, we observed that it is somewhat noisy. Some entries have little to no content, while others are comprised of large listings. Many entries have characters from diverse writing systems. This results in additional noise introduced during the imperfect translation process. Moreover, there are translation errors in the translated queries (such as the presence of non-existent words), as well. We believe that the positive effect of the improved lemmatization being somewhat small can be at least partially attributed to these issues. Accordingly, some future work could be dedicated to improving the translated dataset, e.g., manually correcting the

query translations, performing translation quality estimation to redo or filter out low quality document translations, and filtering out entries with no useful content. Consequently, we would expect a larger positive effect of improved lemmatization on a corrected dataset. However, we believe that the noisiness of the dataset affects each approach similarly, and thus the relative ranking between the preprocessing methods remains stable.

We also note that both disambiguation approaches are somewhat computationally intensive. In the current implementation of GliNER, the batch processing does not allow different sets of labels for each example, and thus each example must be processed separately, which makes it difficult to make use of GPU acceleration during inference. The Vabamorf disambiguator, on the other hand, cannot be accelerated at all. Applying both disambiguation approaches to the large corpus of 4.5M documents took over 50 hours for each, using parallelization with approximately 30 concurrent processes on CPU hardware.

6 Conclusion

This study demonstrates that integrating an external disambiguation model like GliLem with a rule-based morphological analyzer can substantially improve the accuracy of lemmatization in Estonian. The enhanced lemmatization bridges the accuracy gap caused by the limitations of Vabamorf’s built-in disambiguator. This proves our initial hypothesis that the main weakness of Vabamorf is in fact its inability to correctly select the lemma candidate in context.

Additionally, we estimated the effect of improved lemmatization accuracy on an information retrieval downstream task. Although the precise effect is difficult to estimate due to the noisiness of the original data and additional noise introduced by imperfect machine translation, we observed small consistent improvements in Recall, and especially in the setting with a high k , suggesting that improved lemmatization might translate into actual improvements in a hybrid information retrieval setting.

Acknowledgments

This research was supported by the Estonian Research Council Grant PSG721.

References

- Stefan Butcher, Charles LA Clarke, and Gordon V Cormack. 2016. *Information Retrieval: Implementing and Evaluating Search Engines*. MIT Press.
- Aleksei Dorkin and Kairit Sirts. 2023. Comparison of Current Approaches to Lemmatization: A Case Study in Estonian. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 280–285, Tórshavn, Faroe Islands. University of Tartu Library.
- Aleksei Dorkin and Kairit Sirts. 2024. Sõnajaht: Definition Embeddings and Semantic Search for Reverse Dictionary Creation. In *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (*SEM 2024)*, pages 410–420, Mexico City, Mexico. Association for Computational Linguistics.
- Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2020. The Pile: An 800GB Dataset of Diverse Text for Language Modeling. *ArXiv*, abs/2101.00027.
- Luyu Gao, Zhuyun Dai, Tongfei Chen, Zhen Fan, Benjamin Van Durme, and Jamie Callan. 2021. Complementing Lexical Retrieval with Semantic Residual Embedding. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28–April 1, 2021, Proceedings, Part I 43*, pages 146–160. Springer.
- Faegheh Hasibi, Fedor Nikolaev, Chenyan Xiong, Krisztian Balog, Svein Erik Bratsberg, Alexander Kotov, and Jamie Callan. 2017. DBpedia-Entity V2: A Test Collection for Entity Search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR ’17*, pages 1265–1268. ACM.
- Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-Efficient Transfer Learning for NLP. In *International conference on machine learning*, pages 2790–2799. PMLR.
- Heiki-Jaan Kaalep and Tarmo Vaino. 2001. Complete Morphological Analysis in the Linguist’s Toolbox. *Congressus Nonus Internationalis Fenno-Ugristarum Pars V*, pages 9–16.
- Jenna Kanerva, Filip Ginter, Niko Miekka, Akseli Leino, and Tapio Salakoski. 2018. Turku Neural Parser Pipeline: An End-to-End System for the CoNLL 2018 Shared Task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual parsing from raw text to universal dependencies*, pages 133–142.
- Jenna Kanerva, Filip Ginter, and Tapio Salakoski. 2021. Universal Lemmatizer: A sequence-to-sequence model for lemmatizing Universal Depen-

- dencies treebanks. *Natural Language Engineering*, 27(5):545–574.
- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781.
- Omar Khattab, Christopher Potts, and Matei Zaharia. 2021. Relevance-guided Supervision for OpenQA with ColBERT. *Transactions of the Association for Computational Linguistics*, 9.
- Dohyeon Lee, Seung-won Hwang, Kyungjae Lee, Seungtaek Choi, and Sunghyun Park. 2023. On Complementarity Objectives for Hybrid Retrieval. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13357–13368.
- Xing Han Lù. 2024. BM25S: Orders of Magnitude Faster Lexical Search via Eager Sparse Scoring. *arXiv preprint arXiv:2407.03618*.
- Siim Orasmaa, Timo Petmanson, Alexander Tkachenko, Sven Laur, and Heiki-Jaan Kaalep. 2016. EstNLTk - NLP Toolkit for Estonian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).
- Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at TREC-3. *Nist Special Publication Sp*, 109:109.
- Milan Straka. 2018. UDPipe 2.0 Prototype at CoNLL 2018 UD Shared Task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 197–207, Brussels, Belgium. Association for Computational Linguistics.
- Milan Straka, Jana Straková, and Jan Hajic. 2019. UDPipe at SIGMORPHON 2019: Contextualized Embeddings, Regularization with Morphological Categories, Corpora Merging. *SIGMORPHON 2019*, page 95.
- NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No Language Left Behind: Scaling Human-Centered Machine Translation. *arXiv preprint arXiv:2207.04672*.
- Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*.
- Olia Toporkov and Rodrigo Agerri. 2024a. Evaluating Shortest Edit Script Methods for Contextual Lemmatization. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 6451–6463.
- Olia Toporkov and Rodrigo Agerri. 2024b. On the Role of Morphological Information for Contextual Lemmatization. *Computational Linguistics*, 50(1).
- University of Tartu. 2018. UT Rocket.
- Shijie Wu, Ryan Cotterell, and Mans Hulden. 2021. Applying the Transformer to Character-level Transduction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1901–1907, Online. Association for Computational Linguistics.
- Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. 2024. GLiNER: Generalist Model for Named Entity Recognition using Bidirectional Transformer. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Mexico City, Mexico. Association for Computational Linguistics.
- Daniel Zeman, Joakim Nivre, Mitchell Abrams, Elia Ackermann, Noëmi Aepli, Hamid Aghaei, Željko Agić, Amir Ahmadi, Lars Ahrenberg, Chika Kennedy Ajede, Salih Furkan Akkurt, Gabrielè Aleksandravičiūtė, Ika Alfina, Avner Algom, Khalid Alnajjar, Chiara Alzetta, Erik Andersen, Lene Antonsen, Tatsuya Aoyama, Katya Aplonova, Angelina Aquino, Carolina Aragon, Glyd Aranes, Maria Jesus Aranzabe, Bilge Nas Arıcan, Hórunn Arnardóttir, Gashaw Arutie, Jessica Naraiswari Arwidarasti, Masayuki Asahara, Katla Ásgeirsdóttir, Deniz Baran Aslan, Cengiz Asmazoğlu, Luma Ateyah, Furkan Atmaca, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, Mariana Avelãs, Elena Badmaeva, Keerthana Balasubramani, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, Verginica Barbu Mititelu, Starkaður Barkarson, Rodolfo Basile, Victoria Basmov, Colin Batchelor, John Bauer, Seyyit Talha Bedir, Shabnam Behzad, Kepa Bengoetxea, İbrahim Benli, Yifat Ben Moshe, Gözde Berk, Riyaz Ahmad Bhat, Erica Biagetti, Eckhard Bick, Agnė Bielinskienė,

Kristín Bjarnadóttir, Rogier Blokland, Victoria Bobicev, Loïc Boizou, Emanuel Borges Völker, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Adriane Boyd, Anouck Braggaar, António Branco, Kristina Brokaitė, Aljoscha Burchardt, Marisa Campos, Marie Candito, Bernard Caron, Gauthier Caron, Catarina Carvalheiro, Rita Carvalho, Lauren Cassidy, Maria Clara Castro, Sérgio Castro, Tatiana Cavalcanti, Gülşen Cebiroğlu Eryiğit, Flavio Massimiliano Cecchini, Giuseppe G. A. Celano, Slavomír Čéplö, Neslihan Cesur, Savas Cetin, Özlem Çetinoğlu, Fabricio Chalub, Liyanage Chamila, Shweta Chauhan, Ethan Chi, Taishi Chika, Yongseok Cho, Jinho Choi, Jayeol Chun, Juyeon Chung, Alessandra T. Cignarella, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Daniela Corbetta, Francisco Costa, Marine Courtin, Mihaela Cristescu, Ingerid Løyning Dale, Philemon Daniel, Elizabeth Davidson, Leonel Figueiredo de Alencar, Mathieu Dehouck, Martina de Laurentiis, Marie-Catherine de Marneffe, Valeria de Paiva, Mehmet Oguz Derin, Elvis de Souza, Arantza Diaz de Ilarraza, Carly Dickerson, Arawinda Dinakaramani, Elisa Di Nuovo, Bamba Dione, Peter Dirix, Kaja Dobrovoljc, Adrian Doyle, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Christian Ebert, Hanne Eckhoff, Masaki Eguchi, Sandra Eiche, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Olga Erina, Tomaž Erjavec, Farah Es-saidi, Aline Etienne, Wograinne Evelyn, Sidney Facundes, Richárd Farkas, Federica Favero, Jannatul Ferdaousi, Marília Fernanda, Hector Fernandez Alcalde, Amal Fethi, Jennifer Foster, Cláudia Freitas, Kazunori Fujita, Katarína Gajdošová, Daniel Galbraith, Federica Gamba, Marcos Garcia, Moa Gärdenfors, Fabrício Ferraz Gerardi, Kim Gerdes, Luke Gessler, Filip Ginter, Gustavo Godoy, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta González Saavedra, Bernadeta Griciūtė, Matias Grioni, Loïc Grobol, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Tunga Güngör, Nizar Habash, Hinrik Hafsteinsson, Jan Hajič, Jan Hajič jr., Mika Hämäläinen, Linh Hà Mỹ, Na-Rae Han, Muhammad Yudistira Hanifmuti, Takahiro Harada, Sam Hardwick, Kim Harris, Dag Haug, Johannes Heinecke, Oliver Hellwig, Felix Hennig, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Marivel Huerta Mendez, Jena Hwang, Takumi Ikeda, Anton Karl Ingason, Radu Ion, Elena Irimia, Olájidé Ishola, Artan Islamaj, Kaoru Ito, Siratun Jannat, Tomáš Jelínek, Apoorva Jha, Katharine Jiang, Anders Johannsen, Hildur Jónsdóttir, Fredrik Jørgensen, Markus Juutinen, Hüner Kaşıkara, Nadezhda Kabaeva, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Neslihan Kara, Ritván Karahóga, Andre Kåsen, Tolga Kayadelen, Sarveswaran Kengatharaiyer, Václava Kettnerová, Jesse Kirchner, Elena Klementieva, Elena Klyachko, Arne Köhn, Abdullatif Köksal, Kamil Kopacewicz, Timo Korkiakangas, Mehmet Köse, Alexey Koshevoy, Natalia Kotsyba, Jolanta Kovalevskaitė, Simon Krek, Parameswari Krish-

namurthy, Sandra Kübler, Adrian Kuqi, Oğuzhan Kuyrukçu, Aslı Kuzgun, Sookyoung Kwak, Kris Kyle, Veronika Laippala, Lorenzo Lambertino, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Phuong Lê Hồng, Alessandro Lenci, Saran Lertpradit, Herman Leung, Maria Levina, Lauren Levine, Cheuk Ying Li, Josie Li, Keying Li, Yixuan Li, Yuan Li, KyungTae Lim, Bruna Lima Padovani, Yi-Ju Jessica Lin, Krister Lindén, Yang Janet Liu, Nikola Ljubešić, Olga Loginova, Stefano Lusito, Andry Luthfi, Mikko Luukko, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Menel Mahamdi, Jean Maillard, Ilya Makarchuk, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Büşra Marşan, Cătălina Măranduc, David Mareček, Katrin Marheinecke, Stella Markantonatou, Héctor Martínez Alonso, Lorena Martín Rodríguez, André Martins, Cláudia Martins, Jan Mašek, Hiroshi Matsuda, Yuji Matsumoto, Alessandro Mazzei, Ryan McDonald, Sarah McGuinness, Gustavo Mendonça, Tatiana Merzhevich, Niko Miekka, Aaron Miller, Karina Mischenkova, Anna Missilä, Cătălin Mititelu, Maria Mitrofan, Yusuke Miyao, AmirHosseini Mojiri Foroushani, Judit Molnár, AmirSaeid Moloodi, Simonetta Montemagni, Amir More, Laura Moreno Romero, Giovanni Moretti, Shinsuke Mori, Tomohiko Morioka, Shigeki Moro, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Robert Munro, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Mariam Nakhle, Juan Ignacio Navarro Horriacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, Manuela Nevaci, Luong Nguyễn Thị, Huyền Nguyễn Thị Minh, Yoshihiro Nikaido, Vitaly Nikolaev, Rattima Nitisaroj, Alireza Nourian, Hanna Nurmi, Stina Ojala, Atul Kr. Ojha, Hulda Óladóttir, Adedayo Olúokun, Mai Omura, Emeka Onwuegbuzia, Noam Ordan, Petya Osenova, Robert Östling, Lilja Øvrelid, Şaziye Betül Özateş, Merve Özçelik, Arzuçan Özgür, Balkız Öztürk Başaran, Teresa Pacosi, Alessio Palmero Aprosio, Anastasia Panova, Hyunji Hayley Park, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Guilherme Paulino-Passos, Giulia Pedonese, Angelika Peljak-Lapińska, Siyao Peng, Siyao Logan Peng, Rita Pereira, Sílvia Pereira, Cenel-Augusto Perez, Natalia Perkova, Guy Perrier, Slav Petrov, Daria Petrova, Andrea Peverelli, Jason Phelan, Jussi Piitulainen, Yuval Pinter, Clara Pinto, Tommi A Pirinen, Emily Pitler, Magdalena Plamada, Barbara Plank, Thierry Poibeau, Larisa Ponomareva, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Robert Pugh, Tina Puolakainen, Sampo Pyysalo, Peng Qi, Andreia Querido, Andriela Răăbis, Alexandre Rademaker, Mizanur Rahoman, Taraka Rama, Loganathan Ramasamy, Joana Ramos, Fam Rashel, Mohammad Sadegh Rasooli, Vinit Ravishankar, Livy Real, Petru Rebeja, Siva Reddy, Mathilde Regnault, Georg Rehm, Arij Riabi, Ivan Riabov, Michael Rießler, Erika Rimkutė, Larissa Rinaldi, Laura Rituma, Putri Rizqiyah, Luisa Rocha, Eiríkur Rögnvalds-

son, Ivan Roksandic, Mykhailo Romanenko, Rudolf Rosa, Valentin Roşca, Davide Rovati, Ben Ronzoner, Olga Rudina, Jack Rueter, Kristján Rúnarsson, Shoval Sadde, Pegah Safari, Aleksi Sahala, Shadi Saleh, Alessio Salomoni, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Ezgi Sanıyar, Dage Särg, Marta Sartor, Mitsuya Sasaki, Baiba Saulīte, Yanin Sawanakunanon, Shefali Saxena, Kevin Scannell, Salvatore Scarlata, Nathan Schneider, Sebastian Schuster, Lane Schwartz, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Syeda Shahzadi, Mo Shen, Atsuko Shimada, Hiroyuki Shirasu, Yana Shishkina, Muh Shohibussirri, Maria Shvedova, Janine Siewert, Einar Freyr Sigurðsson, João Silva, Aline Silveira, Natalia Silveira, Sara Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Haukur Barri Símonarson, Kiril Simov, Dmitri Sitchinava, Ted Sither, Maria Skachedubova, Aaron Smith, Isabela Soares-Bastos, Per Erik Solberg, Barbara Sonnenhauser, Shafi Sourov, Rachele Sprugnoli, Vivian Stamou, Steinhór Steingrímsson, Antonio Stella, Abishek Stephen, Milan Straka, Emmett Strickland, Jana Strnadová, Alane Suhr, Yogi Lesmana Sulestio, Umut Sulubacak, Shingo Suzuki, Daniel Swanson, Zsolt Szántó, Chihiro Taguchi, Dima Taji, Fabio Tamburini, Mary Ann C. Tan, Takaaki Tanaka, Dipta Tanaya, Mirko Tavoni, Samson Tella, Isabelle Tellier, Marinella Testori, Guillaume Thomas, Sara Tonelli, Liisi Torga, Marsida Toska, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Utku Türk, Francis Tyers, Sveinbjörn Hórdarson, Vilhjálmur Hörsteinsson, Sumire Uematsu, Roman Untilov, Zdeňka Uřešová, Larraitz Uriá, Hans Uszkoreit, Andrius Utkas, Elena Vagnoni, Sowmya Vajjala, Socrates Vak, Rob van der Goot, Martine Vanhove, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Uliana Vedenina, Giulia Venturi, Veronika Vincze, Natalia Vlasova, Aya Wakasa, Joel C. Wallenberg, Lars Wallin, Abigail Walsh, Jonathan North Washington, Maximilian Wendt, Paul Widmer, Shira Wigderson, Sri Hartati Wijono, Seyi Williams, Mats Wirén, Christian Wittern, Tsegay Woldemariam, Tak-sum Wong, Alina Wróblewska, Mary Yako, Kayo Yamashita, Naoki Yamazaki, Chunxiao Yan, Koichi Yasuoka, Marat M. Yavrumyan, Arife Betül Yenice, Olcay Taner Yıldız, Zhuoran Yu, Arlisa Yuliawati, Zdeněk Žabokrtský, Shorouq Zahra, Amir Zeldes, He Zhou, Hanzhi Zhu, Yilun Zhu, Anna Zhuravleva, and Rayan Ziane. 2023. Universal dependencies 2.12. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Wenxuan Zhou, Sheng Zhang, Yu Gu, Muhao Chen, and Hoifung Poon. 2023. UniversalNER: Targeted Distillation from Large Language Models for Open Named Entity Recognition. *arXiv preprint arXiv:2308.03279*.

Comparative analysis of optical character recognition methods for Sámi texts from the National Library of Norway

Tita Enstad¹, Trond Trosterud², Marie Iversdatter Røsok¹, Yngvil Beyer¹, Marie Roald¹

¹National Library of Norway

²The Arctic University of Norway

tita.enstad@nb.no

Abstract

Optical Character Recognition (OCR) is crucial to the National Library of Norway's (NLN) digitisation process as it converts scanned documents into machine-readable text. However, for the Sámi documents in NLN's collection, the OCR accuracy is insufficient. Given that OCR quality affects downstream processes, evaluating and improving OCR for text written in Sámi languages is necessary to make these resources accessible. To address this need, this work fine-tunes and evaluates three established OCR approaches, Transkribus, Tesseract and TrOCR, for transcribing Sámi texts from NLN's collection. Our results show that Transkribus and TrOCR outperform Tesseract on this task, while Tesseract achieves superior performance on an out-of-domain dataset. Furthermore, we show that fine-tuning pre-trained models and supplementing manual annotations with machine annotations and synthetic text images can yield accurate OCR for Sámi languages, even with a moderate amount of manually annotated data.

1 Introduction

Optical Character Recognition (OCR) converts scanned documents into machine-readable text, which is crucial for making digitised materials available for search and analysis. For the National Library of Norway (NLN), the OCR output, among others, facilitates search for the online library (*Nettbiblioteket*¹) and underpins analysis tools like the DH-Lab toolbox (Birkenes et al., 2023). However, while OCR quality is high for most Norwegian documents, it falls short for Sámi

documents. The resulting text is insufficient for both search and for use in research or as a basis for language technology.

NLN has material in five Sámi languages: North Sámi, South Sámi, Lule Sámi, Inari Sámi and Skolt Sámi. Thus, developing an accurate OCR model for Sámi texts is important for NLN's mission to store and disseminate the materials in the library collection. Furthermore, for languages with limited resources, like Sámi languages, it is vital that the available resources are accessible to be searched and used for research. This paper describes a twofold contribution towards this goal:

1. Developing an OCR model for Sámi languages that improves the transcription accuracy of Sámi text in NLN's collection.
2. Comparing different OCR approaches in terms of transcribing smaller languages such as languages in the Sámi family.

2 Background

2.1 Sámi languages in the National Library of Norway's collection

Of the around 650 000 books and 4.6 million newspaper issues in NLN's digitised collection, about 3000 and 4500 are classified as Sámi, respectively. The classification generally means that the texts are written in Sámi, though some may just address Sámi-related topics.

With more than 20 000 speakers North Sámi is the most widely spoken Sámi language in Norway, Sweden and Finland, and it makes up the largest part of the Sámi collection at NLN. The other Sámi languages in NLN's collection all have less than 500 speakers. South and Lule Sámi are spoken in Norway and Sweden, and the collection contains a good amount of South and Lule Sámi books. Skolt Sámi, previously spoken in Norway and Russia, is now mainly spoken in Finland,

¹<https://www.nb.no/search>

along with Inari Sámi, which has only ever been spoken in Finland. There is much less material in these languages in the collection (< 20 books in total).

All five languages have standardised orthographies that were made or revised in the 1970s, 80s or 90s (Laakso and Skribnik, 2022; Olthuis et al., 2013; Magga, 1994), but the collection also includes earlier works that predate the standardised norms. To some extent these books contain non-standard letters or glyph-shapes and most words are spelled in ways differing from contemporary orthographies.

The Sámi written languages have letters not found in the Norwegian alphabet, but it varies from language to language which letters and how many. The alphabets have some letters in common, but none are identical. See Table 1 for an overview of these characters.

North	South	Lule	Inari	Skolt
Áá		Áá	Áá	Áá
			Ââ	Ââ
			Ää	Ää
	Īī			Õõ
	Öö			
Čč			Čč	Čč
Đđ			Đđ	Đđ
Ŋŋ		Ŋŋ	Ŋŋ	Ŋŋ
Šš			Šš	Šš
Țț				
Žž			Žž	Žž
				ƷƷ
				Gg
				Ǧǧ
				Ǩǫ
				ǰǱ
				’
				;
				,

Table 1: Overview of non-Norwegian characters used in the contemporary orthographies of the Sámi languages in the collection

2.2 Related work

While early OCR approaches often relied on hand-crafted image features combined with shape- and text-analysis (Smith, 2007), modern solutions use deep learning based models to learn informative features from the data itself. In par-

ticular, developments like convolutional neural networks (CNNs), bidirectional long-short-term-memory (LSTMs) (Hochreiter and Schmidhuber, 1997) and the Connectionist Temporal Classification (CTC) loss (Graves et al., 2006) has yielded state-of-the-art results (Shi et al., 2016; Puigcerver, 2017; van Koert et al., 2024; Tarride et al., 2024). Recently, transformer-based machine learning advancements have led to transformer-based OCR models such as TrOCR (Li et al., 2023).

OCR pipelines have also been developed for collections of digitised documents: Tesseract (Smith, 2007) is an open-source OCR framework for line segmentation and text recognition which includes pre-trained OCR models for several languages² and training scripts for training and fine-tuning on custom data. Since 2018, Tesseract has also supported LSTMs.

Another example is Transkribus, a proprietary platform for the recognition of printed and handwritten documents with a built-in interface for (semi-)manual transcription. The platform supports layout analysis and text recognition, using pre-existing or custom-trained models. The text recognition models are based on PyLaia (Puigcerver, 2017; Tarride et al., 2024), which uses a combination of CNNs and bidirectional LSTMs. Transcriptions can be exported, though models are restricted to use within the platform.

A recent advancement is transformers-based OCR. TrOCR is a state-of-the-art text recognition model that combines powerful transformer models for vision and language (Li et al., 2023). Specifically, TrOCR combines the “encoder” of a vision transformer (ViT) (Dosovitskiy et al., 2021), with the language generating “decoder” of a robustly optimised Bidirectional encoder representations from transformers approach (RoBERTa) model (Liu et al., 2020). TrOCR is specialised for text recognition, and will not perform ancillary tasks, like layout analysis. Moreover, while TrOCR is shown capable of outperforming Transkribus and Tesseract (Ströbel et al., 2023; Li et al., 2023), it is still a relatively recent algorithm, and there is still a need to assess its accuracy for low-resource languages.

OCR quality greatly impacts downstream processes (Lopresti, 2008; Järvelin et al., 2016; Ersner and Fitch, 2014). Consequently, parts of

²but none for the Sámi languages

a digitised collection with challenges like unusual fonts, bad scan quality or text in a low-resource language, will be less accessible. Several works have, thus, focused on improving OCR quality for texts with such challenges by e.g. using an ensemble of image preprocessing transforms (Koistinen et al., 2017), comparing various OCR- or hand-written text recognition (HTR)-models for smaller languages (Maarand et al., 2022; Memon et al., 2020; Tafti et al., 2016; Koistinen et al., 2017; He-liński et al., 2012) or post-correcting outputs (Poncelas et al., 2020; Duong et al., 2021).

OCR for low-resource languages is particularly challenging. Not only is there much less labelled data for training, but this problem is exacerbated further by potential changes in orthographies. Rijhwani et al. (2023) showed that including OCR in a semi-automatic annotation suite can aid annotation – even for a low-resource language such as Kwak’wala, where automatic annotation is difficult. Similarly, Yaseen and Hassani (2024) trained a Tesseract-based OCR system for Kurdish, another low-resource language. Agarwal and Anastopoulos (2024) presented a concise survey of OCR for low-resource languages with a focus on Indigenous Languages of the Americas. Finally, Partanen and Rießler (2019) presented an OCR model for the Unified Northern Alphabet, used in the Soviet Union between 1931 and 1937 for Northern Minority languages (which includes Kildin Sámi).

3 Methods

3.1 Data

The main source for the data used in this work is NLN’s digitised collection. Our goal was to create an OCR model for all languages in the collection, rather than one for each language, as this would allow for the most efficient integration into NLN’s digitisation pipeline. However, we realised early that including Skolt Sámi would be difficult because of the three apostrophe characters that indicate pronunciation. This makes transcription difficult without a certain level of language proficiency. Thus, we proceeded with North, South, Lule and Inari Sámi.

In addition to data from NLN, we also used text-data from the GiellaLT corpora³ as basis for synthetic text images and data from the Divvun &

³<https://giellalt.github.io/>

		South	North	Lule	Inari
Docs	GT	5	3	2	3
	Pred	265	1810	235	0
	Val	2	8	2	3
	Test	4	7	4	5
Lines	GT	208	5572	81	280
	Pred	7082	70413	6781	0
	Synth	76971	76949	76970	76497
	Val	53	1837	36	109
	Test	195	353	137	163
	OOD	0	122	0	0

Table 2: Distribution of documents and lines in each of the Sámi languages in the different datasets. GT, Val and Test refer to the data splits of the manually annotated data. Pred is the automatically annotated dataset, Synth is the synthetic dataset (natural language text but generated images) and OOD is the OOD Giellatekno test set.

Giellatekno fork of tesstrain⁴ as basis for an out-of-domain (OOD) test set.

Training data

We trained OCR models using manually transcribed data, machine transcribed data, and synthetic data⁵. See Table 2 for an overview.

Manually transcribed data We used Transkribus⁶ (Kahle et al., 2017) to create the training data from the images of scanned pages. We used the platform’s layout analysis, manually adjusting the results where necessary, then applied text recognition to the documents. Initially, we used a standard model provided by Transkribus. As we progressively corrected the recognised text, we trained new models, which were applied to recognise text in new documents, which we manually corrected to create the manually transcribed data.

Following this procedure, we transcribed 58 Sámi book and newspaper pages to create a manually transcribed training set, henceforth referred to as *Ground Truth Sámi* (GT-Sámi).

⁴https://github.com/divvungiellatekno/tesstrain/tree/main/training-data/nor_sme-ground-truth

⁵As these texts contain copyrighted materials, the transcribed data sets can not be shared openly.

⁶We used the Transkribus Expert Client v1.28.0 and <https://app.transkribus.org> v4.0.0.150

Additionally, we already had 82 pages with 2998 manually transcribed Norwegian text lines (produced similarly as for GT-Sámi) that we included as training data. We refer to this data as *Ground Truth Norwegian* (GT-Nor).

Synthetic data To add more annotated Sámi text, we created synthetic data, which we refer to as the *Synthetic Sámi* dataset (Synth-Sámi). We used the SIKOR Sámi text corpus (SIKOR, 2021) as a basis of well-formed Sámi text, and generated images for the text lines (adding an uppercase version for $\simeq 10\%$ of the lines), using `CorpusTools`⁷ to parse the XML files in the converted-directory of the `corpus-sma`⁸, `corpus-sme`⁹, `corpus-smj`¹⁰ and `corpus-smn`¹¹ repositories. The images were created with `Pillow`¹² and `Augraphy` (Groleau et al., 2023), with variation in fonts and colours, and a varying degree of imperfections and noise added, resulting in 307 387 lines¹³.

Automatically transcribed data As mentioned earlier, we trained Transkribus models incrementally while annotating data. Eventually, our Transkribus model¹⁴ performed well on North, South and Lule Sámi, and we decided to automatically transcribe a larger amount of Sámi text with this model. We extracted page 30 from North, South and Lule Sámi books in NLN’s collection and transcribed them automatically, which resulted in 2380 pages forming the *Predicted Sámi* (Pred-Sámi) dataset. This boosted the amount of data, but naturally, the transcriptions may not be correct.

Validation data

To evaluate during training and to select the best performing models for each architecture, we created a validation dataset. This dataset consists of 25 pages manually transcribed following the procedure described for GT-Sámi. Lines were

⁷<https://github.com/divvun/CorpusTools>

⁸<https://github.com/giellalt/corpus-sma/>

⁹<https://github.com/giellalt/corpus-sme/>

¹⁰<https://github.com/giellalt/corpus-smj/>

¹¹<https://github.com/giellalt/corpus-smn/>

¹²<https://python-pillow.org/> (Version 10.4.0)

¹³Code to generate synthetic data is on GitHub: https://github.com/Sprakbanken/synthetic_text_images

¹⁴Transkribus modelID 115833, publicly available in Transkribus

selected from different books than the GT-Sámi training data while keeping a similar language distribution.

Test data

To compare the OCR approaches we used two test sets: one from NLN’s collection and one from Divvun & Giellatekno’s tesstrain data.

NLN test data As a goal of this work was to improve the transcriptions of Sámi documents in NLN’s collection, we created a test set based on current transcriptions (baseline) of 21 pages from 18 books and 2 newspapers provided by NLN¹⁵. NLN stores these transcriptions as Analyzed Layout and Text Object-Extensible Markup Language (ALTO-XML) files with line segmentations and transcriptions. By matching the ALTO-XML transcriptions with manually annotated data, we created a test-set containing 848 text-lines.

Giellatekno test data The Giellatekno test data *nor-sme* was made for evaluating OCR reading of dictionaries. It consists of 122 lines of dictionary data, thus text both in Norwegian and (contemporary) North Sámi. The dataset is available on Giellatekno’s GitHub¹⁶. We refer to this dataset as the OOD Giellatekno test set.

3.2 Evaluation metrics

Following previous work (Neudecker et al., 2021; Agarwal and Anastasopoulos, 2024), we used the character error rate (CER) and word error rate (WER) evaluation metrics. Specifically, we calculated collection level CER and WER (concatenating lines, with a space to separate them for WER) with Jiwer¹⁷.

We also calculated an F_1 score for characters specific to the different Sámi languages, and an overall F_1 score for all non-Norwegian Sámi characters. The F_1 score is given by $F_1 = 2TP/(2TP + FN + FP)$, where TP, FP and FN is the number

¹⁵We chose distinct books for the train, validation and test sets. However, due to few Inari Sámi books, 1 book is in both the train and test sets and 2 are in both the validation and test sets, but there is no page-overlap.

¹⁶https://github.com/divvungiellatekno/tesstrain/tree/main/training-data/nor_sme-ground-truth. We have corrected four transcriptions and used our corrected version of the test set which can be found on https://github.com/MarieRoald/tesstrain/tree/fix-transcriptions/training-data/nor_sme-ground-truth

¹⁷<https://github.com/jitsi/jiwer> (Version 3.0.4)

of true positives, false positives and false negatives, respectively. To measure TP, FP and FN in an OCR-setting, we only considered character counts, not location. Thus, for a given character, c , we set $TP_c = \min(n_c^{(\text{true})}, n_c^{(\text{pred})})$, $FN = \max(n_c^{(\text{true})} - n_c^{(\text{pred})}, 0)$ and $FP = \max(n_c^{(\text{pred})} - n_c^{(\text{true})}, 0)$, where $n_c^{(\text{true})}$ and $n_c^{(\text{pred})}$ are the number of c characters in the ground truth and predicted transcriptions, respectively. To compute an overall F_1 , we combined the TP, FN, and FP across all lines and characters-of-interest.

To examine the types of errors our models made, we calculated the most common errors. Specifically, we used Stringalign (Moe and Roald, 2024), which implements optimal string alignment. Note that, in theory, multiple alignments can exist (e.g. if two letters are swapped), in which case Stringalign picks one.

3.3 Models and training

A goal of this work was evaluating different state-of-the-art OCR frameworks for Sámi text recognition. Specifically, we compared Transkribus, Tesseract and TrOCR. For each approach, we trained on several dataset combinations and chose the model based on mean(CER, WER) on the validation data for test-set evaluation.

Transkribus

We used Transkribus Expert for training Transkribus models¹⁸. We used standard parameters, but opted “Using existing line polygons for training”, and changed the batch size from 24 to 12¹⁹. We set 100 as maximum numbers of epochs, and 20 as early stopping. We used Transkribus print M1²⁰ as base model for 4 of the 5 models. All Transkribus models were run with the setting “Use language model”²¹.

Tesseract

We used the official tesstrain repository²² and Tesseract 5.4.1 for training. We experimented with both training models from scratch and fine-tuning

existing models. During early experiments, we tried fine-tuning Norwegian, Finnish, and Estonian models using our Sámi dataset, and observed that the model with the Norwegian base adapted faster and performed better on our validation set. Thus, we continued training with the Norwegian base²³.

As tesstrain does not support dynamic learning rate and only exposes a few training hyperparameters to the user, we trained our models in 1-20 epoch increments, updating the learning rate until the model checkpoints no longer showed improvements on the validation set.

TrOCR

We used Huggingface Transformers (Wolf et al., 2020) to fit the TrOCR models, initialising with the parameters from the microsoft/trocr-base-printed repository. This model is pre-trained on both synthetic and printed text (Li et al., 2023). For fine-tuning, we had an initial learning rate of 10^{-6} , decreasing it by a constant amount for each iteration until it reached 10^{-7} at the final iteration. For models fine-tuned without Pred-Sámi, we trained for 200 epochs, evaluating and storing model parameters every fifth epoch. However, due to the data size and hardware limitations, models that included Pred-Sámi were only fine-tuned for 100 epochs, evaluating and storing model parameters every second epoch and selecting the checkpoint with the lowest validation CER.

Pre-training with synthetic data

We trained additional TrOCR and Tesseract models using synthetic data to assess the effect of adding such data²⁴. After training all models without synthetic data, we retrained with the smallest amount of hand-annotated data (GT-Sámi) and best performing data combination, this time initialising with a model pre-trained on Synth-Sámi.

In particular, due to time and hardware limitations, we trained models on synthetic data in two stages inspired by the two-stage procedure in e.g (Li et al., 2023). For the first stage, we trained for five epochs on Synth-Sámi. For the second stage, we initialised with the best checkpoint from the

¹⁸<https://help.transkribus.org/model-setup-and-training>

¹⁹We changed this parameter after advice from the Transkribus team due to problems with the training stopping with `exitCode = 1`

²⁰Transkribus ModelID 39995

²¹Which uses PyLaia’s n-gram model functionality to inform character predictions (Tarride et al., 2024).

²²<https://github.com/tesseract-ocr/tesstrain> (Version 1.0.0, commit hash 45cacc5)

²³https://github.com/tesseract-ocr/tesdata_best/blob/main/nor.traineddata

²⁴We did not train Transkribus models with synthetic data as it does not support an easy way to train based on line images and because of its page-based pricing model.

w/o base	GT-Sámi	GT-Nor	Pred-Sámi	Synth base	Transkribus			Tesseract			TrOCR		
					CER	WER	mean	CER	WER	mean	CER	WER	mean
✓	✓				1.59	5.67	3.63	5.53	24.70	15.11			
	✓				1.28	4.34	2.81	2.05	9.84	5.95	1.98	9.29	5.64
	✓	✓			1.31	4.35	2.83	2.37	11.39	6.88	1.95	8.88	5.42
	✓		✓		1.48	4.02	2.75	1.85	8.17	5.01	1.28	5.00	3.14
	✓	✓	✓		1.07	3.58	2.33	1.81	7.96	4.89	1.32	5.14	3.23
	✓			✓				1.78	8.78	5.28	1.15	5.04	3.09
	✓		✓	✓							1.08	4.29	2.69
	✓	✓	✓	✓				1.79	7.70	4.75			

Table 3: CER, WER, and mean(CER, WER) on the validation set. The checkmarks indicate whether models were trained from scratch (i.e. not fine-tuning an existing base model) (first column) and what datasets were part of the training data

first stage (lowest CER) and continued training on real data.

4 Results

Code for training Tesseract and TrOCR models, creating synthetic data and more detailed dataset information is available through the supplement on GitHub²⁵.

4.1 NLN validation data

Transkribus models

As shown in Table 3, CER and WER decreased when we used the Transkribus Print M1 as the base model in addition to GT-Sámi. Hence, we continued to use the base model in the subsequent training. Supplementing GT-Sámi with GT-Nor did not improve performance, while supplementing with Pred-Sámi increased CER but decreased WER. However, adding both GT-Nor and Pred-Sámi led to the best-performing model on the validation set.

Tesseract models

From Table 3, we see that the model trained on GT-Sámi with a Norwegian base model greatly outperformed the corresponding model without a base model. We therefore continued training all Tesseract models from the Norwegian base model. Adding GT-Nor to the training data worsened the validation performance. However, adding

Pred-Sámi to the training data improved validation performance, and adding both further improved the performance. Using Synth-Sámi also improved performance, and the model performed best in terms of mean(CER, WER) when all training datasets were used.

TrOCR models

For TrOCR, we observed that including GT-Nor in the training had a slight improvement when only training with GT-Sámi and no improvement when training with GT-Sámi and Pred-Sámi (see Table 3). Moreover, while including Pred-Sámi improved performance, pre-training with Synth-Sámi had a larger effect. The overall best-performing model was trained with both Synth-Sámi and Pred-Sámi in addition to GT-Sámi.

4.2 NLN test data

Table 4, shows that while Transkribus achieves a lower CER for most languages, it obtains a higher WER and a lower special character F_1 -score compared to TrOCR. Tesseract performed worst on this dataset. However, all models greatly improve compared to the baseline, with the CER and WER being reduced by factors between 3.8 and 5.6.

The special character F_1 -score in Table 4 shows that the baseline struggles with non-Norwegian Sámi characters. While the F_1 score does not take letter position into account, we also see the same pattern reflected in Table 5, which shows that seven of the ten most common mistakes for the baseline are replacing a non-Norwegian Sámi special character. In contrast, we see that our three

²⁵https://github.com/Sprakbanken/nodalida25_sami_ocr

		Transkribus	Tesseract	TrOCR	Baseline
CER ↓ [%]	Overall	0.61	0.89	0.74	3.38
	South	0.33	1.09	0.33	2.05
	North	0.53	0.73	1.20	3.99
	Lule	0.34	0.26	0.66	2.46
	Inari	1.22	1.43	0.43	4.36
WER ↓ [%]	Overall	3.19	4.65	2.96	18.71
	South	2.42	7.45	2.33	15.98
	North	1.66	2.90	3.41	20.08
	Lule	3.27	1.84	3.47	13.27
	Inari	6.18	7.13	2.40	22.62
Sámi letter F ₁ ↑ [%]	Overall	96.03	93.81	96.97	52.54
	South	90.24	83.02	93.92	24.52
	North	98.57	97.13	97.27	55.85
	Lule	97.91	97.88	97.06	51.75
	Inari	94.70	93.22	98.84	68.61

Table 4: CER, WER and Sámi letter F1 on NLN test data. The score for each language and overall score across languages are listed. Transkribus, Tesseract and TrOCR refer to the best performing model on the validation set for each model type. Baseline is the current OCR output in the online library. The downward arrows indicate that a low score is better, while the upward arrow indicates that a high score is better.

models make fewer mistakes, and their ten most common mistakes are less systematically replacing distinctive Sámi characters and include, e.g. insertions and deletions.

4.3 Giellatekno test data

In contrast to the NLN test data, the Tesseract model performed the best on the OOD test data from Giellatekno for all metrics (see Table 6). Transkribus was worst in terms of CER and WER, while TrOCR was worst in terms of the F₁ score.

In Table 7, we see the most common errors on the Giellatekno test set. The Transkribus model seems to have a tendency to add punctuation marks, and mistake ø for e. All models fail to transcribe ü (of which there are only two in the Giellatekno test set). This is not surprising, as the letter rarely appears in the training data²⁶.

5 Discussion and conclusions

From Tables 3 and 4, we observe a jump in performance for the test set compared to the validation set. This increase is expected, as the test set annotations are of higher quality (more accurate line segmentations).

²⁶The letter ü appears 59 times in Synth-Sámi, 9 times in Pred-Sámi and 5 times in GT-Nor.

We see that applying a two-stage training using synthetic data for the first stage always improved the results. As such, if manual annotations are limited, the addition of synthetic data is worth considering. Furthermore, while the Pred-Sámi improved performance, its effect was less than including synthetic data. It would, thus, be interesting to investigate if further training on Synth-Sámi could eliminate the effect of Pred-Sámi. Finally, we note that including GT-Nor had a minimal effect when combined with Pred-Sámi. This finding, combined with the effect of pre-trained base models, suggests that language-independent features are already learned by the base models and highlights the value of language-specific data for fine-tuning on low-resource languages.

Unfortunately, as this work focuses on low-resource languages, few digitised texts exist. There is, therefore, a slight overlap between the books (but not pages) in the test set and the validation and training sets for Inari Sámi which could bias our results for the Inari Sámi language. Still, Inari Sámi obtained the worst CER and WER for Transkribus and the worst CER and second worst WER for Tesseract. Despite low amount of Inari Sámi, we included it in our analysis as there is an overlap between this alphabet and the North

Transkribus				Tesseract				TrOCR				Baseline			
Error	n_e	n_m	n_c	Error	n_e	n_m	n_c	Error	n_e	n_m	n_c	Error	n_e	n_m	n_c
‘á’→‘ǎ’	16	35	287	‘ĩ’→‘i’	24	27	160	‘Á’→‘A’	9	11	28	‘ǎ’→‘ǎ’	313	418	1136
‘â’→‘a’	14	35	287	‘â’→‘á’	22	29	287	‘‘’→‘i’	7	–	–	‘ĩ’→‘i’	137	139	160
‘Á’→‘A’	9	10	28	‘ď’→‘d’	12	14	173	‘Š’→‘S’	6	6	6	‘ǎ’→‘ǎ’	103	180	287
‘/’→‘ ’	9	9	10	‘Á’→‘A’	10	11	28	‘‘’→‘i’	5	–	–	‘-’→‘-’	75	77	82
‘i’→‘ĩ’	7	13	3299	‘‘’→‘d’	8	–	–	‘‘’→‘ ’	4	–	–	‘š’→‘s’	72	95	215
‘ď’→‘d’	7	11	173	‘‘’→‘á’	7	–	–	‘i’→‘ĩ’	4	21	3299	‘ď’→‘d’	48	61	173
‘š’→‘ ’	6	6	215	‘‘’→‘i’	7	–	–	‘ǎ’→‘ǎ’	4	14	1136	‘ǎ’→‘a’	46	418	1136
‘ä’→‘á’	5	6	150	‘s’→‘S’	7	8	1509	‘Č’→‘C’	4	4	8	‘â’→‘á’	30	180	287
‘i’→‘i’	5	5	160	‘â’→‘ǎ’	6	29	287	‘á’→‘a’	3	14	1136	‘â’→‘ä’	26	180	287
‘ ’→‘-’	4	–	–	‘ ’→‘ ’	5	6	509	‘a’→‘u’	3	8	3247	‘č’→‘c’	26	62	163

‘a’→‘b’: model transcribed “a” as “b”
‘a’→‘ ’: model incorrectly deleted “a”
‘ ’→‘b’: model incorrectly inserted “b”

n_e : Error count
 n_m : Misses of the character left of →
 n_c : Occurrences of the character left of →

Table 5: Top ten most common errors on the NLN test data. Transkribus, Tesseract and TrOCR refers to the best performing model on the validation set for each model type. Baseline is the current OCR output in the online library.

	Transkribus	Tesseract	TrOCR
CER ↓ [%]	0.70	0.12	0.43
WER ↓ [%]	5.85	1.02	3.31
F1 ↑ [%]	100.00	100.00	98.33

Table 6: CER, WER and Sámi letter F_1 on the OOD Giellatekno test set. The downwards arrows indicate that a low score is better, while the upwards arrow indicates that a high score is better.

Sámi alphabet, and our OCR models could improve upon NLN’s transcription for Inari Sámi.

All models improved considerably compared to the baseline and are good candidates for a re-OCR process. If transcription accuracy is the main focus, then Transkribus appears to perform the best. However, while Tesseract achieved the worst performance for the NLN test set, it performed the best on the OOD Giellatekno test set. Tesseract also has other benefits: it is available as open-source software and requires less compute than a TrOCR model.

While language-specific annotations are valuable, they are demanding to create, particularly for low-resource languages without good base models for semi-automatic annotations. However, our results show that by fine-tuning pre-trained models and augmenting manually annotated data with machine-annotated data and synthetic text images,

we can achieve accurate OCR for Sámi languages, even with modest amounts of manual annotations.

6 Further work

As NLN’s collection includes works predating the standardised Sámi orthographies, a more accurate evaluation of the OCR could be gained by examining performance across different time periods. Moreover, training specialised models to transcribe non-standard letters or glyph-shapes could enable more detailed down-stream studies of changes in orthographies. Another gap is training OCR for other Sámi languages, such as Skolt Sámi.

Given that our results show that initialising on a dataset of synthetic text images was beneficial, it is worth exploring further. The models in this work are only trained on synthetic data for five epochs, indicating that potential improvements could be made by training on synthetic data for longer, i.e. until convergence. Moreover, creating a larger synthetic dataset with greater variation of text, fonts and augmentations (e.g. additional scanning augmentations or simulating non-standard orthographies), could improve the results further.

As this study focuses on the text recognition step of the OCR pipeline and compares three models, future research should explore additional OCR components and models. E.g. examining the ef-

Transkribus				Tesseract				TrOCR			
Error	n_e	n_m	n_c	Error	n_e	n_m	n_c	Error	n_e	n_m	n_c
‘’ → ‘.’	12	—	—	‘ü’ → ‘i’	1	2	2	‘ü’ → ‘i’	2	2	2
‘ø’ → ‘e’	4	5	13	‘ü’ → ‘u’	1	2	2	‘’ → ‘,’	1	—	—
‘’ → ‘,’	2	—	—	‘t’ → ‘f’	1	1	220	‘t’ → ‘l’	1	2	220
‘ü’ → ‘u’	2	2	2	‘n’ → ‘m’	1	1	164	‘te’ → ‘s’	1	2	28
‘’ → ‘k’	1	—	—					‘l’ → ‘’	1	1	169
‘ø’ → ‘o’	1	5	13					‘o’ → ‘n’	1	1	149
‘c’ → ‘’	1	1	23					‘m’ → ‘n’	1	1	69
								‘c’ → ‘e’	1	1	23
								‘-’ → ‘_’	1	1	18
								‘ŋ’ → ‘ž’	1	1	9
								‘=’ → ‘2’	1	1	4
								‘x’ → ‘s’	1	1	2

‘a’ → ‘b’: model transcribed “a” as “b”

n_e : Error count

‘a’ → ‘’: model incorrectly deleted “a”

n_m : Misses of the character left of →

‘’ → ‘b’: model incorrectly inserted “b”

n_c : Occurrences of the character left of →

Table 7: Top ten most common errors on the OOD Giellatekno test data. Transkribus, Tesseract and TrOCR refers to the best performing model on the validation set for each model type.

fect of different line segmentation models and assessing if performance can be improved by fine-tuning the line segmentation or using end-to-end models. Additionally, extending the range of models examined — to include tools such as PyLaia (Puigcerver, 2017; Tarride et al., 2024) (which is part of Transkribus’ pipeline), Loghi (van Koert et al., 2024), GOT-OCR (Wei et al., 2024) or larger TrOCR models — could yield improvements. Lastly, including post processing, e.g. with tools from GiellaLT (Pirinen et al., 2023), could improve OCR quality.

Acknowledgments

We would like to thank Arne Martinus Lindstad for his contributions to the annotation of the data and valuable feedback.

References

- Milind Agarwal and Antonios Anastasopoulos. 2024. A concise survey of OCR for low-resource languages. In *Proceedings of the 4th Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP)*, pages 88–102. Association for Computational Linguistics.
- Magnus Breder Birkenes, Lars Johnsen, and Andre Kåsen. 2023. NB DH-LAB: A corpus infrastructure for social sciences and humanities computing.

In *CLARIN Annual Conference Proceedings, 2023*, pages 30–34, Leuven, Belgium. CLARIN.

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. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Quan Duong, Mika Härmäläinen, and Simon Hengchen. 2021. An unsupervised method for OCR post-correction and spelling normalisation for Finnish. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 240–248, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.

John Evershed and Kent Fitch. 2014. Correcting noisy OCR: Context beats confusion. In *DATeCH ’14: Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage*, DATeCH ’14, pages 45–51, New York, NY, USA. Association for Computing Machinery.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning, ICML ’06*, page 369–376, New York, NY, USA. Association for Computing Machinery.

- Alexander Groleau, Kok Wei Chee, Stefan Larson, Samay Maini, and Jonathan Boorman. 2023. Augraphy: A data augmentation library for document images. In *Document Analysis and Recognition - ICDAR 2023: 17th International Conference, San José, CA, USA, August 21–26, 2023, Proceedings, Part III*, page 384–401, San José, CA, USA. Springer Nature Switzerland.
- Marcin Heliński, Miłosz Kmiecik, and Tomasz Parkoła. 2012. Report on the comparison of Tesseract and ABBYY FineReader OCR engines.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.
- Anni Järvelin, Heikki Keskustalo, Eero Sormunen, Miamaria Saastamoinen, and Kimmo Kettunen. 2016. Information retrieval from historical newspaper collections in highly inflectional languages: A query expansion approach. *Journal of the Association for Information Science and Technology*, 67(12):2928–2946.
- Philip Kahle, Sebastian Colutto, Günter Hackl, and Günter Mühlberger. 2017. Transkribus - a service platform for transcription, recognition and retrieval of historical documents. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 04, pages 19–24, Kyoto, Japan. IEEE.
- Rutger van Koert, Stefan Klut, Tim Koornstra, Martijn Maas, and Luke Peters. 2024. Loghi: An end-to-end framework for making historical documents machine-readable. In *Document Analysis and Recognition – ICDAR 2024 Workshops*, pages 73–88, Athens, Greece. Springer Nature Switzerland.
- Mika Koistinen, Kimmo Kettunen, and Tuula Pääkkönen. 2017. Improving optical character recognition of Finnish historical newspapers with a combination of fraktur & antika models and image preprocessing. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 277–283, Gothenburg, Sweden. Association for Computational Linguistics.
- Johanna Laakso and Elena Skribnik. 2022. Graphization and orthographies of Uralic minority languages. In *The Oxford Guide to the Uralic Languages*, pages 91–100. Oxford University Press.
- Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. 2023. TrOCR: Transformer-based optical character recognition with pre-trained models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37 No. 11, pages 13094–13102.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. RoBERTa: A robustly optimized BERT pre-training approach. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Daniel Lopresti. 2008. Optical character recognition errors and their effects on natural language processing. In *Proceedings of the second workshop on Analytics for Noisy Unstructured Text Data*, pages 9–16, New York, NY, United States. Association for Computing Machinery.
- Martin Maarand, Yngvil Beyer, Andre Kåsen, Knut T. Fosseide, and Christopher Kermorvant. 2022. A comprehensive comparison of open-source libraries for handwritten text recognition in norwegian. In *Document Analysis Systems: 15th IAPR International Workshop, DAS 2022*, pages 399–413, La Rochelle, France. Springer International Publishing.
- Ole Henrik Magga. 1994. Hvordan den nyeste nord-samiske rettskrivningen ble til. In *Festskrift til Ørnulv Vorren*. Tromsø Museum, Universitetet i Tromsø, Tromsø, Norway.
- Jamshed Memon, Maira Sami, Rizwan Ahmed Khan, and Mueen Uddin. 2020. Handwritten optical character recognition (OCR): A comprehensive systematic literature review (SLR). *IEEE access*, 8:142642–142668.
- Yngve Mardal Moe and Marie Roald. 2024. Stringalign, version 5499dc8, [Software]. <https://github.com/yngvem/stringalign>.
- Clemens Neudecker, Konstantin Baierer, Mike Gerber, Christian Clausner, Apostolos Antonacopoulos, and Stefan Pletschacher. 2021. A survey of ocr evaluation tools and metrics. In *Proceedings of the 6th International Workshop on Historical Document Imaging and Processing, HIP ’21*, page 13–18, New York, NY, USA. Association for Computing Machinery.
- Marja-Liisa Olthuis, Suvi Kivelä, and Tove Skutnabb-Kangas. 2013. *Revitalising indigenous languages: How to recreate a lost generation*. Multilingual matters, Bristol, UK.
- Niko Partanen and Michael Rießler. 2019. An OCR system for the unified northern alphabet. In *Proceedings of the Fifth International Workshop on Computational Linguistics for Uralic Languages*, pages 77–89, Tartu, Estonia. Association for Computational Linguistics.
- Flammie Pirinen, Sjur Moshagen, and Katri Hiovain-Asikainen. 2023. GiellaLT — a stable infrastructure for Nordic minority languages and beyond. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 643–649, Tórshavn, Faroe Islands. University of Tartu Library.

- Alberto Poncelas, Mohammad Aboomar, Jan Buts, James Hadley, and Andy Way. 2020. A tool for facilitating OCR postediting in historical documents. In *Proceedings of LT4HALA 2020 - 1st Workshop on Language Technologies for Historical and Ancient Languages*, pages 47–51, Marseille, France. European Language Resources Association (ELRA).
- Joan Puigcerver. 2017. Are multidimensional recurrent layers really necessary for handwritten text recognition? In *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, volume 1, pages 67–72, Kyoto, Japan. IEEE.
- Shruti Rijhwani, Daisy Rosenblum, Michayla King, Antonios Anastasopoulos, and Graham Neubig. 2023. User-centric evaluation of OCR systems for kwakʼwala. In *Proceedings of the Sixth Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 19–29, Remote. Association for Computational Linguistics.
- Baoguang Shi, Xiang Bai, and Cong Yao. 2016. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2298–2304.
- SIKOR. 2021. SIKOR UiT the Arctic University of Norway and the Norwegian Saami Parliament’s Saami text collection, version 01.12.2021 [data set]. <http://gtweb.uit.no/korp>.
- Ray Smith. 2007. An overview of the Tesseract OCR engine. In *Ninth international conference on document analysis and recognition (ICDAR 2007)*, volume 2, pages 629–633, Los Alamitos, CA, USA. IEEE, IEEE Computer Society.
- Phillip Benjamin Ströbel, Tobias Hodel, Walter Boente, and Martin Volk. 2023. The adaptability of a transformer-based ocr model for historical documents. In *Document Analysis and Recognition – ICDAR 2023 Workshops*, pages 34–48, San José, CA, USA. Springer Nature Switzerland.
- Ahmad P Tafti, Ahmadreza Baghaie, Mehdi Assefi, Hamid R Arabnia, Zeyun Yu, and Peggy Peissig. 2016. OCR as a service: an experimental evaluation of Google Docs OCR, Tesseract, ABBYY FineReader, and Transym. In *Advances in Visual Computing 12th International Symposium, ISVC 2016, December 12-14, 2016, Proceedings, Part I 12*, pages 735–746, Las Vegas, NV, USA. Springer.
- Solène Tarride, Yoann Schneider, Marie Generali-Lince, Mélodie Boillet, Bastien Abadie, and Christopher Kermorvant. 2024. Improving automatic text recognition with language models in the PyLaia open-source library. In *Document Analysis and Recognition - ICDAR 2024*, pages 387–404. Springer Nature Switzerland.
- Haoran Wei, Chenglong Liu, Jinyue Chen, Jia Wang, Lingyu Kong, Yanming Xu, Zheng Ge, Liang Zhao, Jianjian Sun, Yuang Peng, Chunrui Han, and Xiangyu Zhang. 2024. General OCR theory: Towards OCR-2.0 via a unified end-to-end model. <https://arxiv.org/abs/2409.01704>.
- 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.
- Blnd Yaseen and Hossein Hassani. 2024. Making Old Kurdish publications processable by augmenting available optical character recognition engines. <https://arxiv.org/abs/2404.06101>.

LAG-MMLU: Benchmarking Frontier LLM Understanding in Latvian and Giriama

Naome A. Etori¹, Arturs Kanepajs², Kevin Lu³, and Randu Karisa²

¹University of Minnesota-Twin Cities

²Independent Researcher

³Bellarmino College Preparatory

etori001@umn.edu

Abstract

This paper evaluates the language understanding capabilities of various large language models (LLMs) through an analysis of 112 translated and human-edited questions from the Multitask Language Understanding (MMLU) dataset, focusing specifically on two underrepresented languages: Latvian and Giriama. The study compares the performance of six state-of-the-art (SOTA) models, with OpenAI's o1-preview model demonstrating superior performance across all languages, significantly outperforming non-proprietary models in Latvian and Giriama. Human editing of automated translations from English to Latvian yielded only a small, statistically insignificant improvement in performance estimates, suggesting that machine-translated benchmarks may be sufficient for comparing model performance in languages with established digital resources like Latvian. However, automated translation to Giriama proved infeasible, and model performance in Giriama remained poor, highlighting the persistent challenges LLMs face with low-resource languages. These findings underscore the need for high-quality datasets and improved machine translation capabilities for underrepresented languages, emphasizing the importance of localized benchmarks and human evaluation in addressing cultural and contextual limitations in AI models.

1 Introduction

The potential benefits of advanced artificial intelligence (AI) are vast, but to ensure these advantages are globally accessible, it's crucial that AI systems perform well across multiple languages. Previous research has highlighted a significant disparity between the performance of frontier large language

models (LLMs) in English compared to other languages, particularly those with limited resources (Cohere For AI team, 2024; OpenAI, 2024; Dubey et al., 2024).

Recently, there has been growing interest in assessing the capabilities of LLMs, with studies such as HELM (Liang et al., 2022), BIG-Bench (Srivastava et al., 2022), LAMBADA (Paperno et al., 2016) evaluating various model functions. However, these evaluations mostly focus on English, leaving a gap in assessing LLMs' multilingual performance. As new language technologies based on LLMs rapidly emerge, evaluating their multilingual effectiveness is crucial (Blasi et al., 2021).

As AI models continue to evolve, it's essential to monitor how this language gap is narrowing. Users working with models in various languages could greatly benefit from comparative performance analyses across different linguistic contexts. However, evaluating model performance in non-English languages presents challenges, for example manual translation is time-consuming, and this has forced the NLP community to focus on a selection of tasks and languages only. Moreover, it has become standard practice to machine translate the training set but use human translation for test sets (Choenni et al., 2024). While automated translation of benchmarks is cost-effective, it raises concerns about quality. Conversely, human translations, though potentially more accurate, can be prohibitively expensive. Driven by these considerations this study aims to address the following key questions:

- Q1: Which LLM performs best in both Latvian and Giriama tasks?
- Q2: How do model performance levels differ between English, Latvian, and Giriama?
- Q3: How does human post-editing of translations affect benchmark quality compared to pure machine translation?

In our work, we utilize the Massive Multitask Language Understanding (MMLU) benchmark,

which covers 57 subjects ranging from STEM to humanities and social sciences. Our goal is to enhance the understanding of LLMs performance in low-resource languages, with a specific focus on Latvian and Giriama, and to contribute to the development of AI systems that are both linguistically and culturally inclusive.

2 Related works

2.1 Multilingual models across cultures and languages

State-of-the-art (SOTA) massively multilingual language Models (MMLMs) such as mBERT (Devlin, 2018), XLMR (Conneau, 2019), and mT5 (Xue, 2020) support 100+ languages worldwide and have shown exceptional proficiency in both understanding and generating text across diverse linguistic contexts. Additionally, generative models like GPT-4 (Achiam et al., 2023), LLaMA (Touvron et al., 2023) and BLOOM (Le Scao et al., 2023) are also gaining world recognition for their contributions to advancing natural language generation and understanding. Significant challenges remain in ensuring cultural sensitivity and language equity (Dawson et al., 2024).

Studies have shown that multilingual models perform well on high-resource languages like English, French, and German, but struggle with low-resource languages (Li et al., 2024; Hedderich et al., 2020; Ranathunga and De Silva, 2022), particularly in Africa (Adelani et al., 2021; Alabi et al., 2022; Adebara et al., 2024) and South Asia (Lahoti et al., 2022; Baruah et al., 2021), due to limited training data (Adebara et al., 2024; Magueresse et al., 2020). Challenges such as cultural nuances (Romero et al., 2024; Winata et al., 2024), dialectal variation (Faisal et al., 2024), and code-switching (Winata et al., 2021) further hinder model performance. While efforts like cross-lingual transfer learning and culturally relevant datasets have been made to address these issues (Hu et al., 2020; Winata et al., 2022; Liu et al., 2021), performance gaps persist in underrepresented languages.

2.2 Datasets, benchmarks, or libraries for evaluating multi-lingual models

Most existing multilingual NLP benchmarks such as (Hendrycks et al., 2020; Hu et al., 2020; Wang, 2018; Wang et al., 2019; Guzmán et al., 2019) are heavily skewed toward high-resource languages, particularly those in the Indo-European language family, and reflect predominantly Western cultural

contexts. As a result, these benchmarks fail to capture the linguistic and cultural diversity of the global population, making them less reliable in assessing the performance of multilingual language models (MMLMs) across underrepresented languages and cultures (Bender, 2019).

Recent works have focused on expanding multilingual datasets to better reflect the linguistic and cultural diversity across the world. Projects such as (Romero et al., 2024; Winata et al., 2024; Kirby et al., 2016; Miquel-Ribé and Laniado, 2019; Moran et al., 2022; Adebara et al., 2024; Ifeoluwa Adelani et al., 2024; Costa-jussà et al., 2022) are making strides in enhancing the representation of multilingual models, leveraging community-driven initiatives to build localized datasets. These efforts have highlighted the importance of understanding the cultural context in which language is used, rather than relying solely on translation-based approaches (Tiedemann, 2020).

2.3 Human evaluation of multilingual and multicultural aspects of models

Human ability to understand language is general, flexible, and robust (Wang, 2018; Lin and Och, 2004). Hence, human evaluations are typically considered the gold standard in natural language generation to assess the effectiveness of multilingual models (Clark et al., 2021; Chiang and Lee, 2023), particularly in evaluating their ability to generate text that aligns with diverse linguistic and cultural contexts. Automatic metrics such as BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) even though commonly used, often fail to capture cultural nuances, making human evaluation essential for a more comprehensive assessment (Kocmi et al., 2021).

Human evaluations are essential for assessing how well multilingual models handle grammatical, syntactical, and contextual differences, particularly in low-resource languages where machine models often struggle with culturally specific terms (Costa-jussà et al., 2022). Evaluating multicultural aspects is even more challenging due to cultural references, social norms, and context-dependent meanings. Human raters are better at identifying these nuances, using criteria such as appropriateness, bias detection, and cultural sensitivity (Choenni et al., 2024).

Language	Question and Answers (Question subject: miscellaneous)
English	According to the children’s nursery rhyme what type of ocean did Columbus sail in 1492? A: calm X , B: blue ✓, C: windy X , D: really big X
Giriama	Kulingana na wira wa kitalu cha ahoho ni aina yani ya bahari ambayo Columbus wasafiri makathi ga 1492? A: Kuhurira X , B: buluu ✓, C: peho X , D: bomu jeri X
Latvian (autotranslated)	Saskaņā ar bērnu bērnodārza atskaņu, kāda veida okeānu Kolumbs kuģoja 1492. gadā? A: Mierīgs X , B: zils ✓, C: Vējains X , D: Ļoti liels X
Latvian (autotranslated & edited)	Saskaņā ar bērnodārza pantīnu, kāda veida okeānu Kolumbs kuģoja 1492. gadā? A: Mierīgu X , B: Zilu ✓, C: Vējainu X , D: Ļoti lielu X

Table 1: Sample question translated into Giriama and Latvian (AT: autotranslated, AT+E: autotranslated and edited) with correct answers marked (✓) and incorrect answers marked (X). The correct answer "blue" in English refers to the popular children’s rhyme "In 1492, Columbus sailed the ocean blue," which is a cultural reference that may not resonate in Latvian or Giriama without further explanation.

3 Methodology

3.1 Datasets

The MMLU dataset (Hendrycks et al., 2021) includes 57 subjects spanning various disciplines such as mathematics, history, computer science, law, and more. The dataset features over 15,000 questions from publicly available sources such as practice tests for exams like the GRE and USMLE. These questions are categorized by difficulty, from elementary to advanced professional levels. The benchmark is designed to evaluate models in zero-shot and few-shot settings, aiming to assess their world knowledge and problem-solving ability across diverse subjects.

3.2 Languages covered

Our benchmarks encompass Latvian and Giriama, two languages that are quite distinct both in their geographic origins and linguistic structures:

- **Latvian (lav)**: spoken by approximately 1.75 million people primarily in Latvia, belongs to the Baltic branch of the Indo-European language family and is closely related to Lithuanian, though they are not mutually intelligible. Latvian has lower digital resources as compared to high-resource languages like English, German, or Chinese and limited representation in widely used multilingual benchmarks. The complexity of Latvian, such as its rich morphology (seven cases, gender system, and inflectional forms), further adds to the difficulty of processing it with LLMs, which often struggle with the intricate grammatical structures of low- and medium-resource languages. It remains underrepresented in many NLP applications (Darģis et al., 2024).
- **Giriama (nyf)**: Giriama, or Kigiryama, is a Bantu language spoken by around 700,000 people, primarily in Kilifi County, Kenya. It

is one of the nine (9) Mijikenda languages, classified under the Northeastern Bantu subgroup of the Niger-Congo family. Like many Bantu languages, Giriama is agglutinative, using affixes to express grammatical relations, and features a complex noun class system that affects agreement with verbs and adjectives. Predominantly oral, Giriama has limited written texts, though recent efforts have promoted literacy using the Latin alphabet. Despite these efforts, Giriama remains under-resourced in linguistic and digital documentation.

3.2.1 Dataset collection

We created our dataset by randomly selecting 112 questions and answers from the MMLU (Massive Multitask Language Understanding) benchmark (Hendrycks et al., 2021). The dataset preparation involved three versions:

1. The original English questions (baseline)
2. Machine translations of these questions into Latvian using MyMemory API (MyMemory, 2024)
3. Human-edited translations in both Latvian and Giriama

For Giriama, we skipped machine translation since automatic translation systems frequently misidentified the language as Swahili. This three-version approach enabled us to compare LLM performance across machine-translated and human-edited content.

3.2.2 Translations and annotation process

We recruited one language coordinator, who also doubled as a translator for the Giriama language. The translator holds a master’s degree in computer science and is a native speaker of the language, with extensive experience as a translator. As a token of appreciation, we provided compensation for

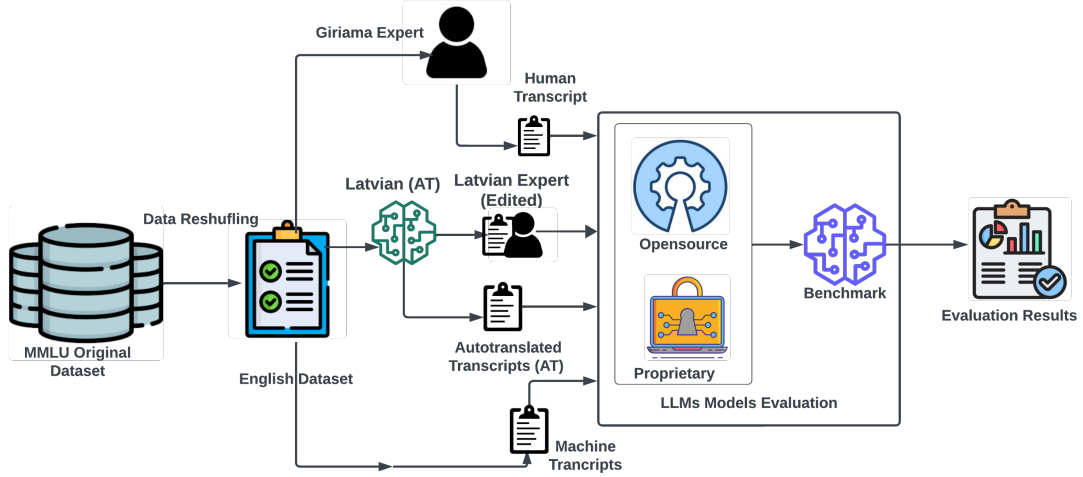


Figure 1: Frontier LLMs in Latvian and Giriama Dataset and Benchmarking pipeline

the work completed. For the Latvian translations, a Latvian-fluent annotator reviewed and edited the machine-translated questions. The focus was on correcting any errors that could hinder comprehension or lead to misinterpretations of the answer options. This human-edited process ensured a higher level of accuracy in both the Giriama and Latvian translations.

3.3 Task covered

Our work focuses on evaluating the multilingual understanding of LLMs by assessing their ability to process translated questions from the MMLU benchmark across three languages: English, Latvian, and Giriama. The translation tasks involve both machine-generated and human-annotated translations. Specifically, the task examines how well the models comprehend and answer 112 questions from English into Latvian and Giriama. The objective is to compare the performance of LLMs in handling machine translations versus human-annotated versions, thereby exploring the necessity and impact of human involvement in translation tasks, particularly in low-resource languages like Giriama.

4 Evaluation metrics

We evaluated the performance of six LLMs on four distinct language tasks: English, Latvian, machine-translated Latvian, and Giriama using an accuracy score. A uniform temperature setting of 0.5 was applied across all models, except for the o1-preview, for which only a fixed temperature of 1 was supported.

For each model, accuracy was computed as the proportion of correct outputs from a test set comprising 112 samples. To account for uncertainty in the performance estimates, we employed the Wilson score interval. This method provides a more accurate estimation of confidence intervals for binomial proportions p such as model accuracy by considering the sample size n and desired confidence level (typically set at $z = 1.96$ for a 95% confidence interval). The Wilson interval is preferred over traditional intervals like Wald due to its robustness, particularly with smaller sample sizes, ensuring more reliable confidence bounds around the accuracy metric.

We tested statistical significance using a two-proportion z-test, comparing each model’s performance against the highest-performing model in its respective task category. This approach allowed us to ascertain whether differences in accuracy were statistically significant or occurred due to random chance. The evaluation process leveraged the UK AISI Inspect framework (AI Safety Institute, 2024), which provided a standardized structure for implementing and automating our assessment.

5 Experiment

5.1 Model choice

We employed a combination of six (6) closed and open large LLMs to evaluate their performance across English, Latvian, and Giriama translations.¹ The closed models selected for this study include

¹Specifically: claude-3-5-sonnet-20241022, gemini-1.5-pro-002, gpt-4o-2024-08-06, Meta-Llama-3.1-405B-Instruct-Turbo, mistral-large-2407, and o1-preview-2024-09-12.

o1-preview, GPT-4o, and versions of Claude and Gemini, all of which are proprietary models known for their SOTA performance and extensive use in commercial applications. These models were chosen due to their established capabilities in handling a wide range of tasks, particularly in high-resource languages like English.

In contrast, open models such as Llama and Mistral were also included in the evaluation due to the transparency regarding their underlying architecture and training data, hence valuable for our use case. We aim to provide a comprehensive comparison of their effectiveness in low-resource languages, while also exploring the potential trade-offs between proprietary solutions and more customizable, open models.

6 Results and discussions

6.1 Model performance on languages

Table 2 presents the performance results of six LLMs across four languages—English, Latvian, machine-translated Latvian (denoted as Latvian (AT)), and Giriama. The results reflect varying degrees of proficiency across these languages, with a notable performance disparity between high-resource (English) and low-resource (Latvian and Giriama) languages.

The **o1-preview** model demonstrated superior performance across all three languages, achieving an accuracy of 87.5% in English, 84.8% in Latvian, and 82.1% in machine-translated Latvian. While the model’s performance declined in Giriama, it still led the other models with an accuracy of 64.3%, suggesting relative robustness in handling lower-resource languages. The relatively small performance gap between English and Latvian shows the model’s effectiveness in transferring knowledge to a non-English, medium-resource language.

Mistral showed the weakest performance across all languages, with English accuracy at 76.8% and a sharp decline in Latvian (57.1%**) and Giriama (34.8%**). This underscores the challenges of Mistral model in processing low-resource languages and its inability to maintain consistent accuracy across diverse linguistic contexts.

o1-preview model demonstrates the highest performance in Giriama, though the differences between o1, GPT-4o, Claude 3.5 Sonnet, and Gemini 1.5 Pro are statistically insignificant. In contrast, Llama 3.1 405B and Mistral Large 2 show notably lower performance, struggling to handle Giriama and Latvian

6.2 Cross-language performance gaps

The performance disparities observed between English (Figure 2), Giriama (Figure 3) and Latvian (Figure 4) underscore the challenges faced by current LLMs in processing both medium- and low-resource languages.

The average gap between English and Latvian performance across all models is 9.3%, which is comparable to approximately two-thirds of the performance difference between GPT-3.5 and GPT-4 in English (OpenAI et al., 2024). However, this gap narrows for higher-performing models like **o1-preview**, where the difference becomes less pronounced. Large differences in this gap are primarily observed in the performance of **Mistral**.

In contrast, Giriama—a low-resource Bantu language—exhibits a much more pronounced performance gap, with average model accuracy dropping sharply to (47.6%), underscoring the limitations of cross-lingual transfer learning in handling languages with limited digital resources and complex linguistic structures.

The results reveal a consistent performance gap between more resourced languages and less resourced languages. On average, the models perform best in English (83.6%), followed by Latvian (74.3%) and machine-translated Latvian (71.3%), with the lowest performance observed in Giriama (47.6%).

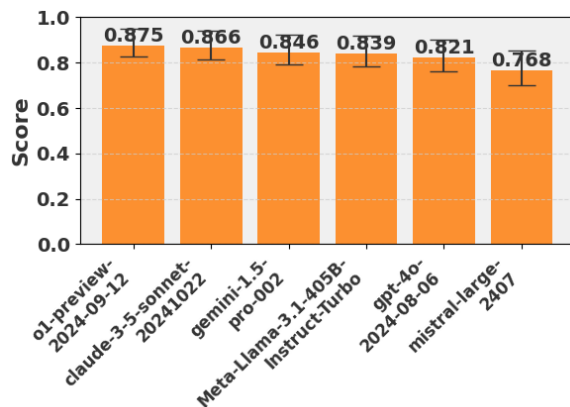


Figure 2: **Model performance in English.** Error bars represent 95% Wilson confidence intervals.

6.3 Impact of human-edited vs. machine-translated Data

For Latvian translations, human editing provided a modest improvement over machine translation, with accuracy increasing by 3.0% on average (see

Model	English	Latvian	Latvian (AT)	Giriama
o1-preview-2024-09-12	0.875	0.848	0.821	0.643
claude-3-5-sonnet-20241022	0.866	0.804	0.777	0.482*
gemini-1.5-pro-002	0.846	0.786	0.732	0.509*
Meta-Llama-3.1-405B-Instruct-Turbo	0.839	0.688**	0.643**	0.411***
gpt-4o-2024-08-06	0.821	0.759	0.723	0.464**
mistral-large-2407	0.768*	0.571***	0.580***	0.348***
AVG	0.836	0.743	0.713	0.476

Table 2: Model performance across languages. AT: autotranslated. Each model: n=112; AVG: n=672. Boldface indicates the highest score in each column. Asterisks indicate statistically significant differences from the highest-scoring model within each language variant (*: $p<0.05$, **: $p<0.01$, ***: $p<0.001$), computed using two-proportion z-test.

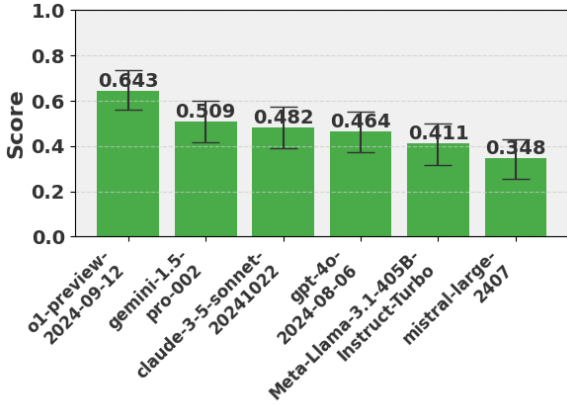


Figure 3: Model performance in Giriama. Error bars represent 95% Wilson confidence intervals.

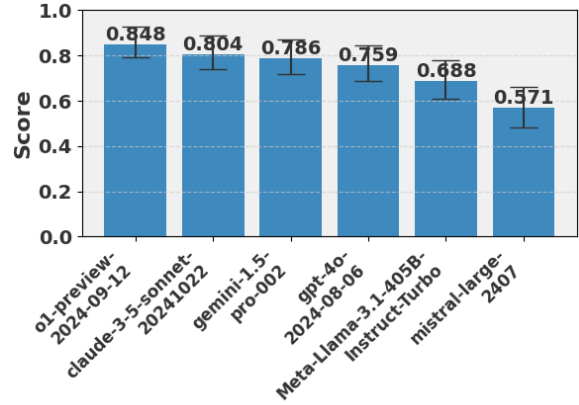


Figure 4: Model performance in Latvian. Error bars represent 95% Wilson confidence intervals.

Table 2). While this difference is not statistically significant, it suggests that human involvement remains valuable for languages with complex morphology and syntax. However, we note that our baseline used a free translation service - SOTA machine translation might further narrow this gap.

Giriama presented more significant challenges. The language was consistently misidentified as Swahili by translation systems, making automatic translation infeasible. This technical limitation, combined with uniformly poor model performance across all tested models, emphasizes the need for increased linguistic resources and human expertise when working with low-resource African languages.

6.4 Impact of the temperature setting

As noted previously, we used a temperature setting of 0.5 for all models except for o1-preview, which only permits temperature=1. To assess whether this non-uniform temperature setting significantly affected our results, we conducted addi-

tional tests on the edited Latvian translations. We ran the five other models at temperature=1 and compared their performance against the temperature=0.5 runs. The results were similar across models, with only Llama 3.1 405B showing slight improvement (+0.9%). On average, performance marginally declined (-0.4%), but none of the differences were statistically significant. We conclude that the non-uniform temperature settings did not materially impact our findings.

6.5 Implications for multilingual LLM development

The substantial performance drop from English to Giriama across all models reflects the broader challenges in scaling LLMs for low-resource languages. While advancements in multilingual modeling have closed some gaps for medium-resource languages like Latvian (Dargis et al., 2024), this study highlights the considerable distance yet to be covered in adequately supporting low-resource languages, particularly African languages like Giriama. These

results underscore the importance of developing more inclusive benchmarks and expanding the availability of high-quality training data to ensure that LLMs are more equitable across diverse linguistic contexts.

6.6 Bias and considerations for future Research

Anecdotal evidence showed that some of the tested models were much better at translating questions and answers than the free translation service. Future research could make use of the LLM translation capabilities. However, it is important not to bias the results in favor of one model or another: it is not inconceivable that a given model finds its own translations easier to interpret than those of other models (which is another hypothesis to explore). Alternatively, it is possible to use other translation services and human translation services together or separately.

These, as other benchmark results, may be subject to bias due to potential data contamination. (Bean et al., 2024). The English MMLU dataset is more likely to have been included in or influenced the models' training data. This could lead to an overestimation of the performance gap between languages, as models might have prior exposure to the English questions.

Cultural context introduces another potential source of bias and reduced relevance in this study. For example, Professional Law questions are based on the U.S. legal system, not Kenyan or Latvian law, which may lead to less accurate responses when questions are presented in Giriama or Latvian. This mismatch between the source material's cultural context and the target languages could affect model performance independently of linguistic factors. Future research could assess the impact of cultural context by using a larger sample size and analyzing model performance in culturally sensitive subcategories like Professional Law. However, U.S.-centric legal questions are inherently limited in evaluating legal expertise within other contexts. Adapting such questions to local contexts is crucial but may require costly specialist knowledge.

Expanding the sample size in future studies could yield more robust results. The scope of this investigation was primarily constrained by two factors: the human resources required for editing translations and the available resources for model API access.

7 Conclusion

Our evaluation of six frontier LLMs across English, Latvian, and Giriama reveals several critical insights about the current state of multilingual AI capabilities:

1. **Model-specific language gaps:** While all models showed performance degradation in non-English languages, proprietary models (particularly o1-preview with only a 2.7% English-Latvian gap) maintained relatively consistent performance compared to open-source alternatives (up to 19.7% gap). This suggests that recent advances in commercial AI systems are beginning to address historical English-centric bias, though significant gaps remain in open-source alternatives.
2. **Translation quality impact:** For Latvian, human editing of machine translations improved accuracy by only 3.0% on average, indicating that automated translations may be sufficient for benchmark creation in languages with established digital infrastructure. This finding could significantly reduce the cost and effort of developing multilingual evaluations.
3. **Low-resource language challenges:** The dramatic performance drop in Giriama (average accuracy 47.6% vs 83.6% in English) reveals fundamental limitations in current approaches to low-resource language support. The failure of machine translation for Giriama highlights how technological gaps compound the challenges of language accessibility.

These findings have immediate implications for both research and deployment. For research, they highlight the viability of using machine translation for creating benchmarks in medium-resource languages and the need for better methods to support low-resource languages. For deployment, our results suggest that while LLMs are becoming viable for medium-resource languages like Latvian, significant work remains before they can reliably serve low-resource language communities.

Future work should prioritize two key areas: (1) developing more efficient methods for extending LLM capabilities to low-resource languages without requiring extensive compute or data resources, and (2) creating evaluation frameworks that explicitly measure both linguistic accuracy and cultural appropriateness. The substantial gap in low-resource language performance emphasizes that achieving truly equitable AI requires not just tech-

nical advancement, but sustained investment in linguistic resources and community engagement.

8 Limitations

Our work presents several limitations that should be acknowledged. First, no formal quality control measures, such as inter-annotator agreement (IAA) or Cohen’s Kappa, were employed to assess the consistency and reliability of the translations in our dataset. This could affect the overall validity of the translation quality.

The dataset size is relatively small, consisting of only 500 questions per language. While this dataset provides preliminary insights, the dataset size limits the generalizability of the results, and larger datasets would be necessary to draw more robust conclusions.

This study’s scope was limited to six language models and two non-English languages due to API access costs. A more comprehensive evaluation would require greater financial resources to test additional models and languages.

Finally, Giriama, as a low-resource language, faces unique challenges due to limited linguistic resources, which may lead to oversimplified translations and insufficient validation, affecting the dataset’s quality. Unlike Latvian, which has more established digital resources, Giriama may lack the tools for thorough quality control, increasing the risk of inaccuracies.

9 Ethical considerations

Native speakers translated the MMLU dataset into Giriama and Latvian to ensure linguistic and cultural accuracy. However, several potential ethical concerns arise in this process:

- **Cultural Relevance and Sensitivity:** While linguistic fidelity was prioritized, the dataset contains many questions grounded in Western, specifically American, cultural contexts such as historical references to Columbus or moral standards in the US. When translating such questions into Latvian or Giriama, there is a risk of imposing culturally foreign concepts onto the target audience, potentially alienating speakers or distorting meaning. For instance, some questions may have no direct equivalent in Giriama or Latvian law and moral philosophy. This can lead to mistranslation or misunderstanding, as the target audience may not relate to or fully grasp the original cultural context.

- **Linguistic Complexity and Vocabulary Gaps:** Many questions in the dataset involve highly technical and specialized terminology from subjects such as law, science, and ethics (such as "neurotransmitters," "Pauli exclusion principle"). Low-resource languages like Giriama may not have established vocabulary for such specialized terms, resulting in challenges for accurate translation. Translators must decide whether to borrow terms from English or create new ones, both of which have ethical implications that could undermine linguistic purity or lead to confusion or lack of consistency in the target language.
- **Cultural Bias in Translation:** The MMLU dataset reflects Western-centric knowledge and perspectives, which pose ethical challenges when translating into low-resource languages like Giriama or Latvian. Without careful adaptation, cultural differences in political ideologies, social norms, or gender roles may be misrepresented, leading to misunderstandings. These biases can hinder the performance of language models by failing to accurately capture the nuances of the target cultures, reducing their effectiveness in real-world applications.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Ife Adebara, AbdelRahim Elmadany, and Muhammad Abdul-Mageed. 2024. Cheetah: Natural language generation for 517 african languages. *arXiv preprint arXiv:2401.01053*.
- David Ifeoluwa Adelani, Jade Abbott, Graham Neubig, Daniel D’souza, Julia Kreutzer, Constantine Lignos, Chester Palen-Michel, Happy Buzaaba, Shruti Rijhwani, Sebastian Ruder, et al. 2021. Masakhaner: Named entity recognition for african languages. *Transactions of the Association for Computational Linguistics*, 9:1116–1131.
- AI Safety Institute. 2024. Inspect - ai safety institute. <https://inspect.ai-safety-institute.org/>

ai-safety-institute.org.uk/.
Accessed: 2024-10-15.

- Jesujoba O Alabi, David Ifeoluwa Adelani, Marius Mosbach, and Dietrich Klakow. 2022. Adapting pre-trained language models to african languages via multilingual adaptive fine-tuning. *arXiv preprint arXiv:2204.06487*.
- Rupjyoti Baruah, Rajesh Kumar Mundotiya, and Anil Kumar Singh. 2021. Low resource neural machine translation: Assamese to/from other indo-aryan (indic) languages. *Transactions on Asian and Low-Resource Language Information Processing*, 21(1):1–32.
- Andrew M. Bean, Simi Hellsten, Harry Mayne, Jabez Magomere, Ethan A. Chi, Ryan Chi, Scott A. Hale, and Hannah Rose Kirk. 2024. LINGOLY: A Benchmark of Olympiad-Level Linguistic Reasoning Puzzles in Low-Resource and Extinct Languages. *eprint*: 2406.06196.
- Emily Bender. 2019. The# benderrule: On naming the languages we study and why it matters. *The Gradient*, 14:34.
- Damian Blasi, Antonios Anastasopoulos, and Graham Neubig. 2021. Systematic inequalities in language technology performance across the world’s languages. *arXiv preprint arXiv:2110.06733*.
- Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? *arXiv preprint arXiv:2305.01937*.
- Rochelle Choenni, Sara Rajaei, Christof Monz, and Ekaterina Shutova. 2024. On the evaluation practices in multilingual nlp: Can machine translation offer an alternative to human translations? *arXiv preprint arXiv:2406.14267*.
- Elizabeth Clark, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A Smith. 2021. All that’s human’s not gold: Evaluating human evaluation of generated text. *arXiv preprint arXiv:2107.00061*.
- Cohere For AI team. 2024. Policy Primer - The AI Language Gap.
- A Conneau. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.
- Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.
- Roberts Darġis, Arturs Znotins, Ilze Auziņa, Baiba Saulīte, Sanita Reinsone, Raivis Dejus, Antra Klavinska, and Normunds Gruzitis. 2024. Balsutalka. lv-boosting the common voice corpus for low-resource languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2080–2085.
- Fifi Dawson, Zainab Mosunmola, Sahil Pocker, Raj Abhijit Dandekar, Rajat Dandekar, and Sreedath Panat. 2024. Evaluating cultural awareness of llms for yoruba, malayalam, and english. *arXiv preprint arXiv:2410.01811*.
- Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and others. 2024. The Llama 3 Herd of Models. *arXiv preprint arXiv:2407.21783*.
- Fahim Faisal, Orevaoghene Ahia, Aarohi Srivastava, Kabir Ahuja, David Chiang, Yulia Tsvetkov, and Antonios Anastasopoulos. 2024. Dialectbench: A nlp benchmark for dialects, varieties, and closely-related languages. *arXiv preprint arXiv:2403.11009*.
- Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc’Aurelio Ranzato. 2019. The flores evaluation datasets for low-resource machine translation: Nepali-english and sinhala-english. *arXiv preprint arXiv:1902.01382*.
- Michael A Hedderich, Lukas Lange, Heike Adel, Jannik Strötgen, and Dietrich Klakow. 2020. A survey on recent approaches for natural language processing in low-resource scenarios. *arXiv preprint arXiv:2010.12309*.

- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multi-task language understanding. *arXiv preprint arXiv:2009.03300*.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding.
- Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR.
- David Ifeoluwa Adelani, Jessica Ojo, Israel Abebe Azime, Jian Yun Zhuang, Jesujoba O Alabi, Xuanli He, Millicent Ochieng, Sara Hooker, Andiswa Bukula, En-Shiun Annie Lee, et al. 2024. Irokobench: A new benchmark for african languages in the age of large language models. *arXiv e-prints*, pages arXiv–2406.
- Kathryn R Kirby, Russell D Gray, Simon J Greenhill, Fiona M Jordan, Stephanie Gomes-Ng, Hans-Jörg Bibiko, Damián E Blasi, Carlos A Botero, Claire Bower, Carol R Ember, et al. 2016. D-place: A global database of cultural, linguistic and environmental diversity. *PloS one*, 11(7):e0158391.
- Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. *arXiv preprint arXiv:2107.10821*.
- Pawan Lahoti, Namita Mittal, and Girdhari Singh. 2022. A survey on nlp resources, tools, and techniques for marathi language processing. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(2):1–34.
- Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2023. Bloom: A 176b-parameter open-access multilingual language model.
- Zihao Li, Yucheng Shi, Zirui Liu, Fan Yang, Ninghao Liu, and Mengnan Du. 2024. Quantifying multilingual performance of large language models across languages. *arXiv preprint arXiv:2404.11553*.
- Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.
- 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.
- Chin-Yew Lin and Franz Josef Och. 2004. ORANGE: a method for evaluating automatic evaluation metrics for machine translation. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 501–507, Geneva, Switzerland. COLING.
- Zihan Liu, Genta I Winata, Samuel Cahyawijaya, Andrea Madotto, Zhaojiang Lin, and Pascale Fung. 2021. On the importance of word order information in cross-lingual sequence labeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13461–13469.
- Alexandre Magueresse, Vincent Carles, and Evan Heetderks. 2020. Low-resource languages: A review of past work and future challenges. *arXiv preprint arXiv:2006.07264*.
- Marc Miquel-Ribé and David Laniado. 2019. Wikipedia cultural diversity dataset: A complete cartography for 300 language editions. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 13, pages 620–629.
- Steven Moran, Christian Bentz, Ximena Gutierrez-Vasques, Olga Pelloni, and Tanja Samardzic. 2022. Teddi sample: Text data diversity sample for language comparison and multilingual nlp. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1150–1158.
- MyMemory. 2024. Mymemory translation memory - api documentation.

<https://mymemory.translated.net/doc/spec.php>. Accessed: 2024-10-15.

OpenAI. 2024. O1 system card. Technical report, OpenAI. Accessed on October 16, 2024.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, 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 Kafan, 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, Rameesh 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. 2024. GPT-4 Technical Report.

- _eprint: 2303.08774.
- Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The lambda dataset: Word prediction requiring a broad discourse context. *arXiv preprint arXiv:1606.06031*.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. pages 311–318.
- Surangika Ranathunga and Nisansa De Silva. 2022. Some languages are more equal than others: Probing deeper into the linguistic disparity in the nlp world. *arXiv preprint arXiv:2210.08523*.
- David Romero, Chenyang Lyu, Haryo Akbarianto Wibowo, Teresa Lynn, Injy Hamed, Aditya Nanda Kishore, Aishik Mandal, Alina Dragonetti, Artem Abzaliev, Atnafu Lambebo Tonja, et al. 2024. Cvqa: Culturally-diverse multilingual visual question answering benchmark. *arXiv preprint arXiv:2406.05967*.
- Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*.
- Jörg Tiedemann. 2020. The tatoeba translation challenge—realistic data sets for low resource and multilingual mt. *arXiv preprint arXiv:2010.06354*.
- 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*.
- Alex Wang. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.
- Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Super-glue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32.
- Genta Winata, Shijie Wu, Mayank Kulkarni, Thamar Solorio, and Daniel Preotiuc-Pietro. 2022. Cross-lingual few-shot learning on unseen languages. 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 777–791.
- Genta Indra Winata, Samuel Cahyawijaya, Zihan Liu, Zhaojiang Lin, Andrea Madotto, and Pascale Fung. 2021. Are multilingual models effective in code-switching? *arXiv preprint arXiv:2103.13309*.
- Genta Indra Winata, Frederikus Hudi, Patrick Amadeus Irawan, David Anugraha, Rifki Afina Putri, Yutong Wang, Adam Nohejl, Ubaidillah Ariq Prathama, Nedjma Ousidhoum, Afifa Amriani, et al. 2024. World-cuisines: A massive-scale benchmark for multilingual and multicultural visual question answering on global cuisines. *arXiv preprint arXiv:2410.12705*.
- L Xue. 2020. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

Better Benchmarking LLMs for Zero-Shot Dependency Parsing

Ana Ezquerro, Carlos Gómez-Rodríguez, David Vilares

Universidade da Coruña, CITIC

Departamento de Ciencias de la Computación y Tecnologías de la Información

Campus de Elviña s/n, 15071

A Coruña, Spain

{ana.ezquerro, carlos.gomez, david.vilares}@udc.es

Abstract

While LLMs excel in zero-shot tasks, their performance in linguistic challenges like syntactic parsing has been less scrutinized. This paper studies state-of-the-art open-weight LLMs on the task by comparing them to baselines that do not have access to the input sentence, including baselines that have not been used in this context such as random projective trees or optimal linear arrangements. The results show that most of the tested LLMs cannot outperform the best uninformed baselines, with only the newest and largest versions of LLaMA doing so for most languages, and still achieving rather low performance. Thus, accurate zero-shot syntactic parsing is not forthcoming with open LLMs.

1 Introduction

Autoregressive large language models (LLMs) and instruction-based variants (Jiang et al., 2023; OpenAI, 2024; Dubey et al., 2024) are known for their zero-shot and few-shot abilities (Radford et al., 2019). In practical terms, they can serve as versatile systems whose behavior is easily adapted through prompting. Beyond what we experience as everyday users, documented examples in the context of natural language processing (NLP) include question answering (Baek et al., 2023; Li et al., 2024), summarization (Wang et al., 2023), machine translation (Johnson et al., 2017; Wang et al., 2021; Zhang et al., 2023) and information retrieval (Zhuang et al., 2023; Adeyemi et al., 2024; Qin et al., 2024), among many other tasks.

Related, syntactic parsing has long explored few-shot learning approaches. Prior to the development of current LLMs, various methods were studied to perform zero-shot or few-shot parsing, and many of these approaches achieved com-

petitive results. These methods focused on factors such as the quality and quantity of annotations (Meehan-Maddon and Nivre, 2019), cross-lingual learning (Xu and Koehn, 2021), multilingual pre-training (Tran and Bisazza, 2019), and treebank difficulty (Søgaard, 2020; Anderson et al., 2021). However, the effectiveness of zero-shot parsing with LLMs remains a topic of debate. Recent work has showed how state-of-the-art LLMs exhibit low performance in syntactic parsing (Bai et al., 2023; Lin et al., 2023), even when designing manual specific prompts (Li et al., 2023; Blevins et al., 2023). Nonetheless, these results have been deemed sufficient to categorize LLMs as potential zero-shot parsers. While some studies (Tian et al., 2024) suggest that multi-stage complex approaches can yield a competitive zero-shot performance, in this work we will focus on single prompt approaches – similar to what works for other NLP tasks – to evaluate to what extent LLMs can perform the task on their own without externally-provided planning. Studies covering these approaches leave a substantial gap in evaluating LLMs on low-resource setups and often omit comparison with uninformed baselines, which are essential for determining whether LLMs achieve accuracy levels meaningfully above chance.

Contribution We address the lack of comparison against uninformed baselines, and include some that have not been proposed before but offer a higher standard than traditional blind baselines, such as left- or right-branching trees. These baselines provide more robust benchmarks and offer a fairer evaluation of LLMs’ potential as zero-shot parsers. We prioritize depth over breadth by evaluating a wide range of LLMs to identify any substantial differences across them - a contribution that, to our knowledge, has not been thoroughly explored in previous work.¹

¹Code available at github.com/anaezquerro/naipar

2 Zero-shot dependency parsing

Next, we review the notation, benchmarks, uninformed baselines, and introduce the LLMs used. Dependency parsing is the task of obtaining the syntactic structure of a sentence as a set of labeled directed relations (*dependencies*) between words. In zero-shot parsing (whether relying on LLMs or other models), the core idea is to perform the dependency parsing task without using task-specific labeled data during either the pre-training or fine-tuning steps. This approach contrasts with the standard setup for training dependency parsers, where task-specific labeled data is integral to explicitly teaching models syntactic structures in a supervised learning framework. Instead, zero-shot parsing leverages the model’s general pre-trained knowledge to infer syntactic relationships in unseen data. Before the emergence of large autoregressive generative models, pre-training largely avoided task-specific annotations, aligning closely with the zero-shot paradigm. However, given the extensive and diverse nature of the data these models are trained on, it is plausible that some exposure to annotated dependency parsing examples has occurred. This possibility will be examined further in subsequent sections.

Notation Let $W=(w_1, \dots, w_n)$ be a sentence, a dependency graph is defined as $G = (W, A)$, where W is the set of nodes and A the set of arcs. Each arc in A is a tuple (h, d) , where $h \in [0, n]$ is the position of the head node, and $d \neq h \in [1, n]$ the position of the dependent node.² G is a tree T iff (i) is a *connected acyclic* graph, (ii) each word w_i has only one head, so $A = \{(h, d) : d = 1 \dots n\}$, and (iii) there is only one arc of the form $(0, d)$, and w_d is designated as the *root* of the sentence. This work only studies trees.

2.1 Zero-shot (uninformed) baselines

Previous work has reported results on parsing using LLMs (Lin et al., 2023), classifying LLMs as *potentially* zero-shot parsers. We revisit this claim by proposing a comprehensive benchmark and comparing against uninformed baselines (i.e., baselines that generate an output tree without looking at the contents of the input, although sometimes with access to its length). Uninformed baselines are useful to determine whether the

models are meaningfully processing the input or just generating outputs that could be obtained by chance or by using properties that are not specific to the input sentence (e.g. the common trend towards projectivity in human syntax). We use both conventional uninformed baselines that have been used previously in related contexts (e.g. Klein and Manning, 2004) and more sophisticated, though still uninformed, baselines that, to our knowledge, have not yet been applied for this purpose.

2.1.1 Conventional baselines

We now describe baselines that have been used as naive approaches to build simplistic yet valid trees.

Randomized root-based tree generation Our most basic baseline randomly selects a root node, denoted as d' , and creates a dependency from d' to the rest of nodes. Formally, $\hat{A} = \{(0, d'), (d', d) : d = (1, \dots, n) \neq d'\}$.

Right- and left-branching tree generation

This method assigns each word as a dependent of the previous (next) word, with the first (last) word as the root. Right-branching trees are a classic baseline for unsupervised English dependency parsing (Klein and Manning, 2004), as English syntax is predominantly right-branching. The left-branching baseline is included as some languages are predominantly left-branching.

Uniformly random tree generation Another parsing baseline (Klein and Manning, 2004). We use the Aldous (1990) algorithm to guarantee generation of a uniformly random dependency tree.

Sampling from a reference treebank We build the tree distributions of different lengths from a reference treebank. For a sentence, we sampled a dependency tree from the distribution of length n . Note that this is the only one among our baselines that has access to a treebank, although it *is still uninformed* with respect to the input sentence.

2.1.2 Novel uninformed baselines

We refine random tree generation by taking into account observed properties of human language: the scarcity of crossing dependencies (Ferrer-i Cancho et al., 2018) and dependency distance minimization, i.e., the tendency of syntactic structures to minimize the distance between syntactically related words (i.e. the length of dependencies) in order to reduce cognitive processing effort (Liu et al., 2017; Ferrer-i Cancho et al., 2022).

²Graphs have arcs labeled with syntactic functions, but we ignore them here as our evaluation is unlabeled.

Uniformly random projective tree generation

The goal is to generate a projective tree (where dependencies do not cross) uniformly at random. As rejection sampling is too slow, we use Nijenhuis and Wilf (1978)’s algorithm to generate a random unlabeled rooted tree and then assign a random projective arrangement following (Futrell et al., 2015; Alemany-Puig and Ferrer-i Cancho, 2024).³

Uniformly random (projective) optimal-distance tree generation

Again, we start from a uniformly random unlabeled rooted tree. In this case, we give it the linear arrangement that minimizes the sum of dependency distances, using Shiloach (1979)’s algorithm, as well as the minimum-distance *projective* arrangement, with the algorithm by Alemany-Puig et al. (2022).³

2.2 Zero-shot parsing with LLMs

Prompting setup Adopting a strategy similar to Lin et al. (2023), we query LLMs using simple prompts. The prompt includes an introductory sentence requesting output in CoNLL format, followed by a basic example from a reference treebank, where only the ID, HEAD, and DEPREL fields are populated. We selected a random sentence of length 4 to 7 to avoid longer sequences, maintaining a zero-shot setup. Although this may resemble a one-shot setup, the example is intentionally simple, serving only to reduce formatting errors rather than offering linguistic content. Figure 1 breaks down the specific prompt we used.

Postprocessing We account for possible corrupted outputs, such as column mismatches, missing nodes, or multiple roots. From the model’s raw output, we applied two post-processing steps: first, filtering tabular lines and filling fields to match the CoNLL format with correct row and column counts for sentences of length n . Second, we resolved cycles, enforced a unique-root constraint, and replaced out-of-range arcs with root connections to ensure a single-rooted, connected tree. Figure 1 also shows an example of the input and output after the first post-processing step.

3 Experiments

We conduct an in-depth evaluation of LLMs as zero-shot dependency parsers by generating outputs in CoNLL format and comparing them to

³We used the implementation of these algorithms in the LAL library (Alemany-Puig et al., 2021).

Prompt example	
In dependency parsing the CoNLL format for the sentence <The trial begins again Nov 28 .> is:	
1	The _ _ _ _ 2 det _ _
2	trial _ _ _ _ 3 nsubj _ _
3	begins _ _ _ _ 0 root _ _
4	again _ _ _ _ 3 advmod _ _
5	Nov. _ _ _ _ 3 obl:tmod _ _
6	28 _ _ _ _ 5 nummod _ _
7	. _ _ _ _ 3 punct _ _
Now return the CoNLL format for the sentence: <What if Google Morphed Into GoogleOS ?>	
(1) Well-formatted output	
1	What _ _ _ _ 0 nsubj _ _
2	if _ _ _ _ 4 mark _ _
3	Google _ _ _ _ 4 nsubj _ _
4	Morphed _ _ _ _ 0 root _ _
5	into _ _ _ _ 6 case _ _
6	GoogleOS _ _ _ _ 8 nmod _ _
7	? _ _ _ _ 4 punct _ _

Figure 1: Prompt and output after the first post-processing. See Figure 3 for step-by-step process.

uninformed baselines. Unlike Lin et al. (2023), who evaluated only ChatGPT-3.5 due to limited system availability, our work expands the analysis to a broader set of models across a select few languages, albeit on a smaller subset of treebanks. This approach, while time-intensive due to the extensive input and output token requirements, offers a more comprehensive understanding of model performance across different LLMs.

Datasets We selected 4 treebanks from UD 2.14 (Zeman et al., 2024) to conduct experiments in different languages, specifically in English_{EWT}, French_{GSD}, German_{GSD}, and Hindi_{HDTB}.

Evaluation We use the unlabeled attachment score (UAS) and unlabeled exact match (UM) as our primary metrics. For the zero-shot dependency parsers, we report performance after the first post-processing step (ensuring that the CoNLL format file contains all columns) and the second (confirming that the tree is well-formed).

Models We selected several instruction-based models from the Gemma (Gemma Team et al., 2024a,b), LLaMA (Touvron et al., 2023; Dubey et al., 2024), and Mistral (Jiang et al., 2023, 2024) series. Appendix B (Table 3) breaks down the links to all models. All reported results were obtained limiting the inference to half precision.³

4 Analysis of results

Table 1 compares the performance of the tested models with uninformed baselines. We see that only the latest and largest versions of LLaMa (i.e.,

³Preliminary experiments indicated that reduced inference precision had minimal impact on performance.







	English EWT						French GSD						German GSD						Hindi HDTB					
	UAS		UM		%w	UAS	UM		%w	UAS	UM		%w	UAS	UM		%w	UAS	UM		%w			
A	20.74	13.91	100.00	5.99	0.24	100.00	9.79	2.05	100.00	5.92	0.06	100.00	25.34	0.00	100.00	21.34	0.00	100.00	5.67	0.00	100.00			
R	23.30	12.13	100.00	10.67	0.00	100.00	11.59	1.23	100.00	9.38	1.74	100.00	5.59	0.00	100.00	19.99	0.06	100.00	20.06	0.00	100.00			
L	34.41	9.39	100.00	29.78	0.00	100.00	29.59	1.43	100.00	20.53	1.84	100.00	17.92	0.30	100.00									
RD	20.10	10.78	100.00	5.93	0.00	100.00	9.64	1.13	100.00	20.02	2.46	100.00												
RD*	21.45	12.42	100.00	6.06	0.00	100.00	9.38	1.74	100.00															
LI	28.09	11.75	100.00	16.81	0.00	100.00	19.24	1.64	100.00															
LI*	26.99	10.98	100.00	17.08	0.00	100.00	20.53	1.84	100.00															
S	31.14	15.55	100.00	19.43	0.96	100.00	20.02	2.46	100.00															
	v1-2b	15.80	5.23	5.54	7.22	7.17	6.51	0.52	0.00	0.48	9.66	0.91	1.02	0.92	2.15	11.96	-1.03	0.00	0.00	6.92				
	v1-7b	21.26	5.17	6.93	6.88	24.84	14.93	-0.66	0.00	0.00	7.69	16.61	0.99	0.82	1.13	10.75	9.78	-0.08	0.00	0.00	8.43			
	v2-9b	20.20	3.35	6.79	6.50	15.17	13.32	-1.34	0.00	0.72	4.81	14.75	-0.68	0.92	1.13	5.32	12.18	-0.77	0.00	0.06	3.27			
	v2-7b	12.98	10.38	3.18	11.12	24.22	18.20	-1.45	0.00	0.24	1.92	18.70	0.19	0.10	2.05	4.09	10.64	-1.74	0.00	0.00	5.82			
	v2-13b	18.95	4.01	5.83	8.96	14.59	13.64	-0.81	0.00	0.00	22.12	19.40	-1.78	0.00	0.00	32.89	14.77	-1.77	0.00	0.06	18.71			
	v2-70b	13.78	11.38	4.00	10.98	46.89	19.05	-1.42	0.24	0.00	13.70	25.88	-2.41	1.43	0.20	23.34	15.27	-2.33	0.00	0.00	7.47			
	v3-8b	18.34	6.56	7.41	8.09	49.01	14.80	-2.84	0.00	0.00	4.81	29.51	-2.26	1.74	0.10	22.42	17.03	-1.54	0.00	0.06	0.48			
	v3-70b	38.30	0.98	16.37	1.44	58.69	29.20	-0.61	0.96	0.48	33.41	33.91	-1.16	3.17	0.41	28.25	14.24	-1.54	0.00	0.00	26.14			
	v3.1-8b	28.83	-1.31	11.75	2.07	34.38	24.61	-6.35	0.72	0.00	5.29	26.93	-4.67	1.64	0.20	12.90	18.86	-2.11	0.00	0.12	2.43			
	v3.1-70b	39.69	1.46	15.65	2.41	65.86	34.62	-1.30	0.96	0.24	42.79	36.75	-1.14	3.48	0.20	46.57	14.37	-0.69	0.00	0.00	26.14			
	v3.2-1b	15.07	7.35	4.53	8.33	16.75	8.05	-0.65	0.00	0.24	11.06	12.61	-0.9	0.41	1.64	10.03	7.20	-1.9	0.00	0.0	8.37			
	v3.2-3b	18.51	3.98	6.64	6.55	18.68	10.22	-1.1	0.24	0.0	12.74	17.69	0.34	1.13	1.13	10.44	13.84	0.03	0.00	0.0	14.90			
	v1-7b	18.59	3.83	6.55	6.55	16.85	10.32	-0.54	0.00	0.00	5.29	16.37	-1.31	1.23	0.61	5.94	10.63	-0.44	0.00	0.00	0.12			
	v2-7b	23.02	2.10	6.93	6.07	15.12	18.73	-3.28	0.24	0.00	3.85	20.49	-2.90	1.13	0.61	5.02	13.80	-1.11	0.00	0.06	0.36			
	v3-7b	25.04	2.49	7.66	5.39	28.17	27.36	-5.52	0.24	0.48	11.06	28.34	-4.98	1.13	0.51	19.55	19.41	-3.15	0.00	0.00	0.71			
	x1-7b	15.46	3.21	2.63	4.17	26.22	13.00	-1.06	0.24	0.00	3.37	16.67	0.00	0.00	0.00	25.00	13.68	-0.67	0.06	0.00	1.25			
	x1-22b	32.91	0.85	13.72	3.32	57.74	23.75	0.13	0.68	0.24	36.73	22.48	-0.19	2.76	0.92	38.44	19.37	-0.67	0.09	0.26	37.84			
	nemo	20.96	3.89	7.56	7.03	15.74	15.85	-1.49	0.00	0.00	3.61	14.10	-0.57	1.23	0.72	4.09	9.59	-0.06	0.00	0.00	0.48			
	large	28.71	0.81	10.01	4.77	18.25	15.21	-0.58	0.83	0.42	5.00	17.66	0.58	1.74	0.92	7.88	14.46	-1.4	0.00	0.00	26.14			

Table 1: Performance on the test sets. The baselines are: all-to-root (A), left (L) and right (R) branching, random generation (RD), optimal linear arrangement (LI) and sampling (S). The symbol (*) indicates if projectivity is fixed as a constraint. %w is the ratio of outputs that did not require post-processing. We also report results with Gemma () , LLaMA () , and Mistral models() , with versions (v, x) and parameter counts. Subscripts indicate performance boost from the second post-processing step.

the 70B versions of Llama 3 and 3.1) consistently outperform all the baselines in most languages in terms of UAS and UM, and only do so barely (e.g., with the best result on English being about 5.5 points above the left-branching baseline without postprocessing, and close to 7 points with post-processing). The rest of the models clearly fall behind, showing that they are not doing any meaningful parsing at all. In the case of Hindi, no model at all reaches the best baselines. Among our baselines, traditional left (or right in the case of Hindi) branching baselines are the most competitive in terms of UAS,⁴ although baselines based on optimal linear arrangement come close, and the sampling baseline is better in terms of English UM. In Appendix B (Tables 4 to 7) we also include tables showing the individual scores of each model based on the PoS tag of the head in each treebank.

Figure 2 complements Table 1 by illustrating

⁴Superiority of the left-branching baseline on English can be surprising, as right-branching has often been deemed better on English unsupervised parsing (Klein and Manning, 2004; Li et al., 2020); but these papers do not use UD.

the performance of a representative subset of models in terms of dependency displacements (i.e., performance taking into account the difference between the position of the dependent and its head) for the English_{EWT} treebank. We observe that LLaMa v3.1 70B consistently performs better than the sampling and optimal linear arrangement baselines, not only on short dependencies but also on longer rightward dependencies. However, for the rest of the models, the differences with respect to uninformed baselines become subtler. Similar figures for the other evaluated treebanks can be found in the Appendix (Figures 4, 5, and 6).

Overall, the results show that open-weight LLMs are far from being potential zero-shot dependency parsers, contrary to claims about ChatGPT (Lin et al., 2023). Considerable scaling or other improvements would be required for this situation to change.

5 Limitations

Memorization Memorization refers to the LLM’s ability to recall specific patterns, struc-

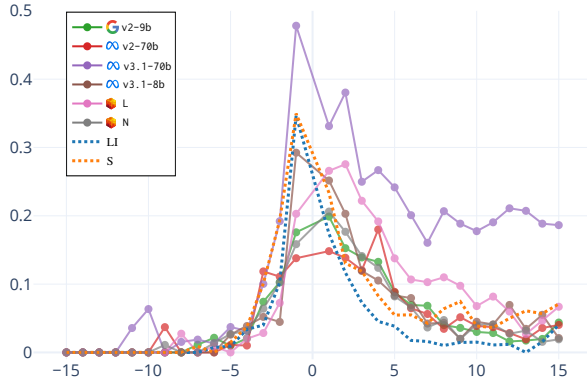


Figure 2: F-score across displacements in the English_{EWT} test set.

tures, or dependencies encountered during pre-training, rather than generalizing to unseen cases (Hartmann et al., 2023). This poses a risk of generalization issues or the regurgitation of chunks of text, which could affect our evaluation but is difficult to quantify (Sainz et al., 2023). Although this is beyond the main scope of our work, we have attempted to briefly analyze this phenomenon. To do so, we crawled a few hundred recent news articles from the New York Times Archive API.⁵ The aim was to collect new text, guaranteeing that no annotations for it were available online when the models were trained. We then produced silver annotations by using a trained graph-based model (Dozat et al., 2017) – a state-of-the-art dependency parser – to parse these articles. In Table 2, we present the results of a few representative models against these silver annotations. The results are consistent with those for UD in Table 1: while UAS scores are lower across the board, this happens both for LLMs and baselines, and likely stems from NYT sentences being longer on average. In relative terms, the same trends as in UD stand, with only LLaMa 3.1-70B clearly outperforming all baselines, so we do not detect evidence of our main results being overestimated due to data contamination.

Prompting The prompting approach used in this study followed a straightforward design. We acknowledge that there may be room for improving parsing performance through more advanced prompt engineering techniques. Our goal was methodological, establishing a set of uninformed baselines rather than optimizing prompt configurations. In this context, approaches such as in-

⁵developer.nytimes.com/docs/archive-product/1



	UAS	UM	%w
A	6.46 _{-14.27}	1.69 _{-12.22}	100
R	24.31 _{1.01}	0.70 _{-11.43}	100
L	20.15 _{-14.25}	2.32 _{-7.07}	100
RD	5.51 _{-14.59}	0.91 _{-11.51}	100
RD*	7.32 _{-14.13}	1.51 _{-10.91}	100
LI	16.39 _{-11.70}	1.77 _{-9.98}	100
LI*	19.54 _{-7.45}	1.40 _{-9.58}	100
S	22.24 _{-8.89}	2.55 _{-13.00}	100
 v3.1-70b	28.83 _{-12.32}	0.00 _{-18.06}	38.72 _{-27.14}
v3.2-3b	8.14 _{-14.35}	0.00 _{-13.19}	11.04 _{-7.64}
 x1-22b	20.37 _{-13.39}	1.00 _{-16.04}	58.92 _{1.18}
large	25.37 _{-4.15}	3.00 _{-11.78}	13.24 _{-5.01}

Table 2: Performance on silver annotations. Subscripts denote the performance drop from Table 1.

context learning (Brown et al., 2020; Chen et al., 2021), chain-of-thought prompting (Wei et al., 2022), and self-consistency (Wang et al., 2022) have shown promise in improving performance by fostering more structured reasoning.

Language selection Our selection of languages was limited to a small set, three of which belong to the Indo-European family. This choice was driven by two key factors. First, although we had the exclusive access to a few 24GB RTX 3090, these were insufficient for running larger models effectively. We also had access to CESGA, the supercomputing center of Galicia; but it was limited to queuing systems, making it difficult to estimate running times and prioritize experiments given the large number of models involved. Additionally, although many models claim to be multilingual, their performance tends to be skewed toward a subset of widely spoken languages. We therefore selected languages that have the most support across models to ensure consistent evaluations.⁶

6 Conclusion

We revisited the potential of autoregressive LLMs as zero-shot dependency parsers. Taking a more conservative approach than previous studies, we compared several LLMs with simple baselines to establish minimal performance benchmarks. Our results show that most LLMs performed on par with uninformed baselines, indicating comparable performance to toy approaches that operate without any access to the input sentence.

⁶Note that not all languages are supported by all models (Table 9). Our selection aims to include widely supported languages to ensure fair comparisons across models.

Acknowledgments

We acknowledge grants SCANNER-UDC (PID2020-113230RB-C21) funded by MICIU/AEI/10.13039/501100011033; GAP (PID2022-139308OA-I00) funded by MICIU/AEI/10.13039/501100011033/ and ERDF, EU; LATCHING (PID2023-147129OB-C21) funded by MICIU/AEI/10.13039/501100011033 and ERDF, EU; and TSI-100925-2023-1 funded by Ministry for Digital Transformation and Civil Service and “NextGenerationEU” PRTR; as well as funding by Xunta de Galicia (ED431C 2024/02), and Centro de Investigación de Galicia “CITIC”, funded by the Xunta de Galicia through the collaboration agreement between the Consellería de Cultura, Educación, Formación Profesional e Universidades and the Galician universities for the reinforcement of the research centres of the Galician University System (CI-GUS). We also extend our gratitude to CESGA, the supercomputing center of Galicia, for granting us access to its resources.

References

- Mofetoluwa Adeyemi, Akintunde Oladipo, Ronak Pradeep, and Jimmy Lin. 2024. Zero-Shot Cross-Lingual Reranking with Large Language Models for Low-Resource Languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 650–656, Bangkok, Thailand. Association for Computational Linguistics.
- David J. Aldous. 1990. The random walk construction of uniform spanning trees and uniform labelled trees. *SIAM J. Discret. Math.*, 3(4):450–465.
- Lluís Alemany-Puig, Juan Luis Esteban, and Ramon Ferrer-i Cancho. 2021. The linear arrangement library. A new tool for research on syntactic dependency structures. In *Proceedings of the Second Workshop on Quantitative Syntax (Quasy, SyntaxFest 2021)*, pages 1–16, Sofia, Bulgaria. Association for Computational Linguistics.
- Lluís Alemany-Puig, Juan Luis Esteban, and Ramon Ferrer i Cancho. 2022. Minimum projective linearizations of trees in linear time. *Information Processing Letters*, 174:106204.
- Lluís Alemany-Puig and Ramon Ferrer-i Cancho. 2024. The expected sum of edge lengths in planar linearizations of trees. *Journal of Language Modelling*, 12(1):1–42.
- Mark Anderson and Carlos Gómez-Rodríguez. 2022. The impact of edge displacement Vaserstein distance on UD parsing performance. *Computational Linguistics*, 48(3):517–554.
- Mark Anderson, Anders Søgaard, and Carlos Gómez-Rodríguez. 2021. Replicating and extending “Because their treebanks leak”: Graph isomorphism, covariants, and parser performance. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 1090–1098, Online. Association for Computational Linguistics.
- Jinheon Baek, Alham Fikri Aji, and Amir Saffari. 2023. Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. In *Proceedings of the First Workshop on Matching From Unstructured and Structured Data (MATCH-ING 2023)*, pages 70–98, Toronto, ON, Canada. Association for Computational Linguistics.
- Xuefeng Bai, Jialong Wu, Yulong Chen, Zhongqing Wang, and Yue Zhang. 2023. Constituency parsing using LLMs. *Preprint*, arXiv:2310.19462.
- Terra Blevins, Hila Gonen, and Luke Zettlemoyer. 2023. Prompting Language Models for Linguistic Structure. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6649–6663, Toronto, Canada. 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. Curran Associates, Inc.
- Yanda Chen, Ruiqi Zhong, Sheng Zha, George Karypis, and He He. 2021. Meta-learning via language model in-context tuning. *arXiv preprint arXiv:2110.07814*.
- Timothy Dozat, Peng Qi, and Christopher D. Manning. 2017. Stanford’s graph-based neural dependency parser at the CoNLL 2017 shared task. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 20–30, Vancouver, Canada. Association for Computational Linguistics.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang,

Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnston, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kam-badur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Al-rassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shao-liang Nie, Sharan Narang, Sharath Rapparth, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei

Chu, Wenhan Xiong, Wenxin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuwei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yun-ing Mao, Zacharie Del-pierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkan Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark,

- Mike Macey, Mike Wang, Miquel Jubert Her-moso, Mo Metanat, Mohammad Rastegari, Mun-ish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratan-chandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sar-gun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Sheng-hao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robin-son, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vitor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mi-hailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xi-aocheng Tang, Xiaofang Wang, Xiaojuan Wu, Xi-aolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yan-jun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. 2024. The Llama 3 Herd of Models. *Preprint*, arXiv:2407.21783.
- Ramon Ferrer-i-Cancho, Carlos Gómez-Rodríguez, and Juan Luis Esteban. 2018. Are crossing depen-dencies really scarce? *Physica A: Statistical Me-chanics and its Applications*, 493:311–329.
- Ramon Ferrer-i Cancho, Carlos Gómez-Rodríguez, Juan Luis Esteban, and Lluís Alemany-Puig. 2022. Optimality of syntactic dependency distances. *Phys-ical Review E*, 105(1).
- Richard Futrell, Kyle Mahowald, and Edward Gibson. 2015. Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, 112(33):10336–10341.
- 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 Tac-chetti, 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 Rozhdestven-skiy, Henryk Michalewski, Ian Tenney, Ivan Gr-ishchenko, Jacob Austin, James Keeling, Jane La-banowski, Jean-Baptiste Lepiau, 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 Reid, Maciej Mikula, Ma-teo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Os-car 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 Klien-chenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu hui Chen, Zafarali Ahmed, Zhitaogong, Tris Warkentin, Ludovic Peran, Minh Gi-ang, Clément Farabet, Oriol Vinyals, Jeff Dean, Ko-ray Kavukcuoglu, Demis Hassabis, Zoubin Ghahra-mani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Sen-ter, Alek Andreev, and Kathleen Kenealy. 2024a. Gemma: Open Models Based on Gemini Research and Technology. *Preprint*, arXiv:2403.08295.
- Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhu-patiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Pe-ter Liu, Pouya Tafti, Abe Friesen, Michelle Cas-bon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Pi-otr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Wal-ton, Aliaksei Severyn, Alicia Parrish, Aliya Ah-mad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, An-thony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Wein-berger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii El-tyshchev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen,

- Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidson, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshtir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Cogan, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. 2024b. Gemma 2: Improving Open Language Models at a Practical Size. *Preprint*, arXiv:2408.00118.
- Valentin Hartmann, Anshuman Suri, Vincent Bind-schadler, David Evans, Shruti Tople, and Robert West. 2023. Sok: Memorization in general-purpose large language models. *arXiv preprint arXiv:2310.18362*.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Léo 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, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Léo 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. *Preprint*, arXiv:2401.04088.
- Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google’s multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.
- Dan Klein and Christopher Manning. 2004. Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 478–485, Barcelona, Spain.
- Jianling Li, Meishan Zhang, Peiming Guo, Min Zhang, and Yue Zhang. 2023. LLM-enhanced Self-training for Cross-domain Constituency Parsing. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8174–8185, Singapore. Association for Computational Linguistics.
- Jun Li, Yifan Cao, Jiong Cai, Yong Jiang, and Kewei Tu. 2020. An empirical comparison of unsupervised constituency parsing methods. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3278–3283, Online. Association for Computational Linguistics.
- Junlong Li, Jinyuan Wang, Zhuosheng Zhang, and Hai Zhao. 2024. Self-Prompting Large Language Models for Zero-Shot Open-Domain QA. 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 296–310, Mexico City, Mexico. Association for Computational Linguistics.
- Boda Lin, Xinyi Zhou, Binghao Tang, Xiaocheng Gong, and Si Li. 2023. ChatGPT is a Potential Zero-Shot Dependency Parser. *Preprint*, arXiv:2310.16654.
- Haitao Liu, Chunshan Xu, and Junying Liang. 2017. Dependency distance: A new perspective on syntactic patterns in natural languages. *Physics of life reviews*, 21:171–193.
- Ailsa Meechan-Maddon and Joakim Nivre. 2019. How to parse low-resource languages: Cross-lingual parsing, target language annotation, or both? In *Proceedings of the Fifth International Conference on Dependency Linguistics (Depling, SyntaxFest 2019)*, pages 112–120, Paris, France. Association for Computational Linguistics.
- Albert Nijenhuis and Herbert Wilf. 1978. *Combinatorial Algorithms for Computers and Calculators*, second edition. Academic Press.
- OpenAI. 2024. GPT-4 Technical Report. *Preprint*, arXiv:2303.08774.
- Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael

- Bendersky. 2024. Large language models are effective text rankers with pairwise ranking prompting. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1504–1518, Mexico City, Mexico. Association for Computational Linguistics.
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners.
- Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10776–10787, Singapore. Association for Computational Linguistics.
- Y. Shiloach. 1979. A minimum linear arrangement algorithm for undirected trees. *SIAM J. Comput.*, 8(1):15–32.
- Anders Søgaard. 2020. Some languages seem easier to parse because their treebanks leak. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2765–2770, Online. Association for Computational Linguistics.
- Yuanhe Tian, Fei Xia, and Yan Song. 2024. Large Language Models Are No Longer Shallow Parsers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7131–7142, Bangkok, Thailand. Association for Computational Linguistics.
- 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.
- Ke Tran and Arianna Bisazza. 2019. Zero-shot dependency parsing with pre-trained multilingual sentence representations. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 281–288, Hong Kong, China. Association for Computational Linguistics.
- Jiaan Wang, Yunlong Liang, Fandong Meng, Beiqi Zou, Zhixu Li, Jianfeng Qu, and Jie Zhou. 2023. Zero-shot cross-lingual summarization via large language models. In *Proceedings of the 4th New Frontiers in Summarization Workshop*, pages 12–23, Singapore. Association for Computational Linguistics.
- Weizhi Wang, Zhirui Zhang, Yichao Du, Boxing Chen, Jun Xie, and Weihua Luo. 2021. Rethinking zero-shot neural machine translation: From a perspective of latent variables. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4321–4327, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- 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. *arXiv preprint arXiv:2203.11171*.
- 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.
- Haoran Xu and Philipp Koehn. 2021. Zero-shot cross-lingual dependency parsing through contextual embedding transformation. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 204–213, Kyiv, Ukraine. Association for Computational Linguistics.
- Daniel Zeman, Joakim Nivre, Mitchell Abrams, Elia Ackermann, Noëmi Aepli, Hamid Aghaei, Željko Agić, Amir Ahmadi, Lars Ahrenberg, Chika Kennedy Ajede, et al. 2024. Universal dependencies 2.14. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.
- Xuan Zhang, Navid Rajabi, Kevin Duh, and Philipp Koehn. 2023. Machine translation with large language models: Prompting, few-shot learning, and fine-tuning with QLoRA. In *Proceedings of the Eighth Conference on Machine Translation*, pages 468–481, Singapore. Association for Computational Linguistics.
- Shengyao Zhuang, Bing Liu, Bevan Koopman, and Guido Zuccon. 2023. Open-source large language models are strong zero-shot query likelihood models for document ranking. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8807–8817, Singapore. Association for Computational Linguistics.

A Post-processing

Figure 3 breaks down the process of obtaining a valid dependency tree (fully connected, no cycles and only one root) from the raw output of the LLMs. In the second post-processing step, to enforce the unique-root constraint we randomly selected a root from the subset of outgoing arcs from node 0, or just a random node if there are no such arcs. Out-of-range arcs were resolved by replacing the head with the root node. To break cycles and connect all components, our post-processing algorithm performs a breadth-first search from the root node, removing those arcs that create cycles and connecting disconnected nodes to the root node.

Prompt example	
In dependency parsing the CoNLL format for the sentence <The trial begins again Nov 28 .> is:	
1 The	2 det
2 trial	3 nsubj
3 begins	0 root
4 again	3 advmod
5 Nov.	3 obl:tmod
6 28	5 nummod
7 .	3 punct
Now return the CoNLL format for the sentence: <What if Google Morphed Into GoogleOS ?>	
Raw output	
Sure! This is the CoNLL format for the sentence <What if Google Morphed Into GoogleOS ?>	
1 What	0 nsubj
2 if	4 mark
3 Google	4 nsubj
4 Morphed	0 root
5 into	6 case
6 GoogleOS	8 nmod
7 ?	4 punct
Let me know if (...)	
(1) Well-formatted output	
1 What	0 nsubj
2 if	4 mark
3 Google	4 nsubj
4 Morphed	0 root
5 into	6 case
6 GoogleOS	8 nmod
7 ?	4 punct
(2) Valid dependency tree	
1 What	4 nsubj
2 if	4 mark
3 Google	4 nsubj
4 Morphed	0 root
5 into	6 case
6 GoogleOS	4 nmod
7 ?	4 punct

Figure 3: Dependency parsing prompt and the resulting tree after the second post-processing step. Figure 1 showed the original tree.

B Additional results

Table 3 shows the reference to each model used in our experimental study. All of them are publicly available in HuggingFace. Tables 4 to 7 show the performance of each approach aggregating the prediction by its part-of-speech tag and Table 8 breaks down the ratio of post-processing

Abbrev.	Repository
v1-2b	google/gemma-2b
v1-7b	google/gemma-7b
v2-9b	google/gemma-2-9b
v2-27b	google/gemma-2-27b
v2-7b	meta-llama/Llama-2-7b-chat-hf
v2-13b	meta-llama/Llama-2-13b-chat-hf
v2-70b	meta-llama/Llama-2-70b-chat-hf
v3-8b	meta-llama/Meta-Llama-3-8B-Instruct
v3-70b	meta-llama/Meta-Llama-3-70B-Instruct
v3.1-8b	meta-llama/Llama-3.1-8B-Instruct
v3.1-70b	meta-llama/Llama-3.1-70B-Instruct
v3.2-1b	meta-llama/Llama-3.2-1B-Instruct
v3.2-3b	meta-llama/Llama-3.2-3B-Instruct
v1-7b	mistralai/Mistral-7B-Instruct-v0.1
v2-7b	mistralai/Mistral-7B-Instruct-v0.2
v3-7b	mistralai/Mistral-7B-Instruct-v0.3
x1-7b	mistralai/Mistral-8x7B-Instruct-v0.1
x1-22b	mistralai/Mistral-8x22B-Instruct-v0.1
nemo	mistralai/Mistral-Nemo-Instruct-2407
large	mistralai/Mistral-Large-Instruct-2407

Table 3: HuggingFace reference to the instruction-based models used in our experiments.

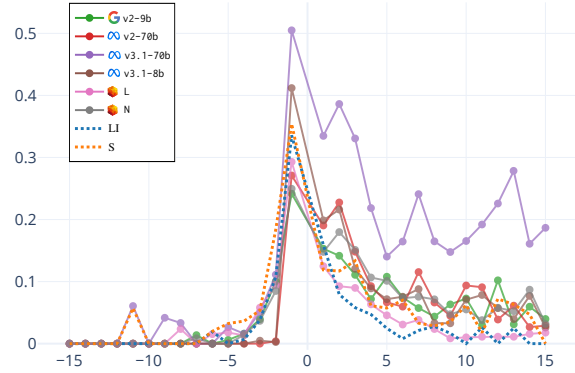


Figure 4: F-score across displacements in the French_{GSD} test set.

steps performed in each experiment. Figures 4 to 6 display the performance on the French_{GSD}, German_{GSD} and Hind_{HDBT} treebanks with respect to dependency displacement (signed dependency distance), following the definition of Anderson and Gómez-Rodríguez (2022), i.e., dependent index minus head index.

C Official language support

Table 9 shows which of our four target languages are supported by each of the models we used, according to the official documentation provided with each model.




		ADJ	ADP	ADV	AUX	CCONJ	DET	INTJ	NOUN	NUM	PART	PRON	PROPN	PUNCT	SCONJ	SYM	VERB	X
	A	9.45	6.05	7.74	6.35	5.16	6.96	25.00	8.56	10.70	6.32	6.43	13.29	9.27	3.65	16.51	6.33	5.13
	L	6.26	5.90	14.80	1.94	0.14	0.58	24.17	6.16	22.14	11.56	13.47	23.11	21.12	1.04	14.68	8.41	53.85
	R	51.06	36.28	43.62	44.72	46.68	55.67	30.00	16.00	26.57	73.50	37.34	22.29	15.28	16.67	25.69	2.19	2.56
	RD	8.56	6.29	8.84	6.35	5.70	6.06	18.33	8.53	9.41	6.63	7.50	12.37	10.50	3.39	21.10	6.79	20.51
	RD*	8.95	6.00	8.42	6.74	5.56	5.80	21.67	7.88	11.62	7.70	6.43	13.00	10.66	4.95	11.93	6.18	10.26
	LI	23.94	15.88	25.00	22.62	14.79	22.83	30.83	12.11	22.14	29.74	24.85	20.65	15.12	11.72	11.93	7.68	25.64
	LI*	25.11	19.03	25.26	23.59	19.13	26.20	25.83	11.34	19.56	28.97	27.12	17.62	14.41	9.11	14.68	3.72	15.38
	S	24.94	23.70	20.49	24.11	22.25	24.78	35.83	14.74	17.71	27.43	21.52	21.52	18.38	11.72	18.35	9.75	17.95
	v1-2b	9.45	7.23	8.16	9.92	6.24	8.43	15.83	7.64	11.07	10.94	9.02	12.18	9.04	5.47	6.42	8.18	10.26
	v1-7b	10.40	6.59	9.35	10.95	7.19	9.17	14.17	7.83	8.12	10.48	11.61	9.82	9.63	7.81	12.84	8.45	5.13
	v2-9b	15.10	7.62	11.14	10.24	8.01	10.86	16.67	10.95	7.38	8.94	9.90	14.16	6.75	4.69	9.17	19.65	2.56
	v2-7b	8.22	4.57	10.54	12.70	3.80	5.69	10.00	8.34	4.06	9.71	8.75	9.39	6.94	3.91	8.26	14.55	2.56
	v2-13b	10.46	1.92	6.46	5.57	1.49	2.11	15.00	10.80	2.40	5.39	3.93	14.68	6.23	2.34	6.42	27.06	0.00
	v2-70b	10.07	7.42	14.88	14.32	4.34	7.64	10.00	12.79	4.80	14.33	12.36	10.21	10.14	5.73	4.59	13.90	7.69
	v3-8b	9.56	5.31	11.56	12.70	4.21	6.54	16.67	9.86	4.43	11.40	9.81	10.83	10.59	3.65	9.17	13.28	10.26
	v3-70b	36.13	13.82	38.69	16.59	8.68	38.22	42.50	30.41	11.62	36.52	33.83	27.15	24.64	8.33	15.60	30.67	17.95
	v3.1-8b	16.00	16.08	25.17	12.96	23.20	19.93	27.50	15.93	9.78	20.65	21.29	24.03	17.44	7.29	19.27	14.40	30.77
	v3.1-70b	42.28	20.26	43.45	23.46	21.98	40.48	45.83	29.68	11.44	51.00	37.90	28.50	28.97	11.46	16.51	30.63	12.82
	v3.2-1b	7.94	5.56	11.56	12.96	5.16	6.48	13.33	8.22	5.17	10.32	9.53	9.87	9.27	5.47	11.01	9.71	10.26
	v3.2-3b	8.22	3.83	9.10	9.53	3.26	4.53	20.83	9.02	8.12	7.86	7.91	12.61	11.18	3.39	13.76	9.52	10.26
	v1-7b	10.85	4.77	8.50	9.40	4.48	5.06	20.00	9.79	4.61	8.17	8.65	11.94	6.17	2.34	5.50	16.47	5.13
	v2-7b	21.09	16.91	16.33	18.02	16.69	24.30	22.50	13.85	8.67	19.72	17.95	17.38	10.53	11.98	13.76	11.09	5.13
	v3-7b	30.76	24.68	27.81	27.28	24.42	30.89	15.00	13.85	8.67	37.60	21.84	17.00	13.15	11.72	19.27	9.79	12.82
	x1-7b	13.40	10.75	13.53	12.57	12.25	12.68	15.22	9.15	10.70	15.57	14.64	11.87	8.39	4.50	11.96	10.53	12.90
	x1-22b	27.57	18.78	31.21	22.88	18.72	29.20	30.83	18.01	13.10	31.12	24.62	23.06	18.86	14.58	18.35	16.51	28.21
	nemo	13.42	7.62	10.29	9.07	7.33	9.80	20.00	11.31	6.27	8.17	9.16	15.50	8.59	3.65	9.17	19.23	20.51
	large	21.48	5.31	15.99	9.27	6.78	10.12	25.83	18.27	10.52	10.48	11.01	22.82	11.95	2.60	19.27	38.85	15.38

Table 4: UAS aggregated by universal part-of-speech tag in the English_{EWT} test set.




		ADJ	ADP	ADV	AUX	CCONJ	DET	INTJ	NOUN	NUM	PRON	PROPN	PUNCT	SCONJ	SYM	VERB	X
	A	4.27	3.65	3.70	5.01	3.61	3.85	22.22	4.50	3.98	4.83	2.65	4.47	1.56	0.00	4.14	0.00
	L	51.23	1.08	23.82	0.84	0.00	0.81	22.22	5.57	19.03	4.47	25.92	11.80	10.94	5.13	9.62	42.86
	R	19.38	36.49	36.55	55.43	29.72	89.79	0.00	1.61	44.25	39.00	2.24	15.09	3.91	23.08	0.00	3.57
	RD	4.60	4.53	4.93	4.18	5.22	3.92	0.00	4.39	3.98	5.01	5.71	4.97	1.56	2.56	4.38	3.57
	RD*	3.28	4.46	5.54	2.51	2.41	3.85	11.11	4.18	3.10	4.11	4.29	4.47	3.91	0.00	5.36	3.57
	LI	28.41	15.74	23.41	21.73	11.65	30.22	0.00	5.47	20.35	20.21	13.67	10.46	5.47	15.38	5.24	17.86
	LI*	27.09	18.38	22.18	23.40	14.46	35.63	0.00	4.98	25.22	21.11	14.90	11.21	6.25	15.38	4.26	28.57
	S	13.79	24.73	21.15	26.46	18.88	34.28	0.00	9.00	21.24	21.11	9.39	12.56	7.81	15.38	7.43	14.29
	v1-2b	10.67	6.82	11.50	10.58	6.43	12.24	0.00	5.79	12.39	8.59	9.39	6.75	3.91	10.26	9.01	10.71
	v1-7b	10.84	9.05	11.91	9.47	5.62	13.52	0.00	4.93	9.29	9.84	7.35	5.99	3.12	5.13	7.92	14.29
	v2-9b	16.42	9.86	11.29	11.42	7.23	19.27	11.11	9.38	3.10	11.45	8.57	6.32	2.34	7.69	17.78	3.57
	v2-7b	10.67	16.15	17.25	26.46	13.25	40.70	22.22	4.72	19.91	22.00	5.10	8.77	2.34	17.95	2.56	0.00
	v2-13b	9.85	11.82	17.45	14.21	8.84	27.86	0.00	5.31	11.50	15.56	5.10	8.35	3.12	10.26	5.72	3.57
	v2-70b	30.71	8.18	21.77	6.69	3.21	24.95	11.11	17.42	6.64	15.38	21.84	8.26	6.25	12.82	19.37	21.43
	v3-8b	13.79	12.30	16.43	17.55	6.83	28.06	11.11	4.02	20.35	17.17	9.18	9.02	1.56	10.26	7.67	3.57
	v3-70b	41.05	19.73	28.34	8.64	6.43	47.73	11.11	26.74	9.73	22.00	34.08	15.01	5.47	5.13	21.92	28.57
	v3.1-8b	32.02	17.30	28.95	25.63	22.49	53.75	11.11	11.36	13.72	25.04	11.02	13.58	3.12	12.82	14.86	7.14
	v3.1-70b	53.20	27.91	37.78	13.93	14.46	59.77	0.00	27.49	15.49	28.09	26.12	23.44	5.47	15.38	29.23	21.43
	v3.2-1b	7.22	5.41	6.16	3.90	3.61	9.87	11.11	5.47	8.85	8.77	4.69	4.81	3.91	12.82	10.96	0.00
	v3.2-3b	10.84	5.81	8.21	9.75	2.41	12.24	11.11	7.18	6.64	7.51	10.00	7.00	3.91	5.13	9.01	14.29
	v1-7b	11.82	6.69	10.06	4.46	4.42	12.17	0.00	6.22	6.19	8.94	9.80	5.90	2.34	7.69	14.98	14.29
	v2-7b	21.35	19.05	17.45	14.48	12.05	33.60	0.00	11.74	7.96	19.32	11.43	10.12	3.12	20.51	12.55	21.43
	v3-7b	26.44	29.39	31.62	31.20	21.29	70.05	11.11	10.40	16.37	29.34	13.27	12.14	3.12	17.95	11.21	14.29
	x1-7b	16.09	6.42	9.03	6.69	4.02	15.89	22.22	10.93	6.64	7.69	7.76	6.16	5.47	5.13	27.65	0.00
	x1-22b	29.23	20.61	26.28	14.76	17.27	38.54	22.22	14.47	18.58	21.29	17.76	13.24	7.03	23.08	14.98	7.14
	nemo	17.24	10.34	14.17	8.64	6.43	20.76	11.11	11.95	4.42	11.63	13.88	8.94	3.91	5.13	19.12	10.71
	large	12.97	13.38	14.99	16.43	11.24	25.42	0.00	5.63	16.37	16.28	8.37	8.60	5.47	7.69	6.21	7.14

Table 5: UAS aggregated by universal part-of-speech tag in the French_{GSD} test set.

		ADJ	ADP	ADV	AUX	CCONJ	DET	INTJ	NOUN	NUM	PART	PRON	PROPN	PUNCT	SCONJ	SYM	VERB	X
A L R RD RD* LI LI* S	A	5.36	4.61	4.36	5.51	3.46	5.26	0.00	4.85	6.01	5.24	3.83	6.26	4.52	4.35	0.00	4.07	16.00
	L	1.75	1.18	12.31	23.77	3.68	0.80	25.00	2.28	9.87	10.48	19.72	30.04	23.13	0.00	25.00	1.58	28.00
	R	69.43	29.95	34.35	6.23	43.29	68.97	0.00	15.49	63.52	47.62	17.16	8.51	10.49	1.86	50.00	2.64	24.00
	RD	4.77	5.73	6.85	5.80	6.49	4.95	25.00	6.33	4.72	4.76	5.25	6.36	6.22	5.59	0.00	6.18	16.00
	RD*	6.33	5.04	5.92	6.23	5.84	5.31	0.00	6.62	4.29	5.71	5.39	7.44	5.88	8.07	0.00	5.35	12.00
	LI	26.39	15.13	18.07	16.23	16.88	28.69	25.00	9.58	34.33	11.43	15.60	18.10	17.25	1.24	0.00	5.05	24.00
	LI*	27.85	15.75	22.12	17.97	16.67	31.83	0.00	8.39	31.76	20.00	20.71	18.59	14.08	1.86	0.00	3.77	20.00
G	S	29.11	18.56	18.69	12.03	16.45	26.57	0.00	12.18	24.46	25.71	15.60	10.67	11.63	3.73	25.00	7.69	20.00
	v1-2b	17.33	10.46	13.71	6.23	9.96	20.78	0.00	7.68	17.60	10.00	11.91	9.39	9.43	4.97	0.00	6.93	8.00
	v1-7b	26.29	11.71	14.95	3.33	16.02	27.94	0.00	8.07	11.16	16.67	9.50	7.63	6.22	0.00	50.00	20.12	12.00
∞	v2-9b	18.31	10.40	12.85	5.65	8.66	18.52	0.00	7.39	8.15	13.33	8.09	10.67	5.96	3.11	25.00	22.76	16.00
	v2-7b	32.23	16.56	19.39	6.38	21.21	35.15	0.00	10.03	25.75	17.62	12.77	9.00	9.22	1.24	50.00	4.90	4.00
	v2-13b	34.54	18.22	23.13	5.58	18.94	36.85	0.00	10.17	28.10	18.10	12.19	9.81	9.32	0.00	0.00	4.35	15.38
	v2-70b	42.16	18.24	28.12	11.74	21.43	46.82	0.00	14.01	12.88	24.29	18.44	18.49	13.83	1.24	0.00	13.26	28.00
	v3-8b	58.13	25.34	29.36	6.96	30.74	58.49	0.00	13.47	23.18	29.52	18.72	11.94	13.95	1.24	50.00	17.41	28.00
	v3-70b	48.69	24.22	31.78	7.25	11.26	47.92	0.00	25.81	18.45	32.38	24.26	29.75	17.93	1.86	0.00	30.22	32.00
	v3.1-8b	45.86	20.55	28.35	9.28	27.92	47.44	0.00	13.79	17.17	30.00	21.70	20.35	15.43	1.86	25.00	13.79	28.00
	v3.1-70b	60.76	26.40	34.19	16.38	23.81	53.80	25.00	27.19	20.60	39.52	23.40	27.50	24.36	1.86	0.00	27.51	24.00
	v3.2-1b	10.42	6.54	9.66	6.38	5.41	14.32	0.00	6.88	10.30	7.14	12.34	8.02	8.54	2.48	25.00	10.85	12.00
	v3.2-3b	30.19	15.88	18.07	6.81	21.86	28.65	25.00	10.77	23.61	20.00	12.62	10.37	8.54	2.48	25.00	9.27	20.00
🔴	v1-7b	21.71	13.14	13.24	4.35	14.29	23.43	0.00	8.13	9.44	11.43	11.77	10.27	7.23	1.24	0.00	16.28	20.00
	v2-7b	33.79	18.12	21.81	4.64	19.48	35.06	0.00	11.06	15.45	18.57	12.77	13.21	9.30	1.24	0.00	14.32	12.00
	v3-7b	62.51	27.90	31.31	7.97	37.88	61.54	0.00	13.95	24.89	37.62	18.01	10.47	11.59	0.62	25.00	8.89	12.00
	x1-7b	11.11	0.00	30.00	8.33	0.00	30.43		13.51	0.00	0.00	22.22	0.00	18.75	0.00		0.00	0.00
	x2-22b	30.19	18.80	21.88	9.42	16.23	32.40	0.00	13.02	19.31	20.48	19.29	17.03	13.49	3.11	0.00	13.34	32.00
	nemo	11.88	6.54	10.20	5.36	6.93	13.04	0.00	9.03	4.29	9.52	6.81	11.06	6.43	0.62	25.00	27.51	20.00
	large	21.81	7.60	11.99	4.78	6.49	14.90	25.00	12.63	6.87	11.90	8.09	12.23	10.23	0.62	25.00	36.17	24.00

Table 6: UAS aggregated by universal part-of-speech tag in the German_{GSD} test set.

		ADJ	ADP	ADV	AUX	CCONJ	DET	NOUN	NUM	PART	PRON	PROPN	PUNCT	SCONJ	VERB	X
A L R RD RD* LI LI* S	A	4.56	4.88	4.93	4.77	5.83	5.77	4.99	2.89	6.06	5.39	4.44	5.41	2.29	4.48	11.11
	L	6.34	83.96	0.66	68.81	0.00	0.00	0.09	0.00	40.32	0.29	0.09	16.78	0.76	4.97	11.11
	R	73.13	0.07	22.37	0.13	27.09	83.09	25.65	75.32	28.06	32.87	31.70	13.55	0.00	5.03	55.56
	RD	5.40	4.58	3.29	5.73	3.78	4.70	4.99	4.76	5.47	5.90	4.15	4.96	3.82	4.97	0.00
	RD*	4.84	4.68	3.29	5.64	5.51	6.44	4.68	4.91	5.02	5.76	4.19	5.04	3.97	5.66	0.00
	LI	28.98	27.87	8.55	38.46	10.08	40.13	11.14	30.30	25.11	15.89	16.34	17.02	0.76	7.08	22.22
	LI*	30.20	28.24	9.87	32.94	12.44	37.85	12.12	27.99	23.93	16.69	15.93	13.22	0.92	7.71	33.33
G	S	23.91	23.91	11.84	27.97	10.71	24.83	13.80	19.19	19.79	15.52	13.36	16.20	4.89	11.70	11.11
	v1-2b	9.77	23.07	5.26	15.43	3.15	12.62	4.74	8.23	13.59	6.78	6.35	8.02	1.83	5.58	11.11
	v1-7b	16.16	8.24	6.25	4.85	6.77	21.34	6.26	11.69	8.12	8.82	8.54	4.71	0.00	15.75	0.00
∞	v2-9b	13.76	17.97	7.89	13.34	5.35	16.11	6.63	11.54	12.11	8.82	8.18	4.30	1.22	18.00	22.22
	v2-7b	14.51	12.82	5.92	11.08	5.51	21.34	5.86	12.41	10.49	9.55	9.08	7.81	1.37	5.14	0.00
	v2-13b	9.91	32.48	4.61	22.66	2.68	18.12	3.96	7.22	16.10	9.91	6.17	9.34	1.83	5.26	0.00
	v2-70b	10.77	28.76	7.32	20.89	4.60	22.68	3.77	10.00	19.72	9.14	8.22	10.45	0.00	3.94	
	v3-8b	20.81	28.20	13.16	23.91	5.51	28.99	6.97	12.70	24.52	11.88	8.29	6.86	0.15	16.32	11.11
	v3-70b	20.61	16.79	11.51	13.01	8.92	22.76	10.29	17.99	16.13	14.98	12.95	8.22	1.03	5.70	0.00
	v3.1-8b	25.27	30.57	12.50	32.86	7.87	38.79	9.83	22.37	20.09	17.42	11.69	7.02	0.00	13.55	11.11
	v3.1-70b	17.67	18.19	6.47	15.46	11.52	22.44	10.38	12.95	18.28	13.69	12.79	9.01	2.40	4.19	0.00
	v3.2-1b	8.08	6.87	2.96	6.19	5.98	11.14	4.23	6.49	5.17	8.75	5.36	6.40	1.07	7.25	0.00
	v3.2-3b	7.56	35.68	5.92	22.49	1.26	8.46	2.86	3.46	16.25	4.74	4.53	7.07	1.07	8.84	11.11
🔴	v1-7b	9.53	19.55	4.28	6.44	2.68	13.42	3.68	5.63	9.16	7.14	6.08	2.15	0.15	17.74	0.00
	v2-7b	15.88	22.49	11.51	9.49	4.88	22.01	7.14	10.53	17.58	8.45	8.29	3.88	0.46	18.64	0.00
	v3-7b	29.64	30.29	16.12	18.31	8.98	44.56	11.01	26.84	21.42	14.58	14.98	6.16	0.46	16.73	22.22
	x1-7b	29.64	30.29	16.12	18.31	8.98	44.56	11.01	26.98	21.42	14.58	14.96	6.16	0.46	16.73	22.22
	x1-22b	28.64	33.08	9.83	20.04	13.70	33.33	11.79	27.62	17.60	16.85	15.19	10.49	1.53	9.16	16.67
	nemo	9.58	11.84	4.61	7.90	3.31	10.07	4.99	7.36	11.37	4.96	5.16	5.12	0.92	20.69	11.11
	large	25.37	14.66	11.51	9.14	12.64	29.81	10.38	21.94	16.49	13.20	13.77	8.02	1.03	5.43	0.00

Table 7: UAS aggregated by universal part-of-speech tag in the Hindi_{HDTB} test set.




		en _{EW} T			fr _{GSD}			de _{GSD}			hi _{HDTB}		
		NP	P1	P2	NP	P1	P2	NP	P1	P2	NP	P1	P2
	v1-2b	13.38	16.18	70.44	7.45	14.90	77.64	7.06	12.90	80.04	6.06	16.27	77.67
	v1-7b	14.44	16.51	69.04	6.97	10.82	82.21	10.75	0.00	89.25	8.43	0.12	91.45
	v2-9b	15.17	0.05	84.79	4.81	0.00	95.19	5.32	0.00	94.68	3.27	0.00	96.73
	v2-7b	24.22	0.00	75.78	1.92	2.64	95.43	4.09	8.70	87.21	5.82	5.23	88.95
	v2-13b	14.59	0.00	85.41	22.12	10.58	67.31	32.89	5.12	61.98	18.71	13.18	68.11
	v2-70b	46.89	0.00	53.11	13.70	0.00	86.30	23.34	0.10	76.56	7.47	8.71	83.82
	v3-8b	49.01	0.00	50.99	4.81	6.25	88.94	22.42	0.00	77.58	0.48	0.00	99.52
	v3-70b	58.69	0.10	41.21	33.41	0.00	66.59	28.25	0.00	71.75	24.76	12.72	62.52
	v3.1-8b	34.38	0.00	65.62	5.29	0.48	94.23	12.90	0.00	87.10	2.43	0.00	97.57
	v3.1-70b	65.86	0.05	34.09	42.79	0.24	56.97	46.57	0.00	53.43	26.14	11.34	62.52
	v3.2-1b	5.63	14.06	80.31	2.16	8.41	89.42	1.84	9.72	88.43	1.37	8.19	90.44
	v3.2-3b	5.15	14.68	80.16	2.40	10.58	87.02	3.99	9.01	87.00	2.73	12.17	85.10
	v1-7b	16.85	0.00	83.15	5.29	0.00	94.71	5.94	0.00	94.06	0.12	0.00	99.88
	v2-7b	15.12	0.00	84.88	3.85	0.00	96.15	5.02	0.00	94.98	0.36	0.00	99.64
	v3-7b	28.17	0.05	71.79	11.06	0.00	88.94	19.55	0.00	80.45	0.71	0.00	99.29
	x1-7b	26.22	13.27	60.51	3.37	0.00	96.63	25.00	12.50	62.50	1.25	0.06	98.69
	x1-22b	41.98	29.22	28.79	32.69	26.92	40.38	37.46	26.41	36.13	30.88	33.02	36.10
	nemo	15.74	0.00	84.26	3.61	0.00	96.39	4.09	0.00	95.91	0.48	0.00	99.52
	large	18.25	0.00	81.75	22.84	15.14	62.02	7.88	0.00	92.12	18.26	11.20	70.54

Table 8: Distribution of the amount of post-processing steps performed in each zero-shot parser. **NP** represents the ratio of generated trees that did not require post-processing (only removing non-tabular lines), **P1** for those trees that only required the first post-processing step (e.g. removing extra columns) and **P2** for those trees that required of the full post-processing step (e.g. breaking cycles).




Model	English	French	German	Hindi
	v1-2b	✓		
	v1-7b	✓		
	v2-9b	✓		
	v2-7b	✓		
	v2-13b	✓		
	v2-70b	✓		
	v3-8b	✓	✓	✓
	v3-70b	✓	✓	✓
	v3.1-8b	✓	✓	✓
	v3.1-70b	✓	✓	✓
	v3.3-1b	✓	✓	✓
	v3.3-3b	✓	✓	✓
	v1-7b	✓	✓	
	v2-7b	✓		
	v3-7b	✓		
	x1-7b	✓	✓	
	nemo	✓	✓	✓
	large	✓	✓	✓

Table 9: Language support across different models. A tick symbol (✓) indicates that the model supports the respective language, while empty cells indicate lack of support.

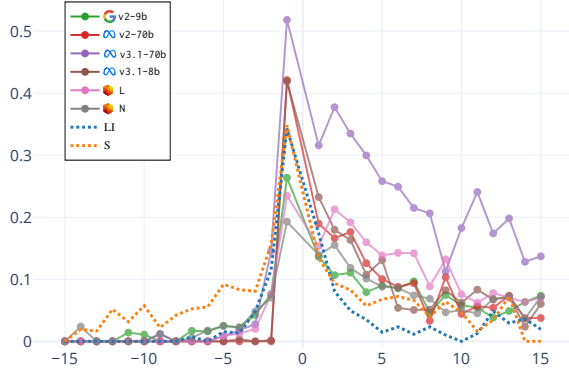


Figure 5: F-score across displacements in the German_{GSD} test set.

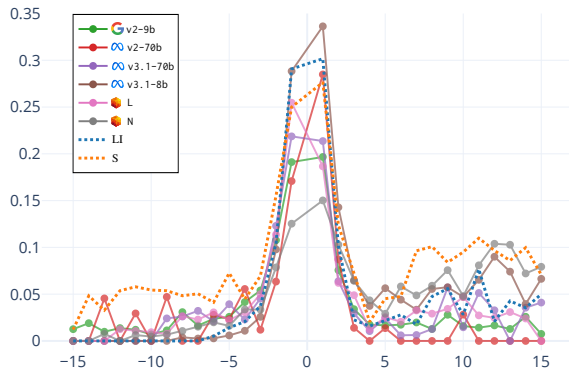


Figure 6: F-score across displacements in the Hindi_{HDTB} test set.

Optimizing Estonian TV Subtitles with Semi-supervised Learning and LLMs

Artem Fedorchenko

Tallinn University of Technology
artem.fedorchenko@taltech.ee

Tanel Alumäe

Tallinn University of Technology
tanel.alumae@taltech.ee

Abstract

This paper presents an approach for generating high-quality, same-language subtitles for Estonian TV content. We fine-tune the Whisper model on human-generated Estonian subtitles and enhance it with iterative pseudo-labeling and large language model (LLM) based post-editing. Our experiments demonstrate notable subtitle quality improvement through pseudo-labeling with an unlabeled dataset. We find that applying LLM-based editing at test time enhances subtitle accuracy, while its use during training does not yield further gains. This approach holds promise for creating subtitle quality close to human standard and could be extended to real-time applications.

1 Introduction

Same-language subtitles for video material, like TV talk shows, investigative pieces, and educational content serve as a valuable resource for the deaf and hard-of-hearing community, non-native speakers and native speakers alike. For instance, recent studies (Mykhalevych and Preply, 2024; Kim et al., 2023) have revealed that 50% of Americans and 85% of the Netflix users overall frequently watch TV and streaming video content with subtitles. Studies show that subtitles can enhance understanding and memory retention. A lot of viewers choose to enjoy their content quietly at home, keeping subtitles on to avoid disturbing their roommates or family.

Subtitles differ from verbatim (word-by-word) transcripts in many aspects. Subtitles represent typically a condensed version of the speech, designed to convey the essential meaning without capturing every word. They may omit filler words, repetitions, and non-verbal sounds, and may rewrite phrases, focusing on clarity and readability for viewers. Since subtitles are displayed on-screen during playback, they are formatted to fit within a limited time frame and limited line length, ensuring they are easy to read while the viewer is watching.

This paper outlines the development of an accurate offline same-language subtitle generation model for Estonian TV content. Using existing human-created subtitles, we fine-tune Whisper (Radford et al., 2022) and explore further improvements with semi-supervised learning and LLM-based post-editing techniques. Our findings demonstrate that Whisper can be trained to closely replicate human subtitling style, creating well-segmented and often rephrased subtitles. Additionally, we find that iterative pseudo-labeling of a large unlabeled dataset improves subtitle quality across all metrics. While a state-of-the-art commercial LLM (OpenAI *gpt-4o*¹) can enhance subtitle quality during test time, its use at training time to improve pseudo-labeled subtitles through post-editing is not effective.

2 Related Work

Both iterative pseudo-labeling and LLM-based post-editing have been an active area of research in the context of verbatim automatic speech recognition (ASR). Pseudo-labeling based semi-supervised learning in ASR has been studied since at least (Zavaliagos et al., 1998) and has been later investigated in several works, e.g. by Vesely et al. (2013); Xu et al. (2020).

To the best of our knowledge, Ma et al. (2023) was the first to show the potential of zero-shot and few-shot LLM-based ASR error correction. This approach has been later extended to take into account uncertainty estimation of ASR outputs (Pu et al., 2023) and retrieval-augmented generation for correcting speech recognition entity name errors (Pusateri et al., 2024).

Xi et al. (2024) showed that LLM-based error correction and data filtering can be also used for refining the pseudo-label transcripts during semi-supervised learning. This work is similar to ours,

¹We used a regular version of GPT-4o, which was accessed on October 14, 2024.

however, it is applied in the context of a code-switched Mandarin-English ASR task.

3 Method

Our method for developing an automated subtitle generation system involves several steps: training with supervised data, using iterative pseudo-labeling, and applying LLM-based error correction.

We start by training the Whisper large-v3 model (Radford et al., 2022) on a supervised dataset. This dataset consists of audio recordings paired with their subtitles.

Next, we use an unsupervised dataset to perform two iterations of pseudo-labeling. In this step, we generate pseudo-labels using the last trained model and combine them with the original supervised dataset, followed by training a new model on this data.

We also apply LLM-based post-editing of the generated subtitles, by instructing the LLM to fix the mistakes in the subtitles and giving it a segment of generated subtitle file. We experiment with applying this LLM-based post-editing in two distinct phases: at test time (i.e., to generated subtitles of the test data) and during training time (i.e., to generated subtitles of the unsupervised dataset).

4 Experiments

4.1 Datasets

As a supervised dataset², we used recordings and the corresponding subtitles from the Estonian national TV. The subtitles had been produced for the deaf and hard-of-hearing community by expert subtitlers. The supervised dataset consists of 993 audio-subtitle pairs, totaling 778 hours of audio, corresponding to 10 different TV show series (multi-party talk shows on various topics, political debates, infotainment programs). We randomly selected 17 recordings out of this set for testing.

The unsupervised dataset contains 7128 audio recordings, amounting to 3923 hours of audio. It contains similar material as the supervised dataset but also contains news program recordings, which the supervised dataset doesn't include.

4.2 Evaluation metrics

While evaluating ASR outputs using word error rate (WER) is relatively straightforward, find-

ing an appropriate metric for evaluating automatic subtitling systems is more complicated. Since subtitles often rephrase spoken content to enhance clarity and readability, WER may not accurately reflect the quality of the subtitles. WER does also not account for the formatting and timing of subtitles, which are crucial for viewer comprehension.

In our work, we use three metrics for comparing machine-generated subtitles against reference subtitles: subtitle edit rate (SubER) (Wilken et al., 2022) and two variations of BLEURT (Selam et al., 2020). SubER is based on a modified version of edit distance that incorporates shifts. This allows it to account for the specific properties of subtitles, such as timing and segmentation. However, SubER doesn't take into account that the same meaning can be conveyed with different words or phrases. Thus, we also use BLEURT for evaluation. BLEURT is a learned metric, trained on subjective human evaluations scores of machine translation references and the corresponding candidate sentences. BLEURT outputs scores that usually in the range of 0..1 (with 1 being a perfect match) and is found to be better correlated with human judgments in several languages than BLEU scores. We used the multilingual BLEURT-20-D12 model introduced by Pu et al. (2021). Furthermore, we use two variations of BLEURT: t-BLEURT and AS-BLEURT, which differ in the way generated subtitles are aligned to references. AS-BLEURT splits the reference subtitles into sentences, aligns generated subtitles to the references (Matusov et al., 2005) and then computes BLEURT score for each sentence, while t-BLEURT does the alignment based on the timing information in the subtitles (Cherry et al., 2021).

4.3 Baseline Model

As a baseline model, we finetuned Whisper on our supervised dataset using a cross-entropy objective. The model was trained for 4 epochs using the AdamW optimizer (Loshchilov and Hutter, 2019) with a learning rate of 1×10^{-5} . We used an effective batch size of 32 audio chunks and applied Stochastic Weight Averaging (SWA) (Izmailov et al., 2018) after the first epoch.

During decoding, we use the Silero VAD model (Silero Team, 2021) to remove non-speech parts.

4.4 Iterative Pseudo-Labeling

Next, to improve performance of the baseline model we used **iterative pseudo-labeling (IPL)**

²<https://cs.taltech.ee/staff/tanel.alumae/data/etv-subtitles/>

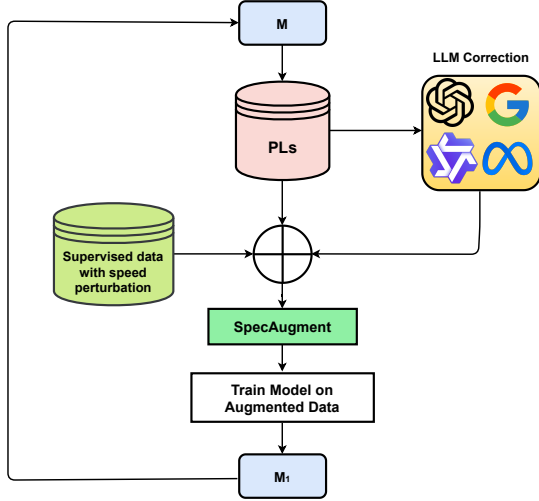


Figure 1: Pseudo-labels generated by model are either passed through LLM or used as is.

— a semi-supervised learning technique that enables model refinement on unlabeled data. Starting with an initial model trained on supervised data, we generate pseudo-labels for unlabeled samples and use these to retrain the model iteratively.

Our approach, which we illustrate in Figure 1, explores two strategies for refining pseudo-labels:

- **Direct Pseudo-Labeling:** Using pseudo-labels generated by the model itself.
- **LLM-Enhanced Pseudo-Labeling:** Refining pseudo-labels with a LLM to correct potential errors and ensure alignment with human subtitling standards.

In both approaches, we combine pseudo-labeled data with the original supervised dataset, modified by applying speed perturbation. To make the model more robust we applied SpecAugment (Park et al., 2019) on spectrogram level.

We did two iterations of training with pseudo-labels, the training setup was similar to the one with supervised data. Additionally, we incorporated weighted loss function:

$$\mathcal{L}_{\text{total}} = (1 - \lambda) \cdot \mathcal{L}_{\text{supervised}} + \lambda \cdot \mathcal{L}_{\text{pseudo-labels}}$$

where $\lambda = 0.35$ was chosen empirically using Optuna (Akiba et al., 2019).

System Instruction:

You are tasked with correcting Estonian subtitles in a subtitle file. **YOU MUST NOT** create, remove, or modify block numbers and timestamps. **ONLY** correct the text within the existing blocks.

Input:

```
1
00:00:00,000 --> 00:00:02,760
Tere õhtust kõigile, algamas
on vestlussaade kahekõne.
2
00:00:02,760 --> 00:00:07,340
Uued rahva poolt palavalt oodatud jõud
on toompeal justkui killustunud.
```

LLM Output:

```
1
00:00:00,000 --> 00:00:02,760
Tere õhtust kõigile, algamas
on vestlussaade "Kahekõne".
2
00:00:02,760 --> 00:00:07,340
Uued rahva poolt palavalt oodatud jõud
on Toompeal justkui killustunud.
```

Figure 2: Example of an LLM instruction used for refining Estonian subtitles. The model corrected the spelling of the TV show name "Kahekõne" and the historical place name "Toompea" in Estonia.

Table 1: Comparison of different LLMs for their performance in error correction.

LLM	SubER↓
-	35.1
GPT-4o	34.2
Llama 3.1 405B (FP8 quant.)	35.5
Qwen 2.5 72B	36.4
Gemma 2 27B	38.4

4.5 LLM-based post-editing

To ensure fast and efficient correction of subtitles using an LLM, we split the generated subtitles into chunks of 40 subtitle blocks. This approach allows for great parallelization without exceeding the maximum token limit per request. An example of the request format is shown in Figure 2.

In the development phase, we evaluated several different LLMs for their suitability for this task. Table 1 shows the SubER results on test data, after applying LLM-based error correction with different LLMs. We compared OpenAI GPT-4o and three of the best open source LLMs from different vendors. As can be seen, only GPT-4o was able to improve SubER-based subtitle accuracy. Based on

Table 2: Results of different models, with or without test-time LLM post-editing.

Finetuning data	Pseudo-label LLM-post-editing?	Test-time LLM-post-editing?	SubER↓	t-BLEURT↑	AS-BLEURT↑
-			59.8	.563	.728
Verbatim transcripts			51.5	.526	.770
Subtitles (A)			35.1	.545	.799
Subtitles (B)		✓	34.2	.582	.810
<i>Pseudo-labeling, iteration 1</i>					
Subtitles + pseudo-labels			34.5	.526	.808
Subtitles + pseudo-labels		✓	33.9	.529	.815
Subtitles + pseudo-labels	✓		34.4	.525	.810
Subtitles + pseudo-labels	✓	✓	33.9	.528	.816
<i>Pseudo-labeling, iteration 2</i>					
Subtitles + pseudo-labels			33.4	.529	.853
Subtitles + pseudo-labels (C)		✓	33.1	.598	.858
Subtitles + pseudo-labels	✓		33.6	.570	.854
Subtitles + pseudo-labels	✓	✓	33.3	.571	.856

these results, we used GPT-4o in our experiments.

During our experiments, we observed that LLMs often struggle to output the exact timestamps and block numbers correctly. To address this, we verified these details against the original subtitles to ensure accuracy and re-requested the LLM to fix the issue, if necessary. We also experimented with one-shot and few-shot prompts but did not observe any significant quality improvement, so we opted not to include them. Additionally, we set a threshold on the number of allowable reference check failures: if the model failed more than 3 times, we reverted to the original subtitle.

4.6 Results

Table 2 lists evaluation results of the native Whisper model (not fine-tuned on additional data), Whisper fine-tuned on 1066 hours of verbatim transcripts from the TalTech Estonian Speech Dataset 1.0 (Alumäe et al., 2023), and after fine-tuning with different sets of subtitle datasets. The table also highlights the effects of LLM-based post-editing applied during both the training and testing phases.

The results indicate that fine-tuning on subtitle data yields notably lower SubER values compared to fine-tuning on verbatim transcripts, demonstrating the different nature of subtitles and verbatim transcripts. However, the BLEURT scores for both the native Whisper model and the version fine-tuned on verbatim transcripts are surprisingly high. This outcome may be attributed to BLEURT’s design as a semantic similarity metric,

which effectively maps both verbatim transcripts and subtitle-like compressed transcripts to proximate points in its semantic space.

To support our interpretation of the achieved results, we computed Wilcoxon signed-rank test (Wilcoxon, 1945) between models **A**, **B** and **C** highlighted in the Table 2. P-value achieved from comparing model **A** to **B** is 0.000, **B** to **C** is 0.004 and **A** to **C** is 0.000. These p-values are all below common significance thresholds (e.g., 0.05), indicating that the differences between the models are statistically significant.

Given that, findings suggest that iterative semi-supervised learning enhances subtitle quality, as evidenced by improvements across all test metrics. LLM-based post-editing applied to decoded subtitles provides additional benefits in most cases. However, contrary to findings in (Xi et al., 2024), applying LLM-based post-editing to pseudo-labeled subtitles in the unsupervised dataset does not yield further improvements.

Although a formal human evaluation of the generated subtitles was not conducted, the authors’ subjective assessment suggests that minimal manual post-editing would be required to achieve error-free subtitles, particularly for in-domain TV data. A sample video from our test dataset, featuring both reference subtitles and subtitles generated by our best model³ is available at <https://www.youtube.com/watch?v=bEow5vGIgZc>. A smaller version of

³<https://huggingface.co/TalTechNLP/whisper-large-v3-et-sub>

this model based on Whisper *large-v3-turbo* can be freely used via a simple web application⁴.

5 Conclusion

In this work, we presented an approach to automated subtitle generation, leveraging the multilingual Whisper model, semi-supervised learning, and LLM-based post-editing. By utilizing supervised and unsupervised datasets, we demonstrated that iterative pseudo-labeling can indeed improve the quality of subtitles. Our results show that applying an LLM during test time has a more significant impact on the results across all the key metrics than during training time. Future work will focus on adapting our approach to real-time scenarios.

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 and Data Mining*.
- Tanel Alumäe, Joonas Kalda, Külliki Bode, and Martin Kaitsa. 2023. Automatic closed captioning for Estonian live broadcasts. In *Proc. NoDaLiDa*, Tórshavn, Faroe Islands.
- Colin Cherry, Naveen Arivazhagan, Dirk Padfield, and Maxim Krikun. 2021. Subtitle translation as markup translation. In *Proc. Interspeech*, pages 2237–2241.
- Pavel Izmailov, Dmitrii Podoprikin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. 2018. Averaging weights leads to wider optima and better generalization. In *Proceedings of the 34th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 876–885. AUAI Press.
- Hyunju Kim, Yan Tao, Chuanrui Liu, Yuzhuo Zhang, and Yuxin Li. 2023. Comparing the impact of professional and automatic closed captions on video-watching experience. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*.
- Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.
- Rao Ma, Mengjie Qian, Potsawee Manakul, Mark Gales, and Kate Knill. 2023. Can generative large language models perform asr error correction? *arXiv preprint arXiv:2307.04172*.
- Evgeny Matusov, Gregor Leusch, Oliver Bender, and Hermann Ney. 2005. Evaluating machine translation output with automatic sentence segmentation. In *Proc. IWSLT*.
- Nadiia Mykhalevych and Preply. 2024. Survey: Why America is obsessed with subtitles. Accessed: 2024-10-25.
- Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. SpecAugment: A simple data augmentation method for automatic speech recognition. In *Interspeech 2019*. ISCA.
- Amy Pu, Hyung Won Chung, Ankur P Parikh, Sebastian Gehrmann, and Thibault Sellam. 2021. Learning compact metrics for MT. In *Proc. EMNLP*.
- Jie Pu, Thai-Son Nguyen, and Sebastian Stüker. 2023. Multi-stage large language model correction for speech recognition. *ArXiv*, abs/2310.11532.
- Ernest Pusateri, Anmol Walia, Anirudh Kashi, Bortik Bandyopadhyay, Nadia Hyder, Sayantan Mahinder, R. Anantha, Daben Liu, and Sashank Gondala. 2024. Retrieval augmented correction of named entity speech recognition errors. *ArXiv*, abs/2409.06062.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust speech recognition via large-scale supervision. *arXiv preprint arXiv:2212.04356*.
- Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proc. ACL*.
- Silero Team. 2021. Silero vad: Pre-trained enterprise-grade voice activity detector. <https://github.com/snakers4/silero-vad>.
- Karel Veselý, Mirko Hannemann, and Lukáš Burget. 2013. Semi-supervised training of deep neural networks. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 267–272. IEEE.
- Frank Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83.
- Patrick Wilken, Panayota Georgakopoulou, and Evgeny Matusov. 2022. SubER: A metric for automatic evaluation of subtitle quality. *arXiv preprint arXiv:2205.05805*.
- Yu Xi, Wen Ding, Kai Yu, and Junjie Lai. 2024. Semi-supervised learning for code-switching ASR with large language model filter. *arXiv preprint arXiv:2407.04219*.
- Qiantong Xu, Tatiana Likhomanenko, Jacob Kahn, Awni Hannun, Gabriel Synnaeve, and Ronan Collobert. 2020. Iterative pseudo-labeling for speech recognition. *arXiv preprint arXiv:2005.09267*.

⁴<https://huggingface.co/spaces/TanelAlumae/whisper-large-v3-et-subs>

George Zavaliagkos, Man-Hung Siu, Thomas Colthurst, and Jayadev Billa. 1998. Using untranscribed training data to improve performance. In *Proc. ICSLP*, volume 1998.

Modeling Multilayered Complexity in Literary Texts

Pascale Feldkamp

CHC / Aarhus University
pascale.moreira@cc.au.dk

Márton Kardos

CHC / Aarhus University
martonkardos@cas.au.dk

Kristoffer L. Nielbo

CHC / Aarhus University
kln@cas.au.dk

Yuri Bizzoni

CHC / Aarhus University
yuri.bizzoni@cc.au.dk

Abstract

We explore the relationship between stylistic and sentimental complexity in literary texts, analyzing how they interact and affect overall complexity. Using a dataset of over 9,000 English novels (19th-20th century), we find that complexity at the stylistic/syntactic and sentiment levels tend to show a linear association. Finally, using dedicated datasets, we show that both stylistic/syntactic features – particularly those relating to information density – as well as sentiment features are related to text difficulty rank as well as average processing time.¹

1 Introduction

Literary texts exemplify language operating at its most refined and demanding: they are capable of generating an experience – often emotional or evocative (Bizzoni and Feldkamp, 2024) – through the sheer force of words (Starr, 2013; Girju and Lambert, 2021; Miall and Kuiken, 1994). In this domain, language’s capacity to evoke emotions, construct worlds, and create experiences is pushed to its limits. To do so, literary texts explore the boundaries of what human language can achieve in terms of expressiveness, depth, and evocative power. It manipulates form and meaning for its effects in a way that seems unmatched in other domains – exhibiting complexity at multiple levels, for example, matching an information-dense style with an unpredictable narrative.

Multidimensional complexity might also be the reason why traditional stylistic metrics for gauging the difficulty of a text – often developed for

nonfiction – such as readability formulae, do not adequately capture the level of complexity of literary texts (Dalvean and Enkhbayar, 2018a); and might be a factor in why literary texts are associated with longer human processing times than nonfiction (Zwaan, 1991; Brysbaert, 2019).

This complexity, however, might not manifest uniformly at all levels: a literary story may be emotionally complex while maintaining a simplified syntax. This is why the problem of modeling complexity at different linguistic levels in literary language presents a particularly intriguing challenge. Understanding how linguistic complexity affects reader experience and whether there are trade-offs between formal and emotional aspects is critical in unraveling the cognitive demands and rewards associated with literary reading.

While many recent studies have sought to gauge the effect of stylistic and syntactic features of complexity for forms of reader appreciation (Brottrager et al., 2022; Barré et al., 2023; Wu et al., 2024; Bizzoni et al., 2023b; Wang et al., 2019; Koolen et al., 2020), the sentiment and emotional dimension has been an overlooked aspect of literary complexity. Complexity at this level is difficult to define. While a metric like simple sentiment standard deviation can be used to gauge the width of the ‘sentiment palette’ that authors are using in a novel, some more sophisticated measures for the complexity of novels’ sentiment arcs – i.e., the trajectory of positive and negative valences across a story – have been developed in recent years, like the approximate entropy or the Hurst exponent of sentiment arcs (Bizzoni et al., 2021, 2022).

Very little work has explored the connection between these different levels of complexity: the relation between complexity at the stylistic level and complexity at the sentiment level. Moreover, little work has tested whether sentiment complexity behaves similar to stylistic and syntactic complexity in relation to reader experience. To address this

¹To ensure reproducibility, all code and raw data are available at: https://github.com/centre-for-humanities-computing/literary_complexity

gap, we pose two research questions. Firstly:

RQ1: *What is the relationship between complexity features at different textual levels (e.g., stylistic/syntactic, and sentiment levels)? We hypothesize two possible relationships between different levels of complexity:*

H1a: *There is a trade-off between complexity at different levels, where, e.g., increased stylistic and syntactic complexity leads to “simplification” at the sentiment level.*

H1b: *Complexity features at different levels co-occur, so that, e.g., higher stylistic and syntactic complexity is associated with greater sentiment complexity.*²

The first two hypotheses carry different consequences. The first hypothesis (H1a) draws from the concept of ‘cognitive compensation’ observed in other domains, which suggests that optimized communication requires distributing readers’ cognitive load across linguistic layers. For example, when lexical complexity increases, syntactic structures may simplify to balance cognitive demands (Degaetano-Ortlieb and Teich, 2022). In this scenario, complexity at one level could functionally balance complexity at another – for instance, syntactic complexity might work alongside sentimental simplicity. In contrast, H1b derives from the idea that aesthetic phenomena function as ‘supernormal stimuli’, intentionally amplifying complexity across levels to heighten engagement, eliciting amplified responses (Dubourg and Baumard, 2022; Costa and Corazza, 2006). This scenario also carries the interesting possibility that works with high stylistic and syntactic complexity also embrace challenging sentiment profiles. Heightened complexity at multiple levels would impose a higher cognitive load on readers, yet could foster a more compelling aesthetic experience.

Secondly, we seek to probe the relation of each feature level to actual reader experience:

RQ2: *What is the relationship between complexity features at different levels of a text and cognitive load experienced by readers?*

H2a: *Features at the sentiment level behave like stylistic and syntactic features in increasing readers’ cognitive load, impacting the reader’s ability to process the text.*

H2b: *Features at the sentiment level have an in-*

*verse behavior to stylistic and syntactic features, so more complexity at the sentiment level decreases readers’ cognitive load.*³

Through these questions, we aim to explore how complexity at different linguistic levels might enhance or compromise one another. In a first part of this study, we investigate the relationship between stylistic/syntactic and sentiment complexity (RQ1) in a large corpus of novels. In the second part, we assess whether sentiment complexity mirrors stylistic/syntactic complexity in its impact on readers’ cognitive load (RQ2), using dedicated datasets on reading time and novels’ difficulty rank.

2 Related Works

Computational literary analyses have long attempted to model textual complexity by analyzing both stylistic and syntactic features. As early as 1893, Sherman used sentence length to study textual complexity. The increasing prominence of Digital Humanities in recent decades has greatly expanded this field. Recent studies have focused on canonical literature (Barré et al., 2023; Brottrager et al., 2022; Wu et al., 2024; Algee-Hewitt et al., 2016), showing that such texts exhibit a higher level of complexity across various dimensions. For example, studies have demonstrated that canonical works tend to have denser nominal styles, lower readability levels, and less predictable sentiment arcs (Wu et al., 2024; Bizzoni et al., 2023b).

Much of the focus on stylistic and syntactic complexity can be traced to formalist literary theory, which emphasizes *stylistic discomfort* as a hallmark of the *literariness* of texts. This theory argues that literary texts slow down reading by creating linguistic unfamiliarity or “foregrounding” (Mukařovský, 1964; van Peer, 1986). While some work has found reader consensus on foregrounding phenomena (van Peer, 1986), no comprehensive taxonomy of such features exists. Still, such features have been implicitly assumed to be formal or stylistic. This aligns with a long-standing debate on formalism in literary analysis, where a superficial focus on form has been claimed to overshadow content (Eagleton, 1983). As an exception, the experimental study of Miall and Kuiken (1994) found that reading times increased with the

²The null hypothesis (1) would naturally be that these levels bear no relation to each other, i.e., are independent.

³The null hypothesis (2) would naturally be that sentiment features bear no relation to readers’ cognitive load.

frequency of foregrounding features, including affective features in their taxonomy.

Other, more theoretical studies have suggested that the extended processing time associated with literary texts (Zwaan, 1991) is linked to emotional and emphatic engagement (Scapin et al., 2023; László and Cupchik, 1995) and to increased reflection on non-literal meaning distinctive to literary reading (Hakemulder, 2020). In short, the complexity of literary texts may evoke more cognitively demanding affective processes than non-fiction, echoing the idea of literary texts as enhanced stimulus objects (Dubourg and Baumard, 2022). Moreover, recent psycholinguistic research has also emphasized how sentiment and emotional engagement affect readers’ cognitive load, showing that negative valence and emotional features can increase reading times and that readers respond rapidly to valence cues (Pfeiffer et al., 2020; Lei et al., 2023; Arfé et al., 2023). These studies suggest that sentiment plays a critical role in reader experience, yet few works have explored the intersection of stylistic, syntactic, and complexity at the sentiment level.

While sentiment analysis (SA) has become a popular method for gauging emotional content in texts (Rebora, 2023), its application in literary analysis remains conceptually and theoretically underdeveloped. Some recent work has applied complexity measures such as approximate entropy and the Hurst exponent to sentiment arcs, suggesting that these measures provide insight into the complexity of narratives at the level of feelings or emotions evoked (Bizzoni et al., 2021, 2022). Yet, the connection between complexity at the stylistic level and the complexity in sentiment trajectories measured by these metrics remains largely unexplored.

We seek to fill this gap by investigating the relationship between stylistic/syntactic complexity and complexity at the sentiment level in literary texts, contributing to the broader understanding of how complexity at different linguistic levels interacts to shape the complexity profile of literature and its readers’ cognitive experience.

3 Methods

3.1 Data

The Chicago Corpus

For our investigation on the relation between features, we use the *Chicago Corpus* of novels in

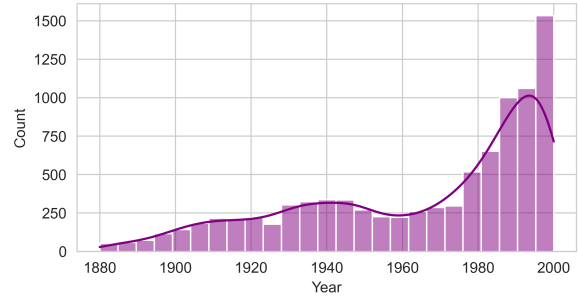


Figure 1: *Chicago Corpus*, temporal distribution of novels.

English ($n = 9,089$) from the period 1880-2000 (see the distribution of the corpus over time in Fig. 1). The novels in our corpus are predominantly by anglophone authors, selected based on the number of worldwide library holdings,⁴ favoring those with broader representation. Since library holdings capture both popular demand and prestigious, curated literature, the corpus spans a diverse range of genres – from Agatha Christie to James Joyce.⁵⁶

Beyond the *Chicago Corpus*, we use two dedicated datasets for part II of our study, where we gauge the relation between features at different levels with proxies of perceived complexity – i.e., reading time from the *Natural Stories corpus* and a list of the difficulty rank of novels (Dalvean and Enkhbayar, 2018a).

Natural Stories Corpus

The *Natural Stories corpus* consists of 10 English stories, each approximately 1,000 words long, totaling 485 sentences. These publicly available narratives, which includes tales by the Brothers Grimm, were revised to incorporate low-frequency and psycholinguistically interesting constructions while maintaining fluency. Self-paced reading (SPR) data was collected from 181 native English speakers, recording reaction times (RTs) for each word in a moving window setup. The dataset was filtered for control comprehension questions and outlier RTs ($< 100ms$ or $>$

⁴As indexed in worldcat.org

⁵See Bizzoni et al. (2024c) for details on the corpus. Recent studies of literary complexity have also used it, such as Wu et al. (2024).

⁶The feature dataset – though not full texts – is available at: https://github.com/centre-for-humanities-computing/chicago_corpus

3000ms).⁷ Note that our analysis operates at the story level, using average sentence RT, as we examine sentiment features based on broader contexts. Average sentence RT per story was calculated from the word RTs.

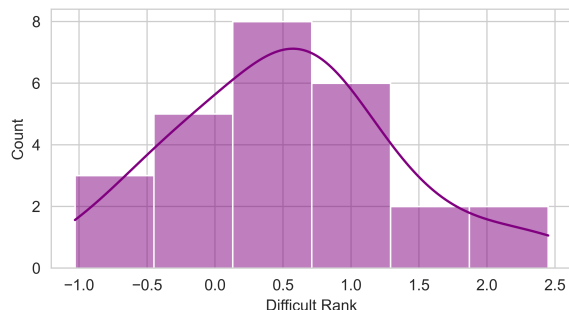


Figure 2: Distribution of Difficulty Rank across the 26 novels.

Difficulty rank of novels

With the aim of matching books to appropriate reader levels, Dalvean and Enkhbayar (2018a) curated a list of 200 novels, each assigned a difficulty rank. This rank is derived from a model trained on a binary prediction task (accuracy 89%) based on 48 linguistic and psycholinguistic features. We use these scores to estimate text complexity for the subset of books extant in the list and in the Chicago corpus, i.e., 26 novels (see Fig. 2). For the titles of the 26 novels, see Table 6 in Appendix B.

3.2 Features

The features utilized in this study have been used in previous works to distinguish textual profiles of different types of literature. The details on each measure can be found in Appendix D (Table 10). We focus on features that supposedly reflect stylistic or syntactic complexity, and have been widely used in recent computational literary studies. Features at the sentiment level were chosen to focus on overall variation and local and global complexity of the sentiment arc (Bizzoni et al., 2023b, 2022).

The sentiment dynamics central to our study are captured by both simple and complex measures. First, sentiment standard deviation (SD) represents the “palette” of sentiment in a novel, quantifying the overall variation in valence scores across

sentences to reflect sentiment range. Beyond this, two advanced measures – approximate entropy and the Hurst exponent – are applied to model more nuanced sentiment arcs linearly within a narrative.

Approximate entropy (*ApEn*) assesses the local complexity and unpredictability within sentiment flows, where lower values signal a repetitive, predictable structure, and higher values indicate intricate, less predictable patterns in the narrative (Mohseni et al., 2022). To capture global coherence, we estimate the Hurst exponent (*H*) with adaptive fractal analysis (AFA) instead of the more commonly used detrended fluctuation analysis (DFA), avoiding the boundary errors and segment discontinuities common to DFA (Hu et al., 2021; Gao et al., 2011). By accounting for non-linear trends, AFA enables a smooth global trend, with higher *H* values suggesting sustained narrative coherence and lower values indicating more abrupt sentiment shifts across scales (Hu et al., 2021; Bizzoni et al., 2023d).⁸

For all sentiment features, which are derived from valence scores, we first annotated all novels at the sentence level for sentiment valence (where 1 represents the positive and -1 the negative polarity) using the *Syuzhet* package (Jockers, 2015). This tool was developed explicitly for literary language, and has shown the best performance for English in the literary domain, also compared to transformer-based models (Bizzoni et al., 2023a). We then calculated the standard deviation, *ApEn*, and Hurst exponent of sentiment arcs for all 9,000 *Chicago Corpus* novels, as well as stories of the *Natural Stories* dataset – taking these features to represent the variance, as well as the local and global predictability – in other words, complexity – of novels’ sentiment profile.

In the following first part of this study, we juxtapose stylistic/syntactic and these sentiment features of complexity across all novels, gauging the correlation between them. We then assess the link between stylistic/syntactic and sentiment levels by trying to predict individual sentiment variables using all the stylistic/syntactic features. This is done on the whole set of over 9,000 novels, making it the largest-scale experiment in this study, as well as the most comprehensive diachronically (end of 19th – 20th century).

⁷The *Natural Stories* data is available at: <https://github.com/languageMIT/naturalstories>

⁸See, recently, Bizzoni et al. (2024b) for the details on the computation of these sentiment measures.

SD sent	1	0.26	0.64	0.8	0.54	-0.69	0.6	-0.09	0.58	0.28	0.48	0.73	0.69	0.26	0.04	0.36	0.43	-0.19	0.14	0.08
Hurst	0.26	1	0.19	0.11	0.2	-0.13	-0.08	-0.27	0.18	0.02	0.11	0.15	0.06	-0.17	-0.35	0.08	0.09	0.06	0.26	0.08
ApEn	0.64	0.19	1	0.56	0.32	-0.35	0.27	-0.12	0.31	0.14	0.21	0.4	0.24	0.11	-0.06	0.44	0.08	-0.01	0.15	0.1
	SD sent	Hurst	ApEn	Sentence length	Wordlength	R Flesch ease	R Dale chall	Function words	Freq "of"	Freq "that"	Nominal verb ratio	NDD mean	NDD SD	TTR verb	TTR noun	MSTTR-100	Perplexity	Compressibility	Bigram entropy	Word entropy

Figure 3: The correlation (Spearman’s ρ) between stylistic/syntactic features and sentiment features. See table 10 in Appendix D for details on the computation of these features and for the label explanations.

3.3 Reading time & Difficulty rank

Features such as readability formulae are established indicators of textual complexity, but sentiment-based features are less studied and their impact on reading time remains unclear. Therefore we relate these features to perceived complexity, taking both reading time and text difficulty rank as proxies of perceived complexity associated with increased cognitive load for the reader.

To assess the relationship between the analyzed features and reader processing time, we first evaluate how well these features correlate with reaction times (RTs) from the *Natural Stories* corpus. This initial step provides indicators of how these features may influence cognitive processing and perceived text complexity.

As a second check, we address the absence of RTs for the novels in the Chicago Corpus by using a scoring list of 200 novels (Dalvean and Enkhbayar, 2018a).⁹ This list assigns a difficulty rank to 26 Chicago Corpus novels, which serves as a proxy for perceived difficulty. By predicting difficulty rank with our feature sets, we aim to further assess the role of sentiment features in the perceived difficulty of literary texts.

4 Results & Discussion

4.1 Part I: Relations between stylistic/syntactic & sentiment features

In part I of this study, we examined feature relations in the novels. We observe a strong correlation between sentiment-level features and a subset of stylistic/syntactic features, as shown in Fig. 3. Notably, readability formulas, word and sentence length, dependency length, lexical richness (‘MSTTR’), indicators of heavy nominal style (e.g., frequency of “of” and nominal

verb ratio), and LLM perplexity – all features commonly associated with harder-to-process and information-rich text – show a particularly strong correlation with sentiment standard deviation. Approximate entropy also displays a similar pattern of correlation with these features, while it appears less correlated with LLM-based perplexity. Additionally, the Hurst exponent, which captures global uncertainty, shows a relationship with these complexity metrics – not least do the sentiment features exhibit correlations internally ($.19 < \rho > .64$).

Most correlations across sentiment features align in the same direction; for instance, lower Flesch Ease readability (indicating lesser readability) correlates with higher sentiment arc entropy (*ApEn*) ($\rho = -.35$), higher sentiment standard deviation ($\rho = -.69$), and a tendentially higher Hurst exponent ($\rho = -.13$). For a more comprehensive view of correlation co-directionality, see the visualizations in Appendix A, Fig. 7.

Note that all sentiment features show a correlation with sentence length, which may partly explain their relationship with sentence-length-dependent metrics, such as readability indices (R Flesch Ease and R Dale-Chall). However, sentiment features are also clearly related to features that bear no relation to sentence length, such as the frequency of the use of “of”, indicating a more nominal (viz. information dense) writing style (Wu et al., 2024), or average and SD of the dependency length.

Given these strong correlations, we employed a linear regression model to determine whether stylistic/syntactic complexity features could predict sentiment-level complexity, particularly sentiment standard deviation. Results show that textual complexity features are indeed predictive of sentiment complexity (Table 1), with sentiment standard deviation exhibiting the strongest predic-

⁹The list is available in Appendix 2 of Dalvean and Enkhbayar (2018b), and in the repository of our paper.

Feature	F-stat	R^2	adj. R^2
Sentiment SD	1803.0	0.787	0.786
ApEn	364.2	0.427	0.426
Hurst	123.1	0.201	0.2

Table 1: Linear regression of sentiment features **based on stylistic/syntactic** features. Here for all, $p < 0.01$.

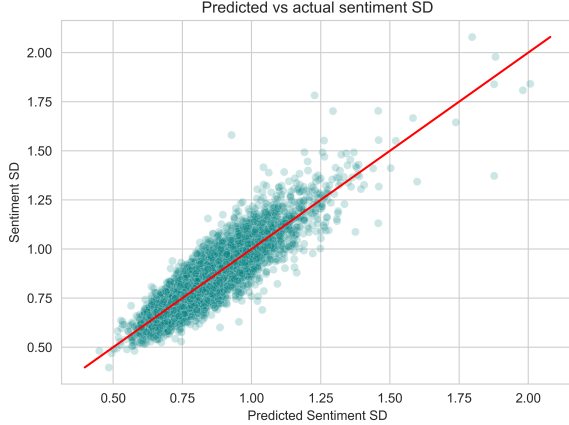


Figure 4: Linear fit between the predicted and actual **sentiment SD** based on the stylistic/syntactic complexity features.

tive relationship (Fig. 4). Interestingly, this relationship is bidirectional: sentiment features also demonstrate predictive power for stylistic and syntactic complexity features, with sentence length, readability formulae, dependency length (avg. & SD) and features like the frequency of “of”, indicating nominal style, displaying the strongest predictive relationships. See a few selected features in Table 2, and a full table in Appendix A, Table 5. This finding underscores a tightly coupled relationship between stylistic/syntactic complexity and sentimental variability, reinforcing hypothesis H1b: higher stylistic and syntactic complexity is associated with increased complexity at the sentiment level. This suggests that stylistic and affective dimensions in literary texts are interdependent, potentially amplifying each other’s complexity in ways that may shape readers’ engagement.

4.2 Part II: Relation of features to proxies of perceived complexity

In part II of this study, to examine the relationship between features and perceived complexity, we conducted two experiments. The first used RTs (reading times) from the *Natural Stories* corpus,

Feature	F-stat	R^2	adj. R^2
Flesch Ease Readability	2717.0	0.481	0.481
Dependency Length	4166.0	0.587	0.587
Nominal Ratio	1117.0	0.276	0.275

Table 2: Linear regression **based on sentiment features** to predict a stylistic/syntactic feature. Here, all $p < 0.01$.

compared to the same features as before,¹⁰ computed across the dataset’s ten stories. The second experiment involved analyzing the difficulty rank of 26 novels from the *Chicago Corpus*. In both cases, we aimed to predict reading time and difficulty rank by exploring correlations between the features and these variables. We employed linear regression based on stylistic/syntactic and sentiment feature sets, using each set separately and then jointly.

Given the relatively small sample sizes in both experiments (10 and 26 data points, respectively), we aimed to strengthen our findings by reducing collinearity in the feature set. To achieve this, we first applied PCA to the entire *Chicago Corpus* to capture the covariance structure and scaling of variables in a larger, more representative dataset. We then applied this PCA model to reduce dimensionality in our smaller dataset, minimizing the risk of overfitting to limited data. Details on this sanity check using PCA for collinearity reduction are presented in Appendix C: for difficulty rank in table 8 and for reading times in table 9.

4.2.1 Reading time

In relating features to reading times, we find that only some stylistic/syntactic and sentiment features exhibit linear correlations with reading time of the stories. These include lexical richness (‘MSTTR’), word entropy, and nominal ratio.

This scarcity of correlation might be due to insufficient datapoints. In a setting with augmented datapoints, the mentioned features remain significantly correlated, while we also see the p-value of sentiment SD and compressibility rising above the significance threshold (.05). For the augmented data setting, see Appendix B, Fig. 9. We show the correlation of the original data for lexical richness, nominal ratio and sentiment SD in Fig. 5.

¹⁰We excluded perplexity, as we could not ensure that publicly available stories were excluded from model training data. For the *Chicago Corpus*, perplexity derives from a self-trained model controlling for overlap (Wu et al., 2024).

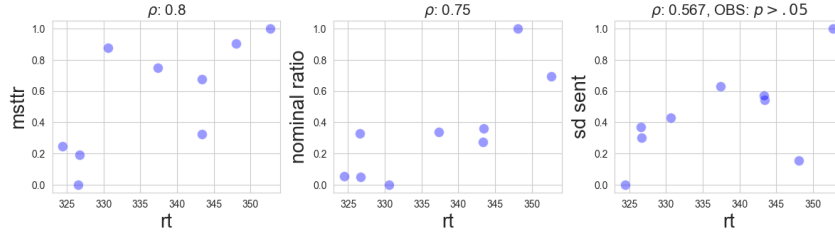


Figure 5: Correlation of selected features with RT, with Spearman’s ρ at the top of plots. Note that for sentiment SD, $p > .05$.

Moreover, correlations between features and RTs tend to be nonlinear, as some features, like readability formulae seem to show clustering both in the original and augmented data setting (see Appendix B, Figs. 8 and 9), but no *linear* correlation. Fig. 5 shows the correlation of RT and selected features. Note that while the correlation has $p > .05$, a tendential association of sent SD and RT can be observed. A larger corpus of annotated fiction is required to robustly confirm this tendency.

Features	F-stat	R^2	adj. R^2	p-val
Styl/Synt	15.38	0.902	0.844	< 0.01
Sentiment	2.01	0.547	0.275	0.231
All	28.76	0.945	0.912	< 0.01

Styl/Synt	<i>Bigram entropy, Nominal ratio, TTR Noun</i>			
Sentiment	<i>All sentiment features used</i>			
All	<i>Nominal ratio, Frequency “of”, Sent SD</i>			

Table 3: Linear regression **predicting RTs** of the *Natural Stories* using two feature sets, the three sentiment features, the three selected stylistic/syntactic features, and three selected features among all features. Below, the selected features in each category using RFE.

As the sample was too scarce, linear regression could not be carried out using the full feature set. Instead, we used Recursive Feature Elimination (RFE) to determine 3 features in the stylistic/syntactic category, and 3 out of all features.¹¹ We thus stay at the number of features corresponding to our number of sentiment features. Results of using linear regression to predict RT are shown in table 3. Notably, RFE leads to selecting sentiment SD as one of the overall top 3 significant features. Considering the scarce data, we consider this a means of comparing feature categories

¹¹RFE was performed using sklearn: https://scikit-learn.org/dev/modules/generated/sklearn.feature_selection.RFE.html

rather than an accurate model, i.e., for predicting RTs on unseen samples.

4.2.2 Difficulty rank

Using the 26 books in Chicago that had an assigned score in the difficulty ranking list, we sought to use different feature categories to predict the score of the novel. Results are shown in table 4. Note that visualizations of the predicted/actual values in Fig. 6 reflect an apparent improvement in our models’ predictive power when adding sentiment features to it. As in the reading time experiment, we do not claim any predictive power of this model but observe the effect of adding sentiment features for gauging difficulty rank.

Features	F-stat	R^2	adj. R^2	p-val
Styl/Synt	3.234	0.873	0.603	0.048
Sentiment	2.469	0.252	0.150	0.089
All	3.413	0.932	0.659	0.089

Table 4: Linear regression predicting **difficulty rank** using two feature sets, and all features.

Note that the p-value tends to be high when using all features, probably due to the limited amount of datapoints (table 4). However, predicted and actual difficulty rank in the sentiment-based model still exhibit a relation (Fig. 7(b)) and the model seems to improve when sentiment features are added (Fig. 7(c)). As in the previous experiment with RT, we also selected features with RFE (see Appendix B, table 7). Here, the features: frequency “of”, nominal ratio, word entropy, and perplexity appeared to be the most important, without sentiment features showing up among the 3 selected features.

5 Conclusion

Our results pertaining to our first question (RQ1) support H1a. Rather than a balance between dif-

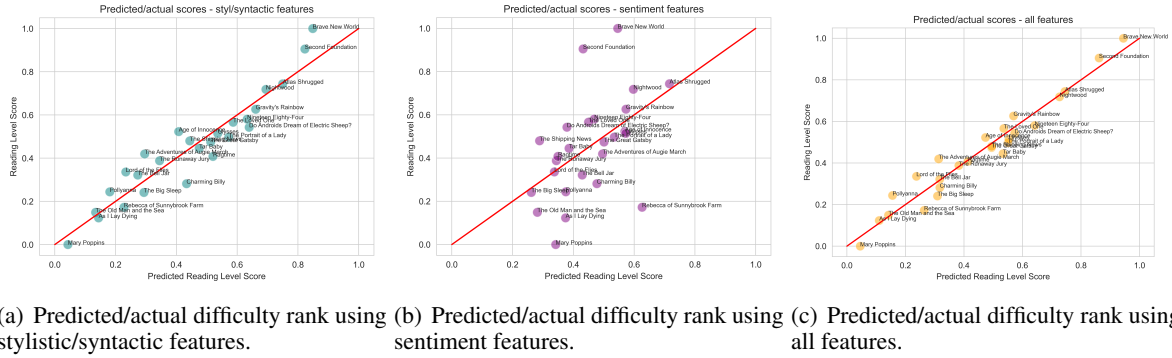


Figure 6: Comparison of the predicted vs actual difficulty rankings using different feature sets.

ferent aspects of language, we find that, at least over the whole *Chicago Corpus*, complexity at the stylistic and syntactic level tends to correspond to complexity at the sentimental level.

Regarding our second question (RQ2), our findings support H2a: it seems that both the stylistic/syntactic and the sentimental complexity impact the cognitive load of the readers, not only when measuring whole novels but even within much shorter stories. We should thus assume that the novels that push both levels to higher complexity are indeed asking more from the readers, and are providing a more challenging experience. It’s not obvious that features like overall variance in sentiment (sentiment SD) and the local and global linear dynamics of the sentiment arc (*ApEn*, Hurst) would relate to perceived complexity, making this finding particularly intriguing.

The question then remains as to why these two levels of complexity are tendentially intensified together, rather than showing a trade-off. In other words, why do works that offer a wider sentimental palette or a less predictable story arc also have, in general, a higher noun-to-verb ratio and a wider vocabulary?

Such literature – complex on multiple levels – may offer higher-quality reading experiences by amplifying both emotional and stylistic profiles. In this way, our findings suggest that literature may function as a “supernatural stimulus”, where every element is intensified simultaneously – a phenomenon that, while possibly engaging in fiction, would be counterproductive in nonfiction or more didactic texts, where clarity and ease are often prioritized. This distinction potentially sets literary texts apart from other domains, though future studies should more rigorously test differences between literary and nonliterary texts with

regard to multidimensional complexity.

The possibility of a trade-off between these dimensions of language is not off the table: it might occur within specific groups of texts with varying degrees of difficulty; and it is also possible that specific works of literature strike a balance differently, depending on their intended audience and the author’s specific style. But in general, our findings suggest that rather than being independent dimensions, style and “content” – taking sentiment here as a semantic element – might have a strong relation in literary texts. The style of the texts might have to align with its semantics, at least at the sentimental level. In this sense, the relevance of linguistic traits associated with the “nominal style” is particularly intriguing. Degaetano-Ortlieb and Teich (2022) has shown that this style is developed and applied in scientific and technical language to convey semantic information more efficiently while requiring a higher degree of concentration and preparation from the reader. This “optimal” strategy of linguistic communication might not be limited to technical prose but be exploited, despite their completely different aims, by literary works as well. In other words, it is possible that some aspects of complexity at the stylistic level are necessary for most works of art to convey the complexity of the sentimental level in a manner that is most effective to the creation of a powerful reading experience.

In the future, we intend to explore the relationship between these levels of complexity in literary language further, better formalizing the relation and role of each of the selected components. We would also examine the relationship between the perceived complexity or difficulty of a text and these features in an experiment setting.

References

- Mark Algee-Hewitt, Sarah Allison, Marissa Gemma, Ryan Heuser, Franco Moretti, and Hannah Walser. 2016. *Canon/Archive. Large-scale Dynamics in the Literary Field*. Stanford Literary Lab.
- Barbara Arfé, Pablo Delatorre, and Lucia Mason. 2023. Effects of negative emotional valence on readers' text processing and memory for text: an eye-tracking study. *Reading and Writing*, 36(7):1743–1768.
- Jean Barré, Jean-Baptiste Camps, and Thierry Poibeau. 2023. Operationalizing Canonicity: A Quantitative Study of French 19th and 20th Century Literature. *Journal of Cultural Analytics*, 8(3).
- Jonah Berger, Yoon Duk Kim, and Robert Meyer. 2021. What Makes Content Engaging? How Emotional Dynamics Shape Success. *Journal of Consumer Research*, 48(2):235–250.
- Yuri Bizzoni and Pascale Feldkamp. 2024. Below the sea (with the sharks): Probing textual features of implicit sentiment in a literary case-study. In *Proceedings of the Third Workshop on Understanding Implicit and Underspecified Language*, pages 54–61, Malta. Association for Computational Linguistics.
- Yuri Bizzoni, Pascale Feldkamp, Ida Marie Lassen, Mia Jacobsen, Mads Rosendahl Thomsen, and Kristoffer Nielbo. 2024a. Good Books are Complex Matters: Gauging Complexity Profiles Across Diverse Categories of Perceived Literary Quality. ArXiv:2404.04022 [cs].
- Yuri Bizzoni, Pascale Feldkamp, and Kristoffer Laigaard Nielbo. 2024b. Global Coherence, Local Uncertainty: 2024 Computational Humanities Research Conference, CHR 2024. In *Proceedings of the Computational Humanities Research Conference 2024*, Aarhus Denmark. CEUR Workshop Proceedings. Publisher: CEUR-WS.org.
- Yuri Bizzoni, Pascale Feldkamp Moreira, Emily Öhman, and Kristoffer L. Nielbo. 2023a. Comparing Transformer and Dictionary-based Sentiment Models for Literary Texts: Hemingway as a Case-study. In *NLP4DH (forthcoming)*, Tokyo, Japan.
- Yuri Bizzoni, Pascale Moreira, Nicole Dwenger, Ida Lassen, Mads Thomsen, and Kristoffer Nielbo. 2023b. Good reads and easy novels: Readability and literary quality in a corpus of US-published fiction. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 42–51, Tórshavn, Faroe Islands. University of Tartu Library.
- Yuri Bizzoni, Pascale Moreira, Mads Rosendahl Thomsen, and Kristoffer Nielbo. 2023c. Sentimental matters - predicting literary quality by sentiment analysis and stylometric features. In *Proceedings of the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, pages 11–18, Toronto, Canada. Association for Computational Linguistics.
- Yuri Bizzoni, Pascale Feldkamp Moreira, Ida Marie S. Lassen, Mads Rosendahl Thomsen, and Kristoffer Nielbo. 2024c. A matter of perspective: Building a multi-perspective annotated dataset for the study of literary quality. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 789–800, Torino, Italia. ELRA and ICCL.
- Yuri Bizzoni, Pascale Feldkamp Moreira, Mads Rosendahl Thomsen, and Kristoffer L. Nielbo. 2023d. The fractality of sentiment arcs for literary quality assessment: the case of nobel laureates. *Journal of Data Mining & Digital Humanities*, NLP4DH.
- Yuri Bizzoni, Telma Peura, Kristoffer Nielbo, and Mads Thomsen. 2022. Fractality of sentiment arcs for literary quality assessment: The case of nobel laureates. In *Proceedings of the 2nd International Workshop on Natural Language Processing for Digital Humanities*, pages 31–41, Taipei, Taiwan. Association for Computational Linguistics.
- Yuri Bizzoni, Telma Peura, Mads Rosendahl Thomsen, and Kristoffer Nielbo. 2021. Sentiment dynamics of success: Fractal scaling of story arcs predicts reader preferences. In *Proceedings of the Workshop on Natural Language Processing for Digital Humanities*, pages 1–6, NIT Silchar, India. NLP Association of India (NLP AI).
- Lloyd R. Bostian. 1983. How active, passive and nominal styles affect readability of science writing. *Journalism quarterly*, 60(4):635–670.
- Judith Brottrager, Annina Stahl, Arda Arslan, Ulrik Brandes, and Thomas Weitin. 2022. Modeling and predicting literary reception. *Journal of Computational Literary Studies*, 1(1):1–27.
- Marc Brysbaert. 2019. How many words do we read per minute? A review and meta-analysis of reading rate. *Journal of Memory and Language*, 109. Place: Netherlands Publisher: Elsevier Science.
- David H. Charney and Jack R. Rayman. 1989. The Role of Writing Quality in Effective Student Résumés. *Journal of Business and Technical Communication*, 3(1):36–53. Publisher: SAGE Publications Inc.
- Marco Costa and Leonardo Corazza. 2006. Aesthetic Phenomena as Supernormal Stimuli: The Case of Eye, Lip, and Lower-Face Size and Roundness in Artistic Portraits. *Perception*, 35(2):229–246. Publisher: SAGE Publications Ltd STM.
- Andreas van Cranenburgh and Rens Bod. 2017. A data-oriented model of literary language. In *Proceedings of the 15th Conference of the European*

- Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, pages 1228–1238, Valencia, Spain. Association for Computational Linguistics.
- Scott A. Crossley, Rod Roscoe, and Danielle S. McNamara. 2014. What Is Successful Writing? An Investigation Into the Multiple Ways Writers Can Write Successful Essays. *Written Communication*, 31(2):184–214. Publisher: SAGE Publications Inc.
- Michael Dalvean and Galbadrakh Enkhbayar. 2018a. Assessing the readability of fiction: A corpus analysis and readability ranking of 200 English fiction texts. *Linguistic Research*, 35:137–170.
- Michael Dalvean and Galbadrakh Enkhbayar. 2018b. A New Fiction Text Complexity Metric for Ranking Fiction Texts. pages 1–29.
- Stefania Degaetano-Ortlieb and Elke Teich. 2022. Toward an optimal code for communication: The case of scientific english. *Corpus Linguistics and Linguistic Theory*, 18(1):175–207.
- Edgar Dubourg and Nicolas Baumard. 2022. Why and How Did Narrative Fictions Evolve? Fictions as Entertainment Technologies. *Frontiers in Psychology*, 13. Publisher: Frontiers.
- Terry Eagleton. 1983. *Literary Theory: An Introduction*, later printing edition edition. Univ of Minnesota Pr.
- Katharina Ehret and Benedikt Szmezcany. 2016. An information-theoretic approach to assess linguistic complexity. In *Complexity, Isolation, and Variation*, pages 71–94. De Gruyter.
- Gerardo Febres and Klaus Jaffe. 2017. Quantifying literature quality using complexity criteria. *Journal of Quantitative Linguistics*, 24(1):16–53. ArXiv:1401.7077 [cs].
- Richard S. Forsyth. 2000. Pops and flops: Some properties of famous english poems. *Empirical Studies of the Arts*, 18(1):49–67.
- Jianbo Gao, Jing Hu, and Wen-wen Tung. 2011. Facilitating joint chaos and fractal analysis of biosignals through nonlinear adaptive filtering. *PLoS ONE*, 6(9).
- Craig L. Garthwaite. 2014. Demand spillovers, combative advertising, and celebrity endorsements. *American Economic Journal: Applied Economics*, 6(2):76–104.
- Roxana Girju and Charlotte Lambert. 2021. InterSense: An Investigation of Sensory Blending in Fiction. ArXiv:2110.09710 [cs].
- Frank Hakemulder. 2020. Finding Meaning Through Literature. *Anglistik*, 31(1):91–110. Publisher: Universitätsverlag WINTER GmbH Heidelberg.
- Qiyue Hu, Bin Liu, Mads Rosendahl Thomsen, Jianbo Gao, and Kristoffer L Nielbo. 2020. Dynamic evolution of sentiments in Never Let Me Go: Insights from multifractal theory and its implications for literary analysis. *Digital Scholarship in the Humanities*, 36(2):322–332.
- Qiyue Hu, Bin Liu, Mads Rosendahl Thomsen, Jianbo Gao, and Kristoffer L Nielbo. 2021. Dynamic evolution of sentiments in *Never Let Me Go*: Insights from multifractal theory and its implications for literary analysis. *Digital Scholarship in the Humanities*, 36(2):322–332.
- Arthur M. Jacobs and Annette Kinder. 2022. Computational analyses of the topics, sentiments, literariness, creativity and beauty of texts in a large Corpus of English Literature. ArXiv:2201.04356 [cs].
- Matthew L Jockers. 2015. Syuzhet: Extract sentiment and plot arcs from text. *Matthew L Jockers blog*.
- Justine Kao and Dan Jurafsky. 2012. A Computational Analysis of Style, Affect, and Imagery in Contemporary Poetry. In *Proceedings of the NAACL-HLT 2012 Workshop on Computational Linguistics for Literature*, pages 8–17, Montréal, Canada. Association for Computational Linguistics.
- Corina Koolen, Karina van Dalen-Oskam, Andreas van Cranenburgh, and Erica Nagelhout. 2020. Literary quality in the eye of the Dutch reader: The national reader survey. *Poetics*, 79:1–13.
- Anqi Lei, Roel M. Willems, and Lynn S. Eekhof. 2023. Emotions, fast and slow: processing of emotion words is affected by individual differences in need for affect and narrative absorption. *Cognition and Emotion*, 37(5):997–1005. Publisher: Routledge eprint: <https://doi.org/10.1080/02699931.2023.2216445>.
- Lei Lei and Matthew L. Jockers. 2020. Normalized Dependency Distance: Proposing a New Measure. *Journal of Quantitative Linguistics*. Publisher: Routledge.
- J. László and Gerald Cupchik. 1995. The role of affective processes in reading time and time experience during literary reception. *Empirical Studies of the Arts*, 13:25–37.
- Suraj Maharjan, John Arevalo, Manuel Montes, Fabio A. González, and Tamar Solorio. 2017. A multi-task approach to predict likability of books. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1217–1227, Valencia, Spain. Association for Computational Linguistics.
- Claude Martin. 1996. Production, content, and uses of bestselling books in quebec. *Canadian Journal of Communication*, 21(4).

- Carey McIntosh. 1975. Quantities of qualities: Nominal style and the novel. *Studies in Eighteenth-Century Culture*, 4(1):139–153.
- David S. Miall and Don Kuiken. 1994. Foregrounding, defamiliarization, and affect: Response to literary stories. *Poetics*, 22(5):389–407.
- Mahdi Mohseni, Volker Gast, and Christoph Redies. 2021. Fractality and variability in canonical and non-canonical english fiction and in non-fictional texts. 12.
- Mahdi Mohseni, Christoph Redies, and Volker Gast. 2022. Approximate entropy in canonical and non-canonical fiction. *Entropy*, 24(2):278.
- Jan Mukařovský. 1964. Standard language and Poetic Language. In Paul L. Garvin, editor, *A Prague School Reader on Esthetics Literary Structure, and Style*, pages 17–30. 1932. Georgetown University Press.
- Willie van Peer. 1986. *Stylistics and Psychology*. Croom Helm.
- Christian Pfeiffer, Nora Hollenstein, Ce Zhang, and Nicolas Langer. 2020. Neural dynamics of sentiment processing during naturalistic sentence reading. *NeuroImage*, 218:116934.
- Simone Rebora. 2023. Sentiment analysis in literary studies. A critical survey. *Digital Humanities Quarterly*, 17(2).
- Giulia Scapin, Cristina Loi, Frank Hakemulder, Katalin Bálint, and Elly Konijn. 2023. The role of processing foregrounding in empathic reactions in literary reading. *Discourse Processes*, 60(4-5):273–293. Publisher: Routledge .eprint: <https://doi.org/10.1080/0163853X.2023.2198813>.
- Emily Sheetz. 2018. Evaluating Text Generated by Probabilistic Language Models.
- Lucius A. Sherman. 1893. *Analytics of Literature: A Manual for the Objective Study of English Prose and Poetry*. Athenaeum Press. Ginn.
- Sanja Stajner, Richard Evans, Constantin Orasan, and Ruslan Mitkov. 2012. What can readability measures really tell us about text complexity? In *Proceedings of Workshop on natural language processing for improving textual accessibility*, pages 14–22, Istanbul, Turkey. Association for Computational Linguistics.
- G. Gabrielle Starr. 2013. *Feeling Beauty: The Neuroscience of Aesthetic Experience*. The MIT Press.
- Joan Torruella and Ramon Capsada. 2013. Lexical statistics and tipological structures: A measure of lexical richness. *Procedia - Social and Behavioral Sciences*, 95:447–454.
- Xindi Wang, Burcu Yucesoy, Onur Varol, Tina Eliassi-Rad, and Albert-László Barabási. 2019. Success in books: Predicting book sales before publication. *EPJ Data Science*, 8(1):31.
- Yaru Wu, Yuri Bizzoni, Pascale Moreira, and Kristoffer Nielbo. 2024. Perplexing canon: A study on GPT-based perplexity of canonical and non-canonical literary works. In *Proceedings of the 8th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2024)*, pages 172–184, St. Julians, Malta. Association for Computational Linguistics.
- Claire M. Zedelius, Caitlin Mills, and Jonathan W. Schooler. 2019. Beyond subjective judgments: Predicting evaluations of creative writing from computational linguistic features. *Behavior Research Methods*, 51(2):879–894.
- Rolf A. Zwaan. 1991. Some parameters of literary and news comprehension: Effects of discourse-type perspective on reading rate and surface structure representation. *Poetics*, 20(2):139–156.

A Relation between features

We attach the visualization of some correlations of stylistic/syntactic features with all three sentiment features (Fig. 7).

Additionally, the results of the extended linear regression are presented in table 5, where we sought to predict each stylistic/syntactic feature individually by sentiment features.

Styl/synt. feature	F-stat	R^2	adj. R^2
Sentence length	6862.0	0.7	0.7
Dependency SD	4295.0	0.594	0.594
Dependency Length	4166.0	0.587	0.587
Flesch Ease Readab.	2717.0	0.481	0.481
Dale-Chall Readab.	2655.0	0.475	0.475
“Of” Frequency	1651.0	0.36	0.36
Word length	1326.0	0.311	0.311
Nominal Verb Ratio	0.612	0.276	0.275
MSTTR	754.5	0.204	0.204
TTR Noun	494.0	0.144	0.144
TTR Verb	442.2	0.131	0.131
“That” Frequency	249.0	0.078	0.078
Bigram Entropy	225.9	0.071	0.071
Compressibility	166.0	0.054	0.053
Perplexity	147.3	0.048	0.047
Function words	146.0	0.047	0.047
Word Entropy	35.65	0.012	0.012

Table 5: Linear regression **based on sentiment features** to predict a stylistic/syntactic feature. The table is ordered by decreasing R^2 . Here for all, $p < 0.01$.

B Reading time & difficulty rank

Here we present the full results of our analysis on the relationship between features and both reading times (RTs) and difficulty rank.

For the **reading time (RT)** experiment, additional correlation coefficients, including stylistic and syntactic feature levels, with RTs from the Natural Stories corpus are provided and visualized in Fig. 8. To increase data points, we further split the stories with a 90% overlap between segments, effectively duplicating the data points. This approach retains as much of the global structure of the stories as possible – a crucial factor for features like the Hurst exponent, which is sensitive to structural changes. A visualization of these correlations is shown in Fig. 9.

For relating features to **difficulty rank (DR)**, we took the overlap of titles between the list of novels in Dalvean and Enkhbayar (2018a) and the *Chicago Corpus*. These are listed in table 6.

Author	Title	DR
Aldous Huxley	<i>Brave New World</i>	2.45
Isaac Asimov	<i>Second Foundation</i>	2.12
Ayn Rand	<i>Atlas Shrugged</i>	1.56
Djuna Barnes	<i>Nightwood</i>	1.47
Thomas Pynchon	<i>Gravity’s Rainbow</i>	1.15
George Orwell	<i>Nineteen Eighty-Four</i>	0.99
Evelyn Waugh	<i>The Loved One</i>	0.94
Philip K. Dick	<i>Do Androids Dream of Electric Sheep?</i>	0.86
Edith Wharton	<i>The Age of Innocence</i>	0.79
James Joyce	<i>Ulysses</i>	0.76
Henry James	<i>The Portrait of a Lady</i>	0.70
Annie Proulx	<i>The Shipping News</i>	0.64
F. Scott Fitzgerald	<i>The Great Gatsby</i>	0.62
Toni Morrison	<i>Tar Baby</i>	0.52
Saul Bellow	<i>The Adventures of Augie March</i>	0.43
E.L. Doctorow	<i>Ragtime</i>	0.39
John Grisham	<i>The Runaway Jury</i>	0.32
William Golding	<i>Lord of the Flies</i>	0.14
Sylvia Plath	<i>The Bell Jar</i>	0.09
Alice McDermott	<i>Charming Billy</i>	-0.05
Eleanor H. Porter	<i>Pollyanna</i>	-0.18
Raymond Chandler	<i>The Big Sleep</i>	-0.19
Kate Douglas Wiggin	<i>Rebecca of Sunnybrook Farm</i>	-0.43
Ernest Hemingway	<i>The Old Man and the Sea</i>	-0.51
William Faulkner	<i>As I Lay Dying</i>	-0.60
P.L. Travers	<i>Mary Poppins</i>	-1.03

Table 6: difficulty rank (DR)(not normalized) of 26 novels in the Chicago Corpus. difficulty rank descending.

As in the RT experiment, we carried out linear regression with Recursive Feature Elimination (RFE) for predicting DR, these results are presented in table 7.

Features	F-stat	R^2	adj. R^2	p-val
Styl/Synt	8.908	0.548	0.487	< 0.01
Sentiment	2.469	0.252	0.150	0.09
All	7.955	0.52	0.455	< 0.01

Styl/Synt	<i>Freq “of”, Perplexity, Word Entropy</i>
Sentiment	<i>All sentiment features used</i>
All	<i>Freq “of”, Nominal Ratio, Word Entropy</i>

Table 7: Linear model **predicting difficulty rank** of novels using two feature sets, the three sentiment features, three selected stylistic/syntactic features, and three selected features among all features. Below, the selected features in each category using RFE.

C Collinearity reduction

To avoid overfitting our feature selection method to the small datasets in the regression models above, we fitted a PCA on the *Chicago Corpus* and projected features in the smaller regression datasets to its first 3 principal components. PCA also helps us avoid the curse of collinearity in regression models, therefore the reported statistics might be more representative of the features’ true

predictive strength. For reading times, results are in table 8; for difficulty rank in table 9.

Features	F-stat	R^2	adj. R^2	p-val
Styl/Synt	13.0	0.629	0.581	< 0.01
Sentiment	6.050	0.441	0.368	< 0.01
All	12.74	0.624	0.575	< 0.01

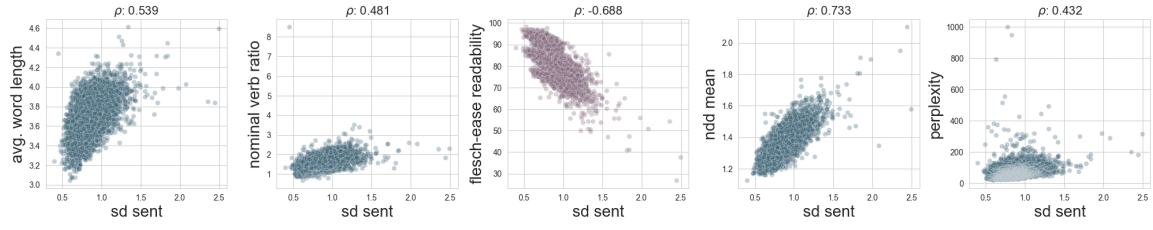
Table 8: Linear model **predicting difficulty rank** of novels using feature sets reduced for collinearity by fitting it to the *Chicago Corpus* PCA (3 components).

Features	F-stat	R^2	adj. R^2	Prob. F-stat
Styl/Synt	84.56	0.977	0.965	< 0.01
Sentiment	18.81	0.904	0.856	< 0.01
All	78.24	0.975	0.963	< 0.01

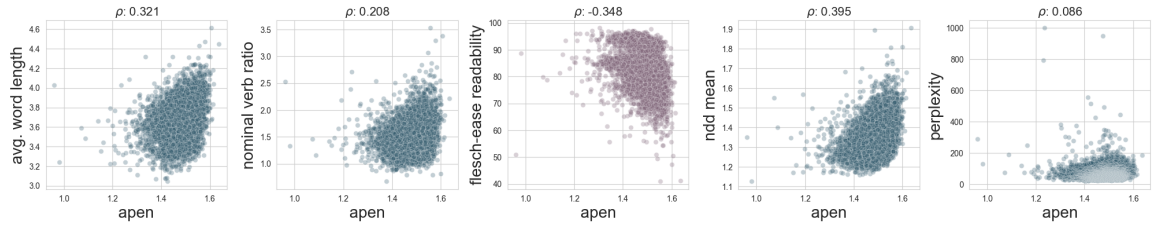
Table 9: Linear model **predicting reading time** of stories using feature sets reduced for collinearity by fitting it to the *Chicago Corpus* PCA (3 components).

D Features

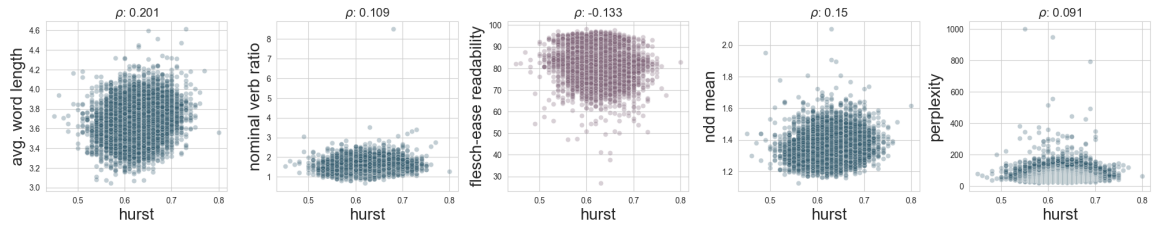
The full set of features with corresponding labels is indexed in table 10.



(a) Correlation between Sentiment SD and stylistic/syntactic features.

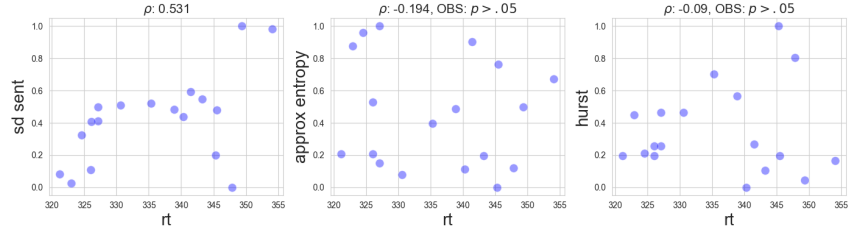


(b) Correlation between $ApEn$ and a few stylistic/syntactic features.

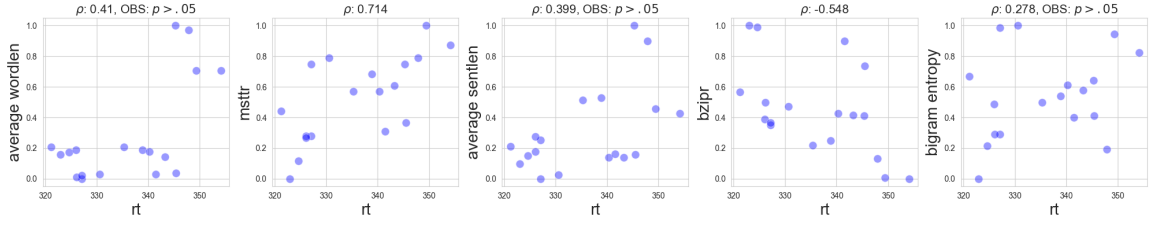


(c) Correlation between Hurst exponent and a few stylistic/syntactic features.

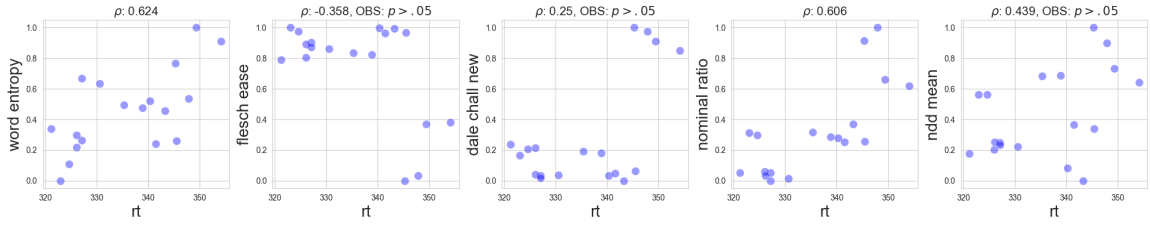
Figure 7: Correlation between **sentiment complexity features** and a few **stylistic or syntactic complexity features**. Note Spearman's ρ at the top of plots.



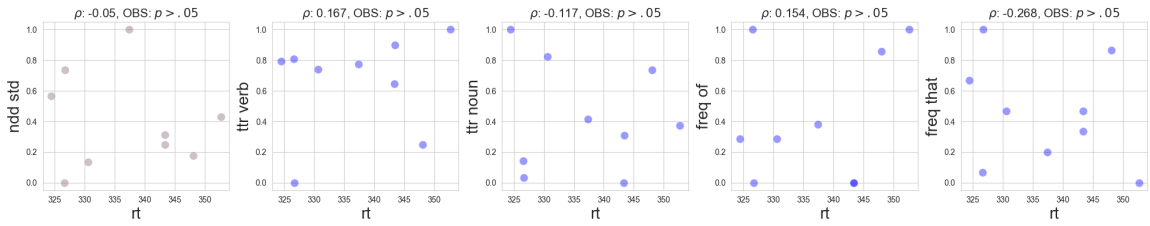
(a) Correlation between RT and **sentiment** features.



(b) Correlation between RT and stylistic/syntactic features.

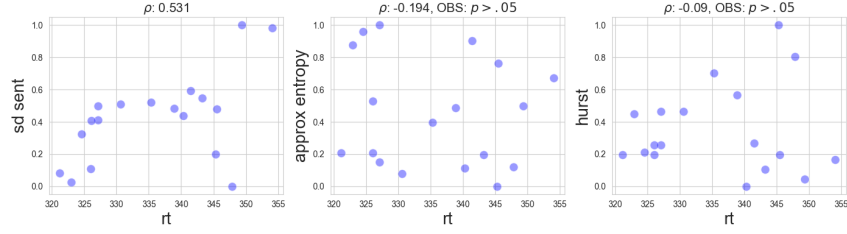


(c) Correlation between RT and stylistic/syntactic features.

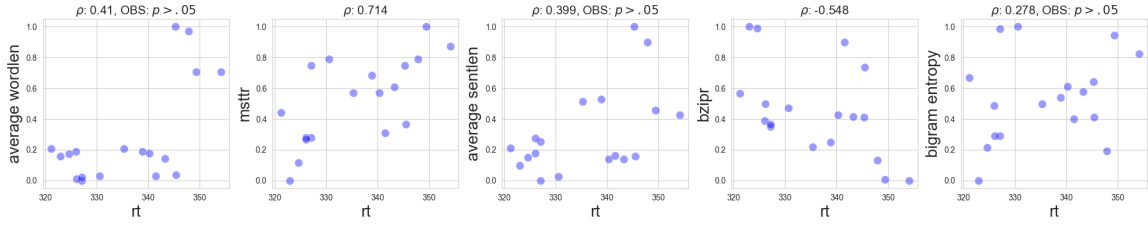


(d) Correlation between RT and stylistic/syntactic features.

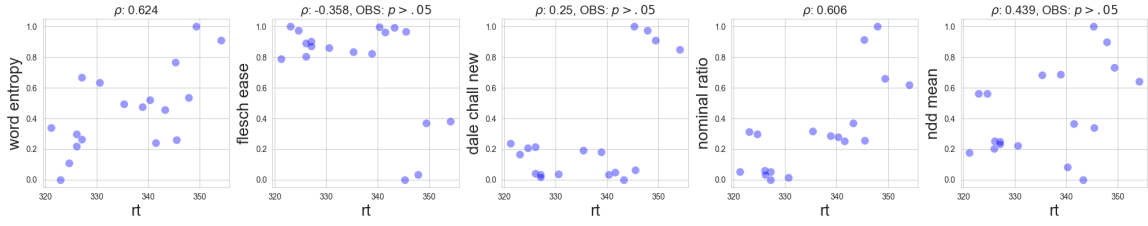
Figure 8: Full visualization of the correlation of features and RTs (10 stories).



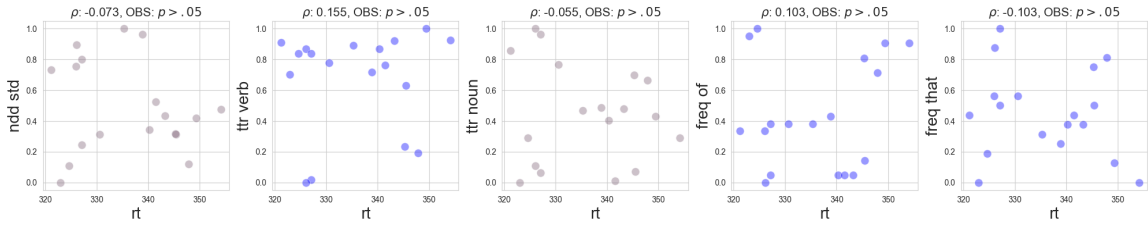
(a) Correlation between RT and **sentiment** features.



(b) Correlation between RT and stylistic/syntactic features.



(c) Correlation between RT and stylistic/syntactic features.



(d) Correlation between RT and stylistic/syntactic features.

Figure 9: Correlation features and RT, **augmented datapoints**. We split stories in two with a 90% overlap. This duplication of datapoints serve to show that the scarcity of correlations between features and RT may be due to a low number of datapoints (10 stories).

Feature	Description	Type	Reference
Type-Token Ratio (MSTTR-100), TTR Noun, TTR Verb	Measures lexical diversity by comparing the variety of words (types) to the total number of words (tokens), indicating a text’s vocabulary complexity and inner diversity. A high TTR represents a richer prose: a higher diversity of elements and a lower lexical redundancy (Torruella and Cap-sada, 2013). TTR of nouns or of verbs quantifies the diversity within these Parts-of-Speech categories. ^a	Stylistic	Forsyth (2000)*, Kao and Jurafsky (2012)*, Algee-Hewitt et al. (2016), Maharjan et al. (2017), Koolen et al. (2020), Brotrager et al. (2022), Ja-cobs and Kinder (2022), Bizzoni et al. (2023c)
Readability (R Flesch Ease, R Dale Chall)	Estimate reading difficulty based variously on sentence length, syllable count, and word length/difficulty. Assessed using five different classic formulae that remain widely used (Stajner et al., 2012). ^b	Stylistic	Martin (1996), Garthwaite (2014), Ma-harjan et al. (2017), Febres and Jaffe (2017), Zedelius et al. (2019)*, Berger et al. (2021)*, Brotrager et al. (2022), Bizzoni et al. (2023b)
Compressibility	Measures the extent to which the text can be compressed, serving as an in-direct indicator of redundancy and lexical variety (Ehret and Szmrecsanyi, 2016). ^c	Stylistic	van Cranenburgh and Bod (2017), Koolen et al. (2020), Bizzoni et al. (2023c)
Word and bigram entropy	Measures the unpredictability in word choices and combinations, with higher entropy indicating greater variety and stylistic complexity.	Stylistic	Algee-Hewitt et al. (2016)
Normalized De-pendency Distance, mean & SD (NDD Mean, NDD STD)	Quantifies the mean and SD in dependency length, following the pro-cedure proposed in Lei and Jockers (2020) .	Stylistic/ Syntactic	Lei and Jockers (2020)
Nominal verb ratio	Quantifies the proportion of nouns and adverbs (over verbs) in the text, re-reflecting the nominal tendency in style, which is often associated with com-plex linguistic structures, denser communicative code, expert-to-expert communication (McIntosh, 1975; Bostian, 1983).	Stylistic/ Syntactic	Charney and Rayman (1989)*, Crossley et al. (2014)*, Wu et al. (2024)
“Of”/“that” frequencies	Frequency of these function words have been seen to indicate, in the case of “of”, a more nominal prose, and in the case of “that”, a more declarative and verb-centered prose.	Stylistic/ Syntactic	Wu et al. (2024)
Function words	Frequency of function words (normalized for text length), suggesting a more information-rich prose when lower.	Stylistic/ Syntactic	Bizzoni et al. (2024a)
Perplexity	Represents the predictability of the prose through a self-trained large lan-guage models (GPT), as outlined in Wu et al. (2024). ^d Higher values indicate greater complexity or unpredictability.	Hybrid	Sheetz (2018), Wu et al. (2024), Wu et al. (2024)
Sentiment SD (SD Sent)	Represents the average variability in sentiment, indicating the range of sentiment within the narrative. ^e	Narrative/ Sentiment	Berger et al. (2021)*, Bizzoni et al. (2023c)
Hurst exponent	Quantifies the long-term auto-correlation of the sentiment arc, ^e with higher values suggesting a more complex, self-similar structure across different scales. ^f	Narrative/ Sentiment	Mohseni et al. (2021), Bizzoni et al. (2021), Bizzoni et al. (2023d)
Approximate en-tropy (APEN)	Assesses the predictability of sequences of the sentiment arc, ^e with lower values indicating greater regularity or simplicity. ^f	Narrative/ Sentiment	Hu et al. (2020), Mohseni et al. (2022), Bizzoni et al. (2023c)

Table 10: **Used features related to stylistic and sentiment complexity.** “References” refer to studies that have used the complexity feature showing some relation between it and reader appreciation. * Denotes studies in domains other than *established prose fiction* (e.g., online stories, movies).

^a We used a common method insensitive to text length: the Mean Segmental Type-Token Ratio (MSTTR). MSTTR-100 represents the overall average of the local averages of 100-word segments of each text.

^b Flesch Reading Ease and New Dale–Chall Readability Formula.

^c We calculated the compression ratio (original bit-size/compressed bit-size) for the first 1500 sentences of each text using bzip2, a standard file-compressor.

^d All perplexity calculations were via gpt2 models, done on the byte pair encoding tokenization used in the series of gpt2 models. To get the mean perplexity per novel, we used a sliding window due to maximum input length. For details on the computation, see Wu et al. (2024).

^e All sentiment analysis was performed using the *Syuzhet* implementation on a sentence-basis (compound score).

^f For details on the measure, please refer to Bizzoni et al. (2023d).

Does Preprocessing Matter? An Analysis of Acoustic Feature Importance in Deep Learning for Dialect Classification

Lea Fischbach¹ and Caroline Kleen¹ and Lucie Flek^{2,3} and Alfred Lameli¹

¹ Research Center Deutscher Sprachatlas, Philipps-Universität Marburg

{lea.fischbach, caroline.kleen, lameli}@uni-marburg.de

² Bonn-Aachen International Center for Information Technology (b-it), University of Bonn

³ The Lamarr Institute for Machine Learning and Artificial Intelligence

flek@bit.uni-bonn.de

Abstract

This paper examines the effect of preprocessing techniques on spoken dialect classification using raw audio data. We focus on modifying Root Mean Square (RMS) amplitude, DC-offset, articulation rate (AR), pitch, and Harmonics-to-Noise Ratio (HNR) to assess their impact on model performance. Our analysis determines whether these features are important, irrelevant, or misleading for the classification task. To evaluate these effects, we use a pipeline that tests the significance of each acoustic feature through distortion and normalization techniques.

While preprocessing did not directly improve classification accuracy, our findings reveal three key insights: deep learning models for dialect classification are generally robust to variations in the tested audio features, suggesting that normalization may not be necessary. We identify articulation rate as a critical factor, directly affecting the amount of information in audio chunks. Additionally, we demonstrate that intonation, specifically the pitch range, plays a vital role in dialect recognition.

1 Introduction

In the realm of deep learning, preprocessing plays a crucial role in optimizing model performance. While many studies focus on text, like (Uysal and Gunal, 2014), others concentrate on Environmental Sound Classification (ESC) or Automatic Speech Recognition (ASR) (Pfau et al., 2000). For instance, Bansal and Garg (2022) are exploring existing papers on preprocessing for ESC. Additionally, some studies focus on using spectrograms for audio processing (Chaiyot et al., 2021). Moreover,

some research has attempted to enhance speech recordings for dialect identification, leading to improved subjective quality (Kakouros et al., 2020). However, these studies did not evaluate whether such preprocessing techniques actually improve the performance in downstream tasks.

Furthermore, despite studies such as (Lounnas et al., 2022), which incorporate noise reduction as a preprocessing step for dialect identification, a comprehensive study on the key aspects of audio preprocessing for dialect identification remains lacking. Often, only individual aspects of preprocessing are considered, as seen in (Pfau et al., 2000), where vocal tract length normalization (VTLN) and speech rate normalization (SRN) are examined.

Large-scale systems such as Whisper (Radford et al., 2023) and Meta’s Massively Multilingual Speech (MMS) project (Pratap et al., 2024) highlight the power of extensive and diverse datasets in advancing ASR and language identification. Whisper, trained on 680,000 hours of multilingual and multitask supervised data, achieves improved robustness to accents, background noise, and technical language, demonstrating the impact of its large dataset. Similarly, Meta’s MMS project tackles the lack of ASR systems for many languages by using religious texts, translated into numerous languages, to build a diverse training dataset. These projects showcase the importance of large datasets for robustness and inclusivity. In contrast, this study addresses the challenges of working with smaller, constrained datasets.

Notably, no paper has been found that investigates the effects of preprocessing raw audio on language or dialect classification. This gap is particularly significant in the context of deep learning-based dialect identification (DID), where understanding the fundamental aspects of audio preprocessing tailored specifically for dialect classification remains under-explored. This issue resonates with

findings in music information retrieval research, where deep learning efforts often prioritize optimizing hyperparameters that define network structure, while the audio preprocessing stage is often not optimized (Choi et al., 2018).

This study aims to bridge this gap by investigating how preprocessing adjustments affect the performance of dialect classification models trained on German audio data. We concentrate on the raw waveform and underscore the importance of different audio features in dialect classification. Specifically, we aim to determine whether adapting audio inputs improves model performance and whether certain features are misleading for the model, causing it to learn irrelevant patterns. Additionally, we explore if deep learning models inherently learn to ignore such variations or if performance even worsens, indicating that these features are important for dialect recognition in German.

Our contributions are threefold:

- We demonstrate that deep learning models for dialect classification are immune to variations in the tested audio features, suggesting that normalizations are not necessary.
- We reveal that the amount of information in an audio chunk is related to the Articulation Rate, impacting model performance.
- We show that intonation, specifically the pitch range within an audio chunk, is important for dialect recognition.

To achieve these contributions, we employ a pipeline that analyzes the significance of various acoustic features, representing a novel approach in the field.

By focusing on these aspects, our work not only fills a significant gap in the existing literature but also provides valuable insights for future research and applications in dialect classification using deep learning.

2 Used Acoustic Features

Used Acoustic Features are **Root Mean Square (RMS) amplitude**, **DC-offset**, **Articulation Rate (AR)**, **Pitch**, and **Harmonics-to-Noise Ratio (HNR)**.

RMS amplitude of a digital audio signal represents its perceived loudness and is simultaneously the mean absolute value of the signal. While RMS measures the average power of a signal, intensity in decibels (dB) quantifies the power relative to a

reference level, typically the threshold of human hearing, on a logarithmic scale. As these metrics are correlated, only RMS is considered in this study.

RMS amplitude reflects both the speaker’s vocal effort and external factors such as the recording equipment and the recording environment, including background noise and microphone distance.

DC-offset (also known as DC-bias), determined by the average amplitude of a segment of the signal, indicates a deviation from the symmetrical nature of a normal voice signal. In a typical symmetric sine signal, the high peak equals the low peak, resulting in an average value near zero over time. However, when a DC offset is present, the symmetry is disrupted, and the average value deviates from zero¹. Despite being imperceptible, it reduces the available dynamic range, limiting the signals amplitude variation. DC-offset is primarily influenced by the recording equipment rather than the speaker.

Articulation Rate (AR) measures syllables per second during speech, excluding pauses, whereas Speech Rate (SR) includes pauses in its calculation. In this study, AR is emphasized over SR, as the audio data has been preprocessed to exclude pauses and non-articulatory elements. Also Otto (2012) states that variations in articulation speed between speakers may be more indicative of individual speaking styles than differences in overall speech tempo. The AR regarding to regional distribution has been minimally investigated thus far. Hahn and Siebenhaar (2016) found that there are differences in AR, but also suggest that this may correlate with other processes such as the elision of segments. They conclude that there must be different sound duration ratios in the different regions.

Pitch, often referred to synonymously as F0, stands for the fundamental frequency of a sound wave. F0 refers to the physical oscillation, while pitch denotes the perceived tonal height of the sound. In tools such as Praat (Boersma and Weenink, 2021), the pitch refers to F0. Pitch normalization, akin to Vocal Tract Length Normalization (VTLN), aims to mitigate speaker-specific variations in speech signals attributed to differences in vocal tract lengths, which are influenced by physiological factors such as sex. In explor-

¹<https://solicall.com/dc-offset-and-audio-filtering/>

ing the connection between pitch and dialects, it’s noteworthy that the typical fundamental frequency doesn’t always align directly with dialect variations. Instead, phenomena such as variations in voice quality due to dialectal influences can affect pitch.

The **Harmonics-to-Noise Ratio (HNR)** quantifies the relationship between periodic components and noise in a signal. It measures acoustic periodicity by comparing the energy of harmonics to that of noise, with the result expressed in decibels (dB), indicating the dominance of periodic components over noise: an HNR of 20 dB signifies 99% of energy in periodic components and 1% in noise, calculated as $10 * \log_{10}(99/1)$. An HNR of 0 dB indicates equal energy distribution between harmonics and noise². In speech analysis, HNR is favored over Signal-to-noise ratio (SNR) for its ability to capture voiced sounds periodicity. HNR primarily reflects characteristics of the speaker’s voice, such as vocal cord vibration regularity and voice quality, but can also be affected by the recording equipment and environmental noise.

3 Experimental Setup

3.1 Used Corpus

This study utilizes automatically segmented audio files (Fischbach, 2024) sourced from the “Regionalsprache.de” (REDE) corpus (Schmidt et al., 2020ff.). The REDE corpus, which consists exclusively of recordings from male speakers, includes recordings from three age groups: young (18–23 years), middle-aged (45–55 years), and older (65+ years) speakers, captured across five different recording situations³.

However, for the purposes of this study, only data from the older generation (65+ years) is analyzed. They are chosen due to their presumed higher dialect competence and to save computing time using only one generation. Furthermore, we only utilize the so-called dialectal “Wenker Sentences”⁴ from the corpus. In this recording situation, an interviewer reads 40 sentences in Standard German, and the dialectal speakers translate these sentences

into their local dialect. In total there are around 18 hours of audio data from the older generation and this recording situation, consisting of audios featuring only the dialectal speakers.

For classification we analyze a total of 20 different German dialects, classified according to Wiesinger (1983) without the transition areas between dialects. Dialects with insufficient variance (less than 3 speakers per dialect) are not further considered.

3.2 Classification Pipeline

The described pipeline is available and visualized on GitHub⁵. Initially, all audio files are preprocessed to standardize their format by converting them to mono, adjusting the bit-depth to 16 bits, and setting the sampling rate to 16 kHz, in line with the specifications of Google’s TRILLsson models (Shor and Venugopalan, 2022), which is used for embedding extraction. The audio files are then divided into 10-second chunks for the extraction of these embeddings. Prior tests have shown this duration to be optimal. Shorter chunks yielded significantly poorer results, likely due to insufficient contextual information, whereas longer chunks offered no further gains, as the additional information in extended audio segments made 10 seconds sufficient. The resulting embeddings are processed through a small convolutional neural network (CNN) consisting of three dense layers with LeakyReLU activations and dropout layers to prevent overfitting. The network is trained using the Adam optimizer (Kingma and Ba, 2015). For model validation and testing, $\lceil \frac{\#S_D}{10} \rceil$ speakers are randomly selected from each dialect, where $\#S_D$ represents the total number of speakers in the respective dialect. To account for variability in results due to different speaker selections, we employ a Monte Carlo cross-validation approach, repeating the data splitting and model evaluation process 250 times with new random speaker selections in each run. This number of iterations was chosen based on prior tests demonstrating its effectiveness in detecting significant differences between experiments. The mean of the weighted F1-score across runs is calculated, and the Mann-Whitney U test (Mann and Whitney, 1947) is used to assess the statistical significance of performance differences between runs.

²<https://www.fon.hum.uva.nl/praat/manual/Harmonicity.html>

³Additional information about the recording situations, the recording locations and the project itself can be found on <https://rede-infothek.dsa.info/>

⁴<https://www.uni-marburg.de/en/fb09/dsa/research-documentation-center/wenkersaetze>

⁵<https://github.com/WoLFi22/DialectClassificationPipeline>

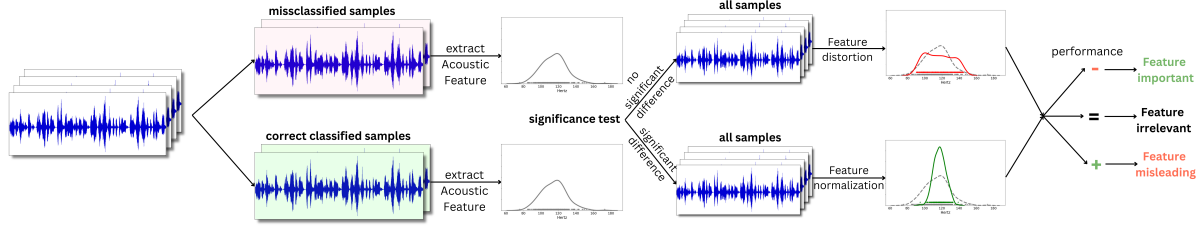


Figure 1: Visualization of the procedure.

3.3 Procedure

The entire procedure is summarized in the pipeline diagram shown in Figure 1, which outlines the steps involved in evaluating the significance of different features for dialect classification. Initially, we conduct analyses to determine significant differences in the distribution of feature values between two groups of audio chunks: those with a high misclassification rate (misclassified in 95%-100% of 250 runs, referred to as "wrongly classified") and those that are frequently classified correctly (correctly classified in 65%-100% of 250 runs, referred to as "correctly classified"). These thresholds are chosen to ensure a balanced representation of both incorrect and correct chunks. This can be inferred from the diagram in Figure 2. The diagram illustrates how many chunks are classified correctly and incorrectly at which percentage threshold, and the ratio of the number of incorrect to correct chunks.

Model performance is evaluated using the weighted F1-score to account for the imbalanced class distribution. If a significant difference is found between the distributions of features (such as pitch) for incorrectly and correctly classified chunks — determined using the Mann-Whitney U test, where a $p\text{-value} < 0.05$ indicates a significant difference — all chunks are normalized according to that specific feature. If the difference is not significant, the chunks are deliberately distorted to assess whether this manipulation affects the model's performance. This approach helps to identify whether a particular feature is important, irrelevant, or even misleading for the deep learning model in classifying (German) dialects. The rationale behind these assessments, such as why feature distortion resulting in decreased model performance indicates the feature's importance, is summarized in Table 1.

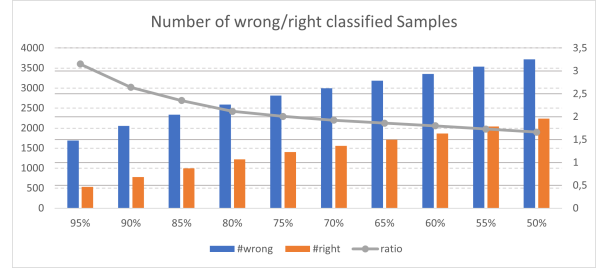


Figure 2: Ratio and number of wrongly and correctly classified chunks with percentage threshold values.

3.4 Feature Extraction

The features are computed using Praat (Boersma and Weenink, 2021) via the Parselmouth Python interface (Jadoul et al., 2018), which facilitates Praat script execution in Python, as detailed below:

RMS Amplitude Extraction: The RMS value is computed using Praat's `Get root-mean-square` function.

DC-Offset Extraction: The DC-offset is calculated as the mean value of the audio chunk.

AR Extraction: We employ a multi-step process to extract the articulation rate. Initially, all audio chunks are peak-normalized to standardize intensity levels, enhancing the accuracy of the syllable recognition algorithm. Following normalization, we use a Praat script from the *Praat Vocal Toolkit* (Corretge, 2012-2024), which identifies syllable nuclei while discarding non-voiced peaks, to mark syllables in the audio segments. This Praat script is described by De Jong and Wempe (2009). The articulation rate is then extracted as the ratio of the number of syllables to the phonation time.

Peak normalization before AR extraction is needed, to address the incorrect detection of pauses in audio chunks where none should exist. As illustrated in Figure 3 a), many pauses were falsely identified in places where speech is present due to fluctuations in intensity, highlighting the

Feature Distortion			Feature Normalization		
-	=	+	-	=	+
Distortion degrades performance as the model relies on the original distribution.	The Feature is irrelevant; distortion has no effect.	Distortion removes misleading information, improving performance.	Normalization removes useful distinctions, degrading performance.	Differences in distribution are irrelevant; no effect.	Normalization reduces noise or bias, improving performance.

Table 1: Effects of Feature Distortion and Normalization on model performance (+ improved, - degraded, = unchanged), indicating the role of the Feature in the model, as in Figure 1.

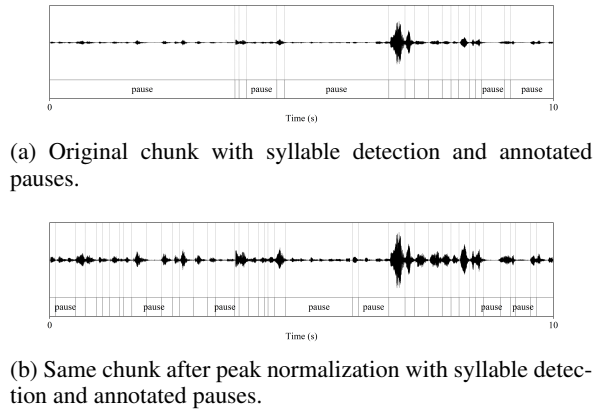


Figure 3: Audio chunk and its detected syllables/pauses.

algorithm’s sensitivity to sound levels. Peak normalization improves syllable recognition by stabilizing these fluctuations, which is particularly beneficial for speech rate (SR) detection, as it is more affected by misclassified pauses compared to AR, but it also improves AR performance.

Figure 3 b) shows the same audio chunk after peak normalization was applied before running the syllable recognition algorithm. Although some incorrect pauses remain, the results are significantly more accurate. Across all chunks, this process reduced the standard deviation of the ratio of speaking duration to audio length (which ideally should be close to 1 for our dataset, as there should be no or only very short pauses), bringing more values closer to 1 and minimizing extreme deviations.

Additionally, our use of 10-second chunks ensures a stable extraction of AR, aligning with findings from Arantes and Lima (2017), where they state that both SR and AR stabilize after approximately 9 seconds.

Pitch Mean and Pitch Standard Deviation

Extraction: We calculate both the mean pitch and the standard deviation of the pitch restricting the analysis to a range of 80 Hz to 170 Hz. Pitch values are extracted using the *To Pitch* function in Praat, followed by the computation of either the mean or the standard deviation.

The pitch range of 80-170 Hz is selected, because Praat’s default settings for pitch extraction often result in high pitches values (up to 240 Hz) and large fluctuations (up to 100 Hz within a chunk), leading to a high standard deviation (± 44.74 Hz). This issue is likely due to flaws in the underlying algorithm (Boersma et al., 1993). Adjusting the pitch range to match the typical frequency range for the speaker’s sex (and age) can mitigate this problem and ensures more accurate pitch detection.

The default pitch range in Praat is set between 75 Hz and 600 Hz. This range can be narrowed to 80-170 Hz, which corresponds to the normal pitch range for male speakers. For instance, a study involving 2472 German-speaking men aged 40–79 years found that the mean fundamental frequency of the conversational speaking voice was 111.9 Hz, with specific averages of 112.9 Hz (± 17.5) for ages 60–69 and 120.6 Hz (± 19.8) for ages 70–79 (Berg et al., 2017). Another study reported a mean pitch of 120 Hz (± 18 Hz) for the German male reading voice (Andreeva et al., 2014). Our adjusted pitch range of 80–170 Hz is therefore well-supported by these findings.

With the new settings, the largest deviation within a chunk decreased by nearly one-third to 34.13 Hz, and the standard deviation was reduced by more than half to 19.09 Hz. Figure 4 visualizes the results of pitch extraction using the two

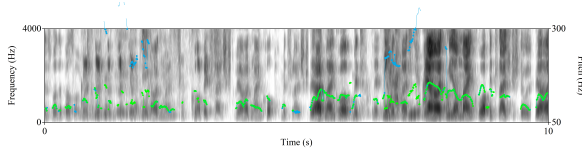


Figure 4: Extracted Pitch with Praat for different pitch ceilings: 75-600 Hz (blue) and 80-170 Hz (green).

ranges: the extracted pitch using standard settings (75-600 Hz) is shown in blue and in green with adjusted settings (80-170 Hz). It is evident that the blue pitch values are often too high, while the green ones are much more stable. By narrowing the pitch range, the algorithm is constrained to estimate the pitch within these plausible bounds, thereby providing results closer to the true pitch.

HNR Extraction: The HNR gets extracted using Praat’s `To Harmonicity (cc)` function.

4 Results

Table 2 summarizes the feature importance analysis. The *p-val* column displays the p-values from Mann-Whitney U tests, comparing feature values between correctly and wrongly classified chunks; a p-value below 0.05 indicates a statistically significant difference. The *Norm./Dist.* column indicates whether the audio chunks were normalized or disturbed, with the *Method* column detailing the specific processing method. The *new Perf.* column presents the mean weighted F1-Score of the model with altered audio chunks, compared to the original score of 0.228. The *Perf. p-val* column contains the p-value from the Mann-Whitney U test comparing the model’s performance with original versus altered chunks, indicating the feature’s impact on performance, as shown in the *Feat. Imp.* column.

4.1 RMS

In the analysis of Root Mean Square (RMS) amplitude, statistically significant distinct distributions can be observed between wrongly and correctly classified chunks with a lower mean RMS for wrongly classified chunks. So RMS gets normalized for each chunk to assess the impact of RMS values on classification results. Peak normalization, which adjusts audio signals relative to their loudest point and has been set so that the highest peak reaches -0.2, and loudness normalization, which aims to standardize perceived loud-

ness, are explored. Where for loudness normalization there is a risk of clipping if the target value is set to high. Both normalization methods resulted in an overall increase in loudness, as depicted in Figure 5 a), with loudness normalization demonstrating a notably reduced standard deviation due to its uniform adjustment to a target loudness level of -14dB.

Neither of the two methods yields a significant difference in classification performance. In both normalization methods, approximately the same errors are observed in assigning chunks to dialects as without normalization.

This finding supports the theory that the initially different distribution of correctly and incorrectly classified chunks was coincidental. Further tests have shown that speakers with almost the same misclassification rate from the same dialect can have very different RMS, likely due to varying recording conditions and consequently different RMS levels. Therefore, the differences in the distributions of correctly and incorrectly classified chunks are likely due to variations in the classification performance of individual speakers, rather than differences in RMS levels. Therefore, it is reasonable to conclude that the volume of individual chunks does not influence the model’s performance, rendering this feature irrelevant.

4.2 DC-Offset

There is a significant difference in distributions of wrongly and correctly classified chunks. To address this, a normalization technique called *Mean Centering* is employed by subtracting the mean of each chunk, effectively minimizing the DC-Offset. However, this normalization yields no difference in classification performance.

Yet, since even the largest offset in our data is minimal (-0.0007), it should be tested again whether a disturbance of the DC-Offset leads to a deterioration in classification performance. Each chunk gets a randomized disturbance within the range of [-0.1, 0.1], which also can be seen in Figure 5 b) at the bottom. However, this perturbation fails to yield any discernible difference in model performance, suggesting that the DC-offset holds little relevance to classification performance, as long as it is in normal ranges.

Feature	p-val	Norm./Dist.	Method	new Perf.	Perf. p-val	Feat. Imp.
RMS	0.000	Norm.	RMS von -14dB	0.287	0.104	=
			Peak of -0.2	0.283	0.306	=
DC-Offset	0.000	Norm.	Mean Centering	0.280	0.635	=
		Dist.	[-0.1, 0.1]	0.283	0.321	=
AR	0.936	Dist.	[2.5, 6] Syll./Sec.	0.259	0.000	-
		Norm.	4.3 Syll./Sec.	0.277	0.870	=
Pitch mean	0.235	Dist.	[90, 160] Hz	0.276	0.626	=
Pitch std	0.012	Norm.	Half orig. Std.	0.236	0.001	-
			Monotonized	0.241	0.000	-
			Std. of 18	0.269	0.087	=
HNR	0.000	Norm.	Praat	0.283	0.127	=
			Noisereduce	0.270	0.419	=

Table 2: Summary of feature importance analysis, including p-values from Mann-Whitney U tests comparing feature values between correctly and incorrectly classified chunks (p-val), processing methods applied to normalize or disturb features (Norm./Dist. and Method), mean weighted F1-Score with altered audio chunks, p-values comparing model performance with original and altered audio chunks (Perf. p-val), and the resulting feature importance (Feat. Imp.).

4.3 AR

The distributions of the wrongly and correctly classified chunks are similar, so AR perturbation is conducted to assess its impact on model performance. AR values are intentionally disturbed between 2.5 and 6 syllables per second, derived from extreme measurements from all chunks as can be seen in the top box plot of Figure 5 c). Model performance significantly declines with disturbed AR chunks compared to normal conditions. To ascertain whether this decline stems solely from extreme differences between chunks or generally from extreme AR values, additional tests are conducted. These involve assessing the model’s behavior with only slow (AR of 3.0) or fast (AR of 6.0) chunks. When the articulation rate is reduced, resulting in slower audio, the chunks are still formed with a fixed length of 10 seconds. As a result, there are more chunks overall, but each chunk contains less information due to the lower tempo. With higher AR, there are fewer chunks, but each chunk contains more information. Chunks with lower AR lead to a 44% increase in length and degraded model performance. Conversely, chunks with higher AR, approximately 71.36% shorter than the original recordings, are showing no significant difference. Considering that longer chunks generally contain

more information, this could explain why the classification performance did not deteriorate or improve with a faster AR, as the model’s performance in earlier tests also did not benefit from chunks longer than 10 seconds. Moreover, reducing the length of chunks with the higher AR up to 8 seconds does not yield a significant difference in model performance. However, caution should be exercised not to increase the AR too much. Another test using chunks of 7 seconds with double the AR compared to the original mean, resulting in an AR of 8.734, shows a significant deterioration. These findings suggest that, to a certain extent, manipulating AR to increase chunk speed can be effective in shortening chunk length for reduced computational workload without compromising classification performance. This approach has the potential to conserve computational resources during classification tasks. Nevertheless, the tradeoff between increased AR and reduced audio length is limited. If the speech speed is too slow, longer chunks should instead be used to ensure sufficient information is captured. Therefore, it is assumed that AR does not influence dialect classification, but rather the amount of information contained in each chunk. This also agrees with De Jong and Wempe (2009) where they state that speech recognizers perform relatively poorly

when speech rate is very fast or very slow. Nevertheless, we aim to normalize the AR, as suggested by Pfau et al. (2000), where Speech Rate Normalization resulted in a reduction of word error rate. To normalize the AR, all chunks are speed-manipulated based on their original AR. The median AR across all audio chunks is 4.36 and the mean is 4.37. Therefore, all audios should have an articulation rate of approximately 4.3. To achieve that, first, the factor between the current and the desired AR is determined using $factor = AR_{old}/AR_{new}$, and then the original audio chunk is speed-manipulated by this factor (factor < 1 results in the audio being faster). Through this approach, the AR is slightly reduced on average, resulting in a slowdown of most audios, as can be seen at the lower mean in Figure 5 c). Normalizing the AR had no impact on the classification performance.

4.4 Pitch

Since there is no significant difference between wrongly and correctly classified chunks where values are extracted with the adjusted pitch range, further testing is conducted to determine if the model's performance would degrade when the pitch is randomly altered. The pitch is varied between 90-160 Hz, a range considered normal for male speaking voice and providing headroom in both directions for pitch extraction with Praat. Despite this manipulation, no significant difference in classification performance can be observed. These findings indicate that pitch does not significantly impact classification.

Additionally, we investigate how the model behaves when adjusting the pitch range by altering the standard deviation. The distribution of pitch standard deviations between correctly and incorrectly classified chunks differs significantly. Specifically, the mean standard deviation of pitch is higher for incorrectly classified chunks. To address these differences, we normalize the pitch range of each chunk. We test several approaches: halving the pitch range, monotonizing the pitch, and normalizing it to a standard deviation of 18 Hz. The model's performance significantly deteriorate when the pitch range is halved or monotonized, while normalization to 18 Hz standard deviation shows no impact on performance. These results, depicted in Figure 5 e), suggest that pitch variation is important for dialect classification, but

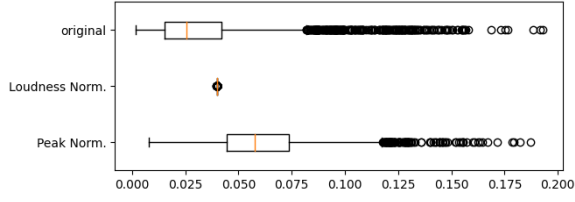
only its intonation (the variance in pitch) rather than its exact magnitude.

The importance of pitch in language and dialect classification is further highlighted by Vicens and Sundara (2013), where features such as minimum, maximum, and mean pitch, as well as the number and characteristics of pitch rises, were used to distinguish between German and American English with an accuracy of 86%. Notably, these features primarily captured pitch variance rather than absolute values, emphasizing the significance of intonation. The study also demonstrated that these pitch features could differentiate between varieties of English, such as American and Australian English, achieving an accuracy of 79%. Their study also concludes that intonation plays a crucial role in helping listeners distinguish between different languages.

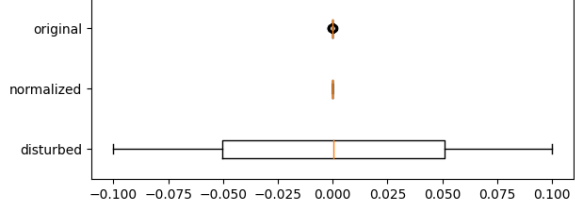
4.5 HNR

Statistically significant differences in the distributions of HNR values are observed between correctly and wrongly classified chunks, with the mean HNR slightly higher for the latter. As illustrated in Figure 5 f), the majority of our audio chunks have a mean HNR value indicating approximately 90% harmonic content, though some chunks exhibit lower HNR values with around 70% harmonic content. Due to the absence of stationary background noises applying a constant bandpass filter is not feasible. Furthermore, since the recordings were downsampled to 16 kHz, any noise above 8 kHz is already filtered out.

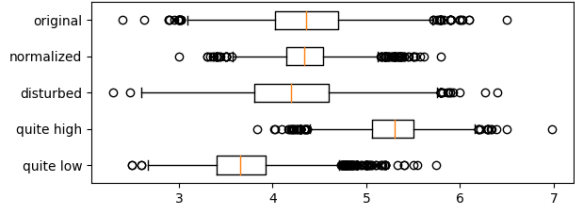
Attempts to reduce noise using Praat's spectral subtraction method, as defined by Boll (1979), yields no significant changes in HNR or improvements in classification performance. We also applied the `noisereduce` library (Sainburg et al., 2020; Sainburg, 2019). Non-stationary noise reduction is applied due to the absence of specific interfering noises, yet this also results in minimal changes in HNR and no significant difference in classification. This result is consistent with findings from Lounnas et al. (2022), where noise reduction using `noisereduce` showed no notable effect on classification performance when using Convolutional Neural Networks (CNNs). Thus, it can be concluded that non-stationary noises have little to no influence on the performance of dialect classification, as long as they do not obscure the speech signal.



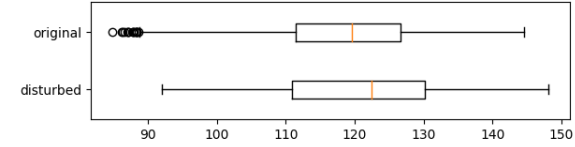
(a) original RMS and after normalization.



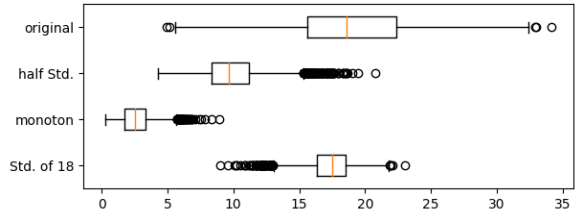
(b) original Mean chunk, after normalization and with random disturbance.



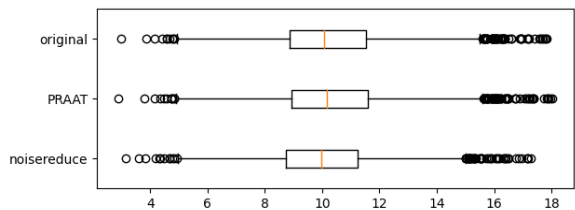
(c) original AR, with normalization, with random disturbance and high AR (6.0) and low (3.0) AR.



(d) original Pitch and with random disturbance.



(e) original STD of Pitch, original STD reduced by half, monotonized and normalized STD of Pitch to 18.



(f) original HNR, HNR reduced with Praat and HNR reduced with noisereduce.

Figure 5: Boxplots for different Audio-chunk features.

5 Discussion

The study's results indicate that pitch variation did not impact model performance among the used group of older males, suggesting its potential applicability across different age groups. However, it's uncertain if pitch normalization would have the same effect in a diverse group, where sex and age may introduce more pitch variation. Future studies should explore the impact of pitch normalization on mixed demographics and evaluate broader techniques such as voice conversion techniques to standardize all audio inputs.

Although the analysis focused on a German dialect dataset, these insights could extend to other corpora. Nonetheless, it is essential to conduct a thorough evaluation of each dataset's features to ensure that preprocessing techniques are well-suited to its specific characteristics and contribute to the classification tasks coherence and relevance.

The precise feature extraction values may vary depending on the extraction methods and parameters used. However, RMS and DC-offset measurements should consistently yield the same results, as these values can be accurately calculated. In contrast, when extracting pitch features, parameters such as the pitch floor and ceiling should be adjusted according to the age and sex of the speakers to obtain more accurate estimations of the true pitch.

Acknowledgements

This research is supported by the Academy of Science and Literature Mainz (grant REDE 0404), the German Federal Ministry of Education and Research (BMBF) (grant AnDy 16DKWN007), the state of North-Rhine Westphalia as part of the Lamarr-Institute for Machine Learning and Artificial Intelligence, LAMARR22B and the Research Center Deutscher Sprachatlas Marburg.

We thank the anonymous reviewers for their valuable feedback.

References

- Bistra Andreeva, Grazyna Demenko, Magdalena Wol-ska, Bernd Möbius, Frank Zimmerer, Jeanin Jügler, Magdalena Oleskowicz-Popiel, and Jürgen Trouvain. 2014. Comparison of pitch range and pitch variation in Slavic and Germanic languages. In *Proceedings to the 7th Speech Prosody Conference, Trinity College Dublin, Ireland, May 20-23, 2014*, pages 776–780. International Speech Communication Association.
- Pablo Arantes and Verônica Gomes Lima. 2017. Towards a methodology to estimate minimum sample length for speaking rate. *Revista do GEL*, 14(2):183–197.
- Anam Bansal and Naresh Kumar Garg. 2022. Environmental Sound Classification: A descriptive review of the literature. *Intelligent Systems with Applications*, 16:200115.
- Martin Berg, Michael Fuchs, Kerstin Wirkner, Markus Loeffler, Christoph Engel, and Thomas Berger. 2017. The Speaking Voice in the General Population: Normative Data and Associations to Sociodemographic and Lifestyle Factors. *Journal of Voice*, 31(2):257.e13–257.e24.
- Paul Boersma and David Weenink. 2021. Praat: doing phonetics by computer [Computer program]. Version 6.1.38, retrieved 2 January 2021 <http://www.praat.org/>.
- Paul Boersma et al. 1993. Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound. In *Proceedings of the institute of phonetic sciences*, volume 17, pages 97–110. Amsterdam.
- Steven Boll. 1979. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on acoustics, speech, and signal processing*, 27(2):113–120.
- Krittaya Chaiyot, Supattra Plermkamon, and Thana Radpukdee. 2021. Effect of audio pre-processing technique for neural network on lung sound classification. In *IOP Conference Series: Materials Science and Engineering*, volume 1137. IOP Publishing.
- Keunwoo Choi, György Fazekas, Mark Sandler, and Kyunghyun Cho. 2018. A comparison of audio signal preprocessing methods for deep neural networks on music tagging. In *2018 26th European Signal Processing Conference (EUSIPCO)*, pages 1870–1874. IEEE.
- Ramon Corrette. 2012-2024. Praat Vocal Toolkit. retrieved 20 January 2024 <https://www.praatvocaltoolkit.com>.
- Nivja H De Jong and Ton Wempe. 2009. Praat script to detect syllable nuclei and measure speech rate automatically. *Behavior research methods*, 41(2):385–390.
- Lea Fischbach. 2024. A Comparative Analysis of Speaker Diarization Models: Creating a Dataset for German Dialectal Speech. In *Proceedings of the 3rd Workshop on NLP Applications to Field Linguistics (Field Matters 2024)*, pages 43–51.
- Matthias Hahn and Beat Siebenhaar. 2016. Sprechtempo und Reduktion im Deutschen (SpuRD). *Studentexte zur Sprachkommunikation: Elektronische Sprachsignalverarbeitung 2016*, pages 198–205.
- Yannick Jadoul, Bill Thompson, and Bart de Boer. 2018. Introducing Parselmouth: A Python interface to Praat. *Journal of Phonetics*, 71:1–15.
- Sofoklis Kakouros, Katri Hiovain, Martti Vainio, and Juraj Šimko. 2020. Dialect identification of spoken North Sámi language varieties using prosodic features. In *Speech Prosody 2020*, pages 625–629.
- Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.
- Khaled Lounnas, Mohamed Lichouri, Mourad Abbas, Thissas Chahboub, and Samir Salmi. 2022. Towards an Automatic Dialect Identification System using Algerian Youtube Videos. In *Proceedings of the 5th International Conference on Natural Language and Speech Processing (ICNLSP 2022)*, pages 258–264.
- H. B. Mann and D. R. Whitney. 1947. On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, 18(1):50 – 60.
- Christina Otto. 2012. Sprechgeschwindigkeit und Geschlechterunterschiede. *Phonetik & Phonologie 8 Friedrich-Schiller-Universität Jena*, 92(1):33.
- Thilo Pfau, Robert Faltlhauser, and Günther Ruske. 2000. A combination of speaker normalization and speech rate normalization for automatic speech recognition. *Proc. 6th International Conference on Spoken Language Processing (ICSLP 2000)*, 4:362–365.
- Vineel Pratap, Andros Tjandra, Bowen Shi, Paden Tomasello, Arun Babu, Sayani Kundu, Ali Elkahky, Zhaoheng Ni, Apoorv Vyas, Maryam Fazel-Zarandi, et al. 2024. Scaling speech technology to 1,000+ languages. *Journal of Machine Learning Research*, 25(97):1–52.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning, ICML’23. JMLR.org*.
- Tim Sainburg. 2019. timsainb/noisereduce: v3.0.0. retrieved 12 February 2021.

- Tim Sainburg, Marvin Thielk, and Timothy Q Genter. 2020. Finding, visualizing, and quantifying latent structure across diverse animal vocal repertoires. *PLoS computational biology*, 16(10).
- Jürgen Erich Schmidt, Joachim Herrgen, Roland Kehrein, and Alfred Lameli. 2020ff. Regionalsprache.de. Forschungsplattform zu den modernen Regionalsprachen des Deutschen. Forschungszentrum Deutscher Sprachatlas Marburg.
- Joel Shor and Subhashini Venugopalan. 2022. TRILLsson: Distilled Universal Paralinguistic Speech Representations. In *Proc. Interspeech 2022*, pages 356–360.
- Alper Kursat Uysal and Serkan Gunal. 2014. The impact of preprocessing on text classification. *Information Processing & Management*, 50(1):104–112.
- Chad Vicens and Megha Sundara. 2013. The role of intonation in language and dialect discrimination by adults. *Journal of Phonetics*, 41(5):297–306.
- Peter Wiesinger. 1983. Die Einteilung der deutschen Dialekte. In Werner Besch, editor, *Dialektologie: Ein Handbuch zur deutschen und allgemeinen Dialektforschung*, volume 1.2 of *Handbücher zur Sprach- und Kommunikationswissenschaft*, pages 807–900. Berlin/New York: de Gruyter, Berlin, New York.

Language of the Swedish Manosphere with Swedish FrameNet

Emilie Francis
Språkbanken Text
University of Gothenburg
emilie.francis@gu.se

Abstract

The manosphere is a loose group of online communities centralised around the themes of anti-feminism, misogyny, racism, and hetero-masculinity. It has gained a reputation for violent extremism, particularly from members of the involuntary celibate (incel) community. Sweden sees one of the highest volumes of online traffic to well-known incel forums in all of Europe. In spite of this, there is little information on manosphere/incel culture in Swedish. This paper uses posts from Flashback’s manosphere subforum automatically annotated with Swedish FrameNet to analyse the language community in a Swedish context. To do so, a lexicon for the Swedish manosphere was created and terms of interest were identified in the Swedish discourse. Analysis of prominent semantic frames linked to these terms of interest presents a detailed look into the language of the Swedish manosphere.

1 Introduction

The ‘manosphere’ is a collection of online communities, including involuntary celibates (incels), Men’s Rights Activists (MRAs), pick-up artists (PUAs), and Men Going their Own Way (MGTOW) (Ging, 2017; Cottee, 2020; Schmitz et al., 2016). Such communities function as online spaces for men to discuss topics related to feminism, masculinity, and relationships through a lens of misogyny and racism (Gajo et al., 2023). Involuntary celibates or ‘incels’ are defined as men who experience frustration at their inability to find a romantic/sexual partner despite desiring one and express this frustration by blaming and denigrating others. Despite several acts of mass violence performed by self-identified incels, the community

itself maintains that the link between incelism and violence is a result of media pigeonholing.¹ Concern over the risk of violence perpetrated by members of the manosphere, particularly by incels, continues to deepen among the public and law enforcement (Matza, 2023; Baele et al., 2024).

According to the Swedish Defence Research Institute (FOI), Sweden ranks among the top countries for traffic to the largest incel forums (Fernquist et al., 2020; European Commission, 2021). It is estimated that Sweden sees 240 visits per million residents compared to the U.S.A.’s 43 per million (Stenavi and Bengtson, 2020). In incel discourse, Sweden is described as the most ‘cucked’² country due to its perceived feminist influence (Wiklund, 2020; Fernquist et al., 2020). The seemingly sudden increase of traffic to incel communities from Sweden has caused both scholars and security agencies to monitor the situation closely.

Although Swedish incels are acknowledged as a substantial demographic in the manosphere, relatively little is known about them. Furthermore, the Swedish manosphere has been overlooked in NLP thus far. This paper aims to bridge the gap between the qualitative research on the Swedish manosphere and NLP by investigating the language of the manosphere through semantic frames automatically annotated with Swedish FrameNet (SweFN). Frame Net is a set of labels developed by Baker et al. (1998) based on the theory of semantic frames by Fillmore (1985) (§ 4.2). The findings of this paper are intended help expand the current SweFN annotations with semantic frames for societal issues related to the manosphere in Sweden. Additionally, it contributes a non-Anglocentric analysis to the body of

¹“Are incels violent?” via Incel Wiki FAQ: https://incels.wiki/w/Incelism_FAQ

²An insult implying humiliation of men by women. Derived from ‘cuckold’ - one’s partner having consensual sex with other men.

research on discourse in the manosphere. In doing so, it yields the following contributions:

- Creation of a comprehensive lexicon for the Swedish manosphere
- Identification of terms of interest and their prominent frames
- An analysis of language in the Swedish manosphere through semantic frames

2 Background

Psycho-social aspects of the manosphere, inceldom, misogyny, and sexual violence have been the subject of numerous studies. Henley et al. (1995) used semantics of the passive voice to study newspaper articles featuring sexual violence. Results indicated that the passive voice was used to obscure agency in reports of sexual violence against women. Minnema et al. (2022b) proposed a framework for studying responsibility framing in reports of violence against women in Italian news. It was observed that roles corresponding to victims were expressed much more frequently than those corresponding to agents. Schmitz et al. (2016) studied twelve MRA websites from the perspective of hegemonic masculinity. Two distinctive groups, both featuring anti-feminist discourse, were revealed. One group promoted the ideology of men's rights through demonization of feminism and aggression towards women, while the other took a political approach and focused on providing evidence of anti-male institutional prejudice and discrimination.

In a study featuring interviews from former incels, participants reported that feelings of isolation, low self-worth, and romantic frustration led them seeking support online (Maryn et al., 2024). Many of these men indicated that feelings of isolation and low self-worth dependent on appearance and success were caused by masculine norms. Maxwell et al. (2020) studied comments on Reddit's *r/Braincels* subforum and identified several themes of which the concept of social isolation was present. In Sweden, many men also claim to suffer from loneliness, especially those living in rural communities (Lindström, 2024; Novak et al., 2023).

Discourse analysis has been a popular method for studying language in the manosphere. Ging (2017) used Critical Discourse Analysis (CDA) to

highlight ideological tropes in a selection of content from frequently cross-referenced sites with a baseline of anti-feminism. Focus on the concept of evolutionary biology gave rise to a deeply misogynist, heterosexist, and racist lexicon. The community itself was also found to have moved away from activism toward personal attacks on feminists. A study of 700 posts from five of the top incel forums found several topics related to the themes of 'incel-culture' and 'incel-identity' (Axelsson and Lindgren, 2021). Discussion of who an incel is and what inceldom entails were prevalent topics. Along with this came discussions of appearance, self-worth, and race. While language was described as negative, encouragement of violence was uncommon.

The link between inceldom and violent extremism has also featured heavily in the body of research. In a study of incel discourse on the Plymouth shooting incident,³ Lounela and Murphy (2024) examined several threads on English incel forums and observed competing discourses around incel violence. While some attempted to justify the Plymouth attack, others condemned all violence. Baele et al. (2021) investigated the incel world-view on the now defunct *Incels.me* forum and found that violent events elicited hope that they may lead to society's recognition of the alleged excesses of feminism. Members engaged in fantasising graphic scenarios centred on the suffering of women and encouraged each other to do so as well. In another study of incel subforums on Reddit, Baele et al. (2024) found that online discussion in the Reddit incelosphere presented an increasing proportion of dehumanising labels and words depicting violence.

In a linguistic analysis of the *Incels.me* forum, Jaki et al. (2019) identified common keywords. The top 100 keywords largely referenced gender and physical traits, while the top 1000 also contained references to sexuality, violence, and hate speech. Yoder et al. (2023) used text analysis to investigate identity construction on the forum *Incels.is* by comparing identity mentions with the white supremacist forum Stormfront. For this, an English lexicon was created by combining multiple sources then expanded using nearest neighbours in the word embedding space. The most frequent identity mentions are for women and minor-

³A mass shooting in Plymouth (UK) where the perpetrator held incel views.

ity/racialised identities, including many derogatory neologisms.

While previous research on the manosphere has been viewed largely through an Anglophone context, several studies have investigated the Italian incel community (Gajo et al., 2023; Gemelli and Minnema, 2024; Minnema et al., 2022a). Gemelli and Minnema (2024) established a dataset labelled with semantic frames for the Italian incel community. The highest-ranked and most informative frames were identified for each subcorpus. To detect hate speech in incel communities, Gajo et al. (2023) used masked language modelling (MLM) with BERT and mBERT adapted to the English and Italian incel domain.

Research on the manosphere in Sweden has largely focused on English forums, even when studying Swedish incels (Fernquist et al., 2020; Axelsson and Lindgren, 2021). In the Swedish context, Lindmark and Kindblom (2021) analysed threads from the Swedish forum Flashback containing the word ‘incel’ or ‘incels’ with techniques from CDA. Users discussed the topic of immigration in relation to a surplus of men in Sweden and how this negatively impacts the white Swedish man (Lindmark and Kindblom, 2021). Swedish incels also commonly express that they feel less desirable than men of colour (Lindström, 2024). Swedish women are described as privileged and positioned as the opposition in a power struggle with men (Lindmark and Kindblom, 2021; Wiklund, 2020; Fernquist et al., 2020). Through discourse analysis and personal communication with Stefan Krakowski, a media expert and scholar of Swedish incels, Wiklund (2020) analysed discourse surrounding incels on Flashback. Search terms, including ‘incel’ and ‘misogyny’, were used to identify threads for analysis. Wiklund (2020) observed that feminism is used to legitimise dehumanisation of women and increased immigration is portrayed as societal problem. The Flashback manosphere was also found to be incredibly sensitive to its portrayal in the media, directing personal attacks at female writers on the Swedish manosphere.

The following sections will outline the methods used to identify terms of interest in the Swedish manosphere and their prominent frames, the means through which they were analysed, and the data used in the analysis.

3 Data

The analysis in this paper focuses on the *Manosfärer, Maskulinism, och Mansrörelser* “Manospheres, Masculinism, and Men’s movements” subforum on Flashback. The hobby subforum was also collected for comparison, as it was the closest in size of the other Lifestyle subforums. Flashback data is collected and annotated in yearly updates which are stored in xml format. The version used in this analysis was collected in March 2024 and contains all threads created between October 2012 and March 2024 (Språkbanken Text, 2024a). Threads have been scrambled to preserve user privacy and copyright. Additionally, usernames have been removed. A total of 12,943 posts were from the manosphere subforum and another 16,565 from the hobby subforum were used in the analysis. Each thread has been automatically annotated with semantic frames from SweFN with the Sparv pipeline (Språkbanken Text, 2024b; Hammarstedt et al., 2022). Only the posts from the manosphere subforum are used in the discussions in §5.

4 Methods

4.1 Lexicon

The Swedish manosphere lexicon was created in two steps. In the first step, a base wordlist of English words from the manosphere was used. As there was no existing lexicon for the Swedish manosphere, a generic lexicon was created following Yoder et al. (2023). A combination of several English sources were utilised for a comprehensive base lexicon (Moonshot, 2020; Klein and Golbeck, 2024; Fernquist et al., 2020). This helped identify words within the Swedish manosphere borrowed directly from English without translation, as well as Swedish-English compounds. Any terms from the base list with zero instances in the data were discarded from the Swedish lexicon.

The second step expanded on the results of the first step by identifying native Swedish or Swedish translations of manosphere terminology. Sentences were lemmatized and stopwords removed to calculate word frequencies. To capture words unique to the Swedish manosphere, the binary log ratio of relative frequencies was calculated for the manosphere and hobby forums. Binary log ratio of relative frequencies, or ‘log ratio’, is a means of comparing the relative frequencies of

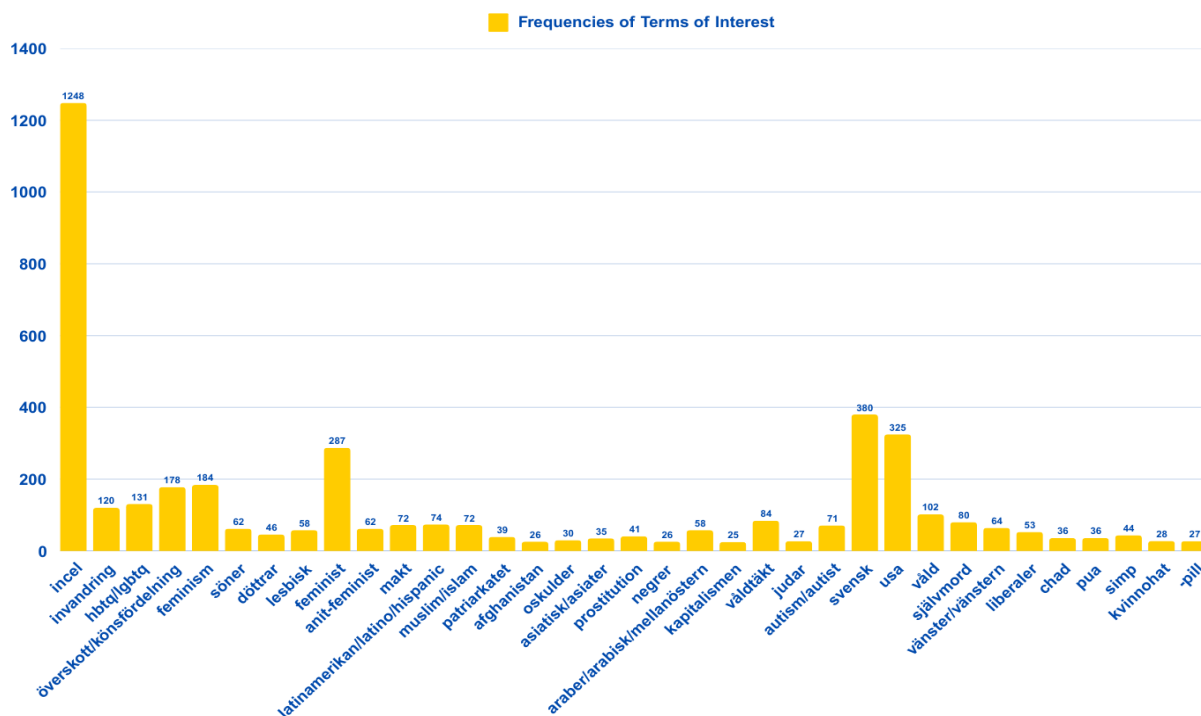


Figure 1: Relative frequencies of the Swedish terms of interest. Words which are direct synonyms or refer to similar concepts have been grouped together in the analysis. Additionally, words with a frequency lower than 25 have been excluded from the graph for space.

words in two corpora (Klein and Golbeck, 2024; Hardie, 2014). It represents how big the difference in frequency is for a keyword between two corpora, where the score is represented as powers of 2 and each point is a doubling in size of the difference between the two corpora. Both the Swedish and combined English wordlists, along with the Swedish lexicon, are available on Github.

Both words from the lexicon and words not specific to the manosphere, but overrepresented in the manosphere forum, were identified as terms of interest. The terms of interest were further narrowed down based on topics presented in the literature from §2 which are considered important to the manosphere in general and in Sweden. These terms of interest were grouped into five themes, discussed further in §4.2 and 5. Figure (1) shows the terms of interest by frequency of mentions.

This is a warning for sections 4.2 to 5.5, as examples contain offensive language and references to violence, sexual abuse, and self-harm.

4.2 Semantic Frames

The analysis in §5 of this paper relies on the concept of semantic frames originally proposed by Fillmore (1985). Semantic frames, indexed in

lexical databases such as FrameNet (Baker et al., 1998), are labels which provide conceptual information grounded in human understanding. Looking at frames allows one to gain insight on how members of a language community understand the world around them. One can determine not only what is said, but how it is understood and what is required for that understanding by interlocutors. Using an example from the data, in the sentence “*women are therefore at high risk of being murdered by their husbands*”, the lexical unit *murdered* triggers the KILLING frame and *women* which triggers the PEOPLE frame fills the ‘victim’ role while *husbands* with the KINSHIP frame would receive the ‘killer’ role.⁴ From repeated instances of sentences expressing KINSHIP in the killer role, it may be inferred that the discourse sees women as victims of domestic homicide.

Similar to Gemelli and Minnema (2024) and Minnema et al. (2022a), the most prominent frames were identified for each term of interest and the context in which they were triggered was analysed. Frames frequently triggered by words which commonly serve a grammatical function,

⁴Capitalised words represent frames used in FrameNet

such as EXISTENCE tagged with ‘*be*’ verbs, were discarded as they add little to the analysis. Prominent frames were found by calculating *ff-icf*.⁵ This is done with a modification of *tf-idf* to compare a frame’s relative frequency in a subcorpus to the larger corpus, giving a score between 0 and 1 (Grootendorst, 2022; Remijnse et al., 2021). The highest ranked frames are considered most representative of the theme.

As only the semantic frame is annotated in the corpus, roles and interactions with other frames in the same sentence were examined manually. ‘Interactions’ are identified as frames which had a higher frequency of co-occurrence with the prominent frame relative to other frames. Thus, frequencies of words which trigger the prominent frame, other frames which occur in the same context, the words which fill its core roles, and the semantic frames of those roles (if available) were calculated for each prominent frame. As it is not feasible to discuss all 63 terms of interest within the scope of this paper, the discussion will focus on the two or three most important words to the theme.

For each theme, two types of results are presented: (1) an identification of frames most representative of the terms of interest for that theme; (2) an analysis of what words are associated with the common frames and what roles they fill in the discourse.

5 Results and Discussion

The prominent frames identified for each term are used to connect related concepts to FEs. Analysing the posts containing these related frames, the words they are triggered by, and the words around them allows one to gain insight into what entities fill which roles.

Frames were ranked from lowest to highest based on their *ff-icf* score with the highest two (or more, if scores were equal) scores being most typical. In the following sections, numbers in brackets indicate the relative frequency of the entity which either triggered the frame or fills one of the frame’s roles in the context of the specific term of interest. As an example, in §5.1, of the instances of RECEIVING for the term ‘bluepill’, 50% had a theme of sympathy or hope. Similarly, in §5.4, of the instances of the frame ORIGIN, 36% were the word ‘Swedish’.

The terms of interest introduced by Figure (1)

⁵frame frequency - inverse corpus frequency

can be categorised by five themes: *inceldom and mental health, feminism and LGBTQ+, race and origin, immigration and male surplus, power and violence*. These themes have been chosen based on the close interaction between these words within each category, such as the claimed ‘cause and effect’ relationship between immigration and ‘male surplus’ discussed in §5.4. All examples have been translated to English from the original Swedish.

5.1 Inceldom and Mental Health

Incel discourse uses the analogy of *pills* to talk about different world-views. *Pill theory* is coined from the movie *The Matrix*, where the protagonist is given the option to accept reality by swallowing a red pill or return to an illusion with the blue pill. One prominent frame associated with these terms is RECEIVING. Recipients in the context of the RECEIVING frame for the *bluepill* are typically men or *man* ‘one’, while the entity that is received tends to be sympathy or hope (0.5).

For the *blackpill*, a fatalistic philosophy focused on external solutions and physical appearance, the frames CAUSATION, EMPLOYED, and COMING TO BELIEVE were most prominent. With CAUSATION, the most common word was ‘therefore’ (0.5) with the effect being blackpill believers ‘losing hope’ or ‘giving up’. The EMPLOYED frame was always triggered by *jobba för* ‘work for’ when users discuss ‘working toward’ something or having ‘nothing to work for’ (0.6). For COMING TO BELIEVE, the content role tends to be realising blackpill theory (0.75). It also commonly appeared with the EMPLOYED frame, where the content of the realisation is filled by ‘nothing to work for’ (0.4).

For *Chads*,⁶ the prominent frames were PEOPLE and REQUIRED EVENT. The PEOPLE frame was triggered by ‘women’ or ‘girls’ (0.51), often with POSSESSION or PERSONAL RELATIONSHIP frame to state having ‘women’, ‘partners’, or ‘friends’ (0.35). The REQUIRED EVENT frame was always triggered by *behöva* ‘need’, such as when contrasting what incels and Chads do or do not need to do within the required situation role (0.71). In the example below, ‘an incel’ and ‘become friends...’ fill the required situation role of the first instance of ‘need to’, while

⁶A ‘Chad’ is defined as the archetypal white male who contrasts with incels in both physicality and access to sex.

‘a Chad/Normie’ and ‘do any of this’ fill the same role for the second. As noted by Maxwell et al. (2020), incels believe there are different rules for Chads.

- (1) “Why should an incel need to become friends with everything and everyone, or go to the gym or go to dance classes when a Chad/Normie doesn’t need to do any of this?”

Autism is a common theme in manosphere discourse, especially among incels given that autism is significantly more common in the incelosphere compared to the rest of the population (Lindström, 2024; Whittaker et al., 2024; Moskalenko et al., 2022). The frames of RECEIVING, LOCATING, PEOPLE BY DISEASE, and KILLING were prominent in relation to this term. *autist/autister* “autistic people” always triggered the PEOPLE BY DISEASE frame. When RECEIVING is triggered, the entity that is received is typically sex, women, or personal relationships where the recipient is ‘an autist’ or ‘incels’ (0.5). The LOCATING frame is triggered in similar contexts, where the perceiver role is autistic men or incels and the sought entity is women or a personal relationship (0.43). KILLING is often triggered by ‘suicide’ (0.43), with the victim role filled by men (0.67). When triggered by verbs like ‘murder’, women are the victims and men killers (0.57).

As *incels* are the most frequently discussed topic, many prominent frames emerged. Of these, COLOR, PEOPLE BY ORIGIN, KILLING, VIOLENCE, and CAUSE MOTION were the most notable. COLOR is often triggered as a descriptor of the PEOPLE frame to describe race, such as ‘black men’ (0.33), ‘black women’ (0.32) and ‘white men’ (0.16). When frames related to violence are triggered, such as *mörda* “murder” or *slå* “beat”, the perpetrator role is filled by ‘incels’ or ‘black men’ with the COLOR frame (0.31). The COLOR and PEOPLE BY ORIGIN frames often appear when debating which race is most incel (0.83). The ORIGIN frame was often triggered by *Svensk* “Swedish”, *Kinesisk* “Chinese”, or *Amerikansk* “American” as a descriptor of ‘men’ or ‘women’ when comparing incels of different demographics and discussing which group is preferred by Swedish women (0.54). This observation is consistent with Lindström (2024). As noted by Axelson and Lindgren (2021), debate over who or what

an incel is constitutes a large part of the dialogue in the manosphere. In Ex.(2), “black” and “white” triggered the COLOR frame, while “Asian” triggered PEOPLE BY ORIGIN in Ex.(3).

- (2) “Black men are more incel than white.”
- (3) “Asian men are the most incel.”

CAUSE MOTION was often triggered by *dra* “draw” when bringing up arguments and “drawing conclusions” (0.43).

5.2 Feminism and LGBTQ+

Feminism and *feminists* on Flashback are often discussed along with *anti-feminsim*. The TEXT and TEXT CREATION frames were triggered often by *bok* “book” or *böcker* “books” (0.3), typically in reference to reading and writing feminist and anti-feminist literature. The POINT OF DISPUTE frame is usually attached to ‘problem’, where the context is problems related to feminism and feminist countries/society (0.59). The dispute frame also appears with the MORALITY EVALUATION frame triggered by ‘evil’ and ‘virtue’, where feminists are the ones evaluated in the evaluatee role (0.54). Again, the ORIGIN frame is often triggered as a descriptor for women or feminists in the form of *Svenska* “Swedish” or *västerländska* “western” (0.52). POSTURE was triggered by *stå* “stand” in discussions on what feminists “stand for” along with ARCHITECTURAL PART frames in the location role (0.33), as in Ex.(4). These appeared comments attributed to anti-feminists on stereotypical gender roles that ‘women belong in the kitchen’, an observation also noted by Ging (2017); Wiklund (2020).

- (4) “Anti-feminist men, with their rumination, want to achieve that women should stay home, stand in the kitchen and take care of the children.”

HBTQ “LGBTQ”, particularly *lesbians*, are frequently featured in Swedish manosphere discourse. LGBTQ often appears with the INTOXICANTS frame. LGBTQ persons are often users in sentences where INTOXICANTS is triggered by *droger* “drugs” (0.5). The KILLING and DEATH frames also came up frequently in association with queer people, but the protagonist role of

the DEATH frame is typically men and the cause of KILLING is drugs or suicide (0.6). A common claim was that the ‘increase’ in queer people has contributed to male drug related deaths. For *lesbisk* “lesbian”, the KINSHIP frame was often evoked with PEOPLE BY AGE (0.39), triggered by *pojkar* “boys” (0.15), *flickor* “girls” (0.13), or *döttrar* “daughters” (0.11). Lesbians typically fill the parental role within the KINSHIP frame. These two frames usually appear with GETTING and REQUIRED EVENT as users discuss the children of LGBTQ parents. In Ex.(5), ‘daughters’ triggered the KINSHIP frame with “lesbians” in the parental role, along with GETTING triggered by ‘get’.

- (5) “I think lesbians should get daughters, mainly because one can assume boys will not see much purpose in life in a lesbian family.”

5.3 Race and Origin

Immigration from Africa and the Middle East makes up a significant portion of newcomers to Sweden in recent years (Eurostat Statistics, 2016), so it is unsurprising that *Araber* “Arabs” and *Muslim* “Muslims” appear in a large part of the Swedish manosphere’s discourse on race and origin. ORIGIN, PEOPLE BY ORIGIN, and PEOPLE BY RELIGION often appear together as descriptors, typically triggered by *svensk* “Swedish” (0.57) and “Muslim” (0.97) in discussions on the effect of Islam on Western countries and Sweden. Many comments express a fear that Swedish/European people and culture is being replaced by Muslims and Islam (0.57), often with the BECOMING frame in claims that Muslims will ‘become the majority’ or Sweden will ‘become Muslim’. In the following example, PEOPLE BY ORIGIN was triggered by “Swedes”, “Arabs”, and “Afghans”, PEOPLE BY RELIGION was triggered by “Muslims”, and ORIGIN was triggered by “Swedish”. These frames appear with BECOMING triggered by “become”, where Swedes (we) fill the entity role which becomes the final category filled by Muslims (them).

- (6) “When Swedes are good Muslims, we will also avoid getting harassed by Arabs and Afghans as we become one of ”them” and with Swedish Sharia law we can be tough on crime in society.”

COLOR, ORIGIN, and PEOPLE BY RELIGION sometimes appear with POSSESSION, typically in posts where ‘Arab’, ‘Muslim’ and ‘black men’ are in the owner role in posts comparing what Muslim/Arab men have in Swedish society versus Islamic society (0.46). When ORIGIN or COLOR are a descriptor for women, they often denote ‘white’ or ‘Swedish’ women (0.54) who are experiencers of the DESIRING frame with Arab and black men as the objects of desire. ‘Swedish women’ also commonly appears with the SEX frame, where the other participant is ‘black’ or ‘Arab’ expressed with a racial epithet (0.5). As observed by Lindström (2024), the discourse claims that Arab and black men ‘take’ Swedish women because these men are more masculine and sexually promiscuous. When COLOR is used to describe women of colour, as mentioned in §5.1, they are the victims of religious society or violence (0.45).

Hispanics, *Latinos*, and *Latinamerikaner* “Latin Americans”, specifically in the U.S.A., are another frequently mentioned demographic marked by PEOPLE BY ORIGIN (0.77). The frames COLOR and DEATH were the most prominent in relation to this group. COLOR is always used as a descriptor of the PEOPLE frame with *vit* “white” (0.65) and *svart* “black” (0.35) to denote race. COLOR and PEOPLE BY ORIGIN often appear with DEATH and DEAD OR ALIVE frames as users compare life expectancy and living conditions between Latinos and other demographics (0.75). In Ex.(7), ‘whites’ and ‘blacks’ triggered the COLOR frame with DEAD OR ALIVE triggered by ‘live’.

- (7) “Hispanics in the US live longer than Whites and Blacks.”

As the focus is on Latin Americans in the U.S., the REGARD, READING, CAUSE TO PERCEIVE frames usually appear with TEXT triggered by ‘article’ or ‘newspaper’ when users claim to have gained some information about Latin Americans through a source (0.6).

The DEGREE and MEASURABLE ATTRIBUTES frames also appear with COLOR and PEOPLE BY ORIGIN, often triggered by ‘Latin Americans’ (0.77) or ‘Asians’ (0.2). When PEOPLE BY ORIGIN is triggered by *asiatisk/asiater* “Asian/Asians”, they are being compared to other demographics in terms of desirability and incel

status (0.67). In this context, DEGREE and MEASURABLE ATTRIBUTES are triggered by *högre* “higher” and *mest* “most”. Many posts mentioning Asians argue that they are either the ‘most incel’ group or have ‘higher’ popularity (0.27), particularly among black women.

5.4 Immigration and the Male Surplus

Invandring “immigration” is a significant topic in the Swedish manosphere. ORIGIN and COLOR are triggered by ‘Swedish’ (0.36) or ‘white’ (0.28) and act as descriptors for the PEOPLE frame (0.64). These commonly appear in discussion on how ‘Swedish/white’ men and women are affected by immigration (0.5), typically debating whether Swedish women benefit from and support immigration. This observation is consistent with previous research by Lindmark and Kindblom (2021). When ORIGIN is *slaviskt* “Slavic”, it is always a descriptor of ‘women’.

The MEASURABLE ATTRIBUTES and CHANGE POSITION SCALE INCREASE triggered by frames were often evoked in the context of immigration as the item/entity that ‘increases’ or is ‘high’ in statements about the state of immigration in Sweden (0.41). These frames are also used in several posts discussing an increase in criminals and incels as a consequence of immigration (0.32). LAW triggered by *invandringpolitik* “immigration policy” and CHANGE OF LEADERSHIP with TEMPORAL SUBREGION are frequently evoked together in discussions about Swedish politics (0.87). In Ex.(8), ‘last election’ triggered the TEMPORAL SUBREGION and CHANGE OF LEADERSHIP frames, while ‘immigration policy’ triggered the LAW frame in the result role for the election.

- (8) “A minority of voting women in Sweden voted for continued generous/irresponsible immigration policies in the last election..”

As observed by Wiklund (2020), the concepts of immigration and male surplus are linked. *Överskott* “surplus”, more specifically *mansöverskott* “surplus of men”, is another significant topic unique to the Swedish manosphere. The CHANGE POSITION SCALE INCREASE frame is triggered by *öka* “increase” where the attribute being increased is ‘surplus’ or ‘surplus of men’ (0.47). The COLOR frame appears most often when discussing black men (0.47) and women

(0.35). COLOR appears with ‘surplus’ and PEOPLE BY ORIGIN triggered by *amerikaner* “Americans” as users discuss a supposed surplus of men among black Americans (0.23). It also appears as a descriptor for ‘black women’, commonly with frames related to violence such as ABUSING (0.23).

One prevalent narrative is that both Sweden and the U.S.A. are experiencing an increasing surplus of men, a point which also came up in Lindström (2024); Lindmark and Kindblom (2021). Male surplus is also linked to the KILLING and DEATH frames (0.67). KILLING is typically triggered by *själv mord* “suicide” with men in the protagonist role (0.67). With DEATH, men are the protagonists and the cause is usually *droger* “drugs” tagged with INTOXICANT (0.67). Swedish manosphere discourse claims that male suicide and drug overdose are a consequence of male surplus. In Ex.(9), ‘suicide’ triggered the KILLING frame with ‘drugs’ tagged as INTOXICANT in the killer role and ‘men in the U.S.’ being the protagonists.

- (9) “The percentage of men in the U.S. who become mentally destroyed has increased by the same amount, they take their life with drugs or suicide. And all this with the world’s highest surplus of men.”

5.5 Power and Violence

Makt “power” and its ownership is a central issue in Sweden’s manosphere (Lindmark and Kindblom, 2021; Wiklund, 2020; Fernquist et al., 2020). By investigating the combination of the PEOPLE and POSSESSION frames, it is observed that ‘women’ are typically in the owner role with ‘power’ being the possession (0.41). When men are in the owner role (0.29), ‘power’ is usually mentioned along with ‘money’ and ‘women’. In this narrative, power is seen as something women have over others and society, but a requirement for men to ‘get’ women. In the following example, ‘women’ are the owners with POSSESSION triggered by ‘have’ and ‘power to decide ...’ is the possession.

- (10) “I don’t understand why flashbackers think women have the power to decide that we should have more crime and more trans people.”

The final term of interest is *våld* “violence”. The most prominent frames for this term are VIOLENCE triggered by *våld* “violence” and RAPE triggered by *våldtäkt* “rape”. Looking at PEOPLE combined with the aforementioned frames, ‘men’ tend to fill the role of perpetrators of violence and rape (0.27) while ‘women’ are in the victim role (0.36). VIOLENCE and RAPE also commonly appear with ‘immigration’ or ‘surplus’ (0.20) as users link these issues to violence and rapes in society. The VIOLENCE and RAPE frames are also evoked in discussions on violence in Sweden (0.14). In Ex.(11), ‘immigration’ appears with ‘rape’, where ‘women’ and ‘children’ are in the victim role. Although ‘robbery’ and ‘social unrest’ are not tagged with a prominent frame, they also represent violent vocabulary typical of discussions including the term ‘immigration’.

- (11) “Welfare society does not work with mass immigration like this for long - plus social unrest, insecurity for women and children who are raped and robbed.”

Many users draw a connection between an imbalance of men and women to increased violence and rape in society, especially in Sweden. The narrative is that a gender imbalance leads to men feeling devalued and leads to more incels and violence because men are unable to secure a partner, an observation consistent with Maryn et al. (2024) and Lindström (2024).

5.6 Discussion

Overall, the results of this analysis are consistent with previous literature using CDA techniques to study both the English and Swedish manosphere. The topic of incel identity remains a strong feature of manosphere discourse, within which the concepts of race and origin play a significant part (Yoder et al., 2023; Ging, 2017). The Swedish manosphere is also characterised by an opposition to feminist philosophy (Ging, 2017; Schmitz et al., 2016). Users often express their frustration of feminism and feminists through insults (Wiklund, 2020; Ging, 2017). Violence is portrayed largely as something perpetrated by men upon women, or men upon themselves (Jaki et al., 2019; Baele et al., 2021; Minnema et al., 2022b). However, while violence against women is a common topic in the manosphere community on Flashback, there

is little evidence that users actually promote acts of violence.

Other salient issues in the Swedish manosphere include immigration and a surplus of men. Immigration, especially people of African and Middle Eastern origin, is seen as a big contributor to Sweden’s sex imbalance (Lindström, 2024; Lindmark and Kindblom, 2021). While Swedish women are seen as benefitting from immigration, white Swedish men in the manosphere feel their social value is negatively impacted. As Sweden is often described in the English incelsphere as ‘cucked’, it is of no surprise that ownership of power is attributed to women in the Swedish manosphere (Fernquist et al., 2020; Wiklund, 2020).

Some unexpected findings also arose in this analysis. The narrative that queer people, particularly lesbians, are drug users and that an increase of LGBTQ+ people contributes to male suicide has not been mentioned in previous research. The argument that lesbian women should have daughters, for which different justifications are given, is also one that seems unique to Flashback’s manosphere discourse. Furthermore, users also appear to discuss and compare how immigration and racial demographics influence men’s value in society in the U.S.A. and Sweden.

6 Conclusion

In this paper, the language of the Swedish manosphere was investigated through the lens of semantic frames. As there was no list of terms specific to Swedish, relative frequencies were used with log ratio to develop a lexicon for the Swedish manosphere. Words with high frequency and/or high log ratio were used to identify terms of interest, which were further narrowed down based on topics observed in previous research. These terms were separated into five general categories with interrelated and overlapping contexts.

By analysing the terms of interest and their prominent frames, it was possible to determine which narratives are shared by the Swedish and English manospheres and which are unique to Sweden. Additionally, it links Swedish incel terms with frames and roles which enables the expansion of existing frame annotation with SweFen. Mapping words from the manosphere which currently lack FrameNet annotations to frames and roles can be used to build upon the current semantic frame schema for Swedish frames for societal issues.

7 Limitations and Future Work

The main limitation of this study was the incomplete annotations of the data. This meant that some frames which appeared frequently were often not directly adjacent to or situated close to the term under investigation. This was largely overcome by manually review frames in context.

Another limitation of this research is that data is limited to one forum. As a consequence, the results of this analysis may be biased toward the language of Flashback users in regards to the manosphere. Unfortunately, there is currently no other source of discourse on the manosphere in Swedish. As Swedish speakers are typically also competent English speakers, it is likely that many serious participants of the manosphere in Sweden gravitate toward established online communities operating in English. Thus, a need to create a dedicated space to discuss the manosphere in Sweden outside of established manosphere communities is low.

In the future, it will be helpful to apply dependency parsing to automatically associate frames with their roles. In addition, it will be necessary to create a semantic frame annotation schema based on the results of this paper that covers discourse in the Swedish manosphere.

References

- Robin M Axelsson and Sandra Persson Lindgren. 2021. The languages of the involuntary celibate : A study of online incel communities. Bachelor's thesis, Mid Sweden University.
- Stephane Baele, Lewys Brace, and Debbie Ging. 2024. A diachronic cross-platforms analysis of violent extremist language in the incel online ecosystem. *Terrorism and Political Violence*, 36:382–405.
- Stephane J. Baele, Lewys Brace, and Travis G. Coan. 2021. From “incel” to “saint”: Analyzing the violent worldview behind the 2018 toronto attack. *Terrorism and Political Violence*, 33:1667–1691.
- Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The berkeley framenet project. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 1:86–90.
- Simon Cottee. 2020. Incel (e)motives: Resentment, shame and revenge. *Studies in Conflict & Terrorism*, 44:93–114.
- European Commission. 2021. Incels: A first scan of the phenomenon (in the eu) and its relevance and challenges for p/cve. Technical report, Directorate General for Migration and Home Affairs.
- Eurostat Statistics. 2016. Swedish migration agency statistics.
- Johan Fernquist, Björn Pelzer, Katie Cohen, Lisa Kaati, and Nazar Akrami. 2020. Hope, cope and rope: Incels i digitala miljöer.
- Charles Fillmore. 1985. Frames and the semantics of understanding. *Quaderni di Semantica*, 6:222–254.
- Paolo Gajo, Arianna Muti, Katerina Korre, Silvia Bernardini, Alberto Barrón-Cedeño, and Cedeño. 2023. On the identification and forecasting of hate speech in inceldom. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 373–384. INCOMA Ltd., Shoumen, Bulgaria.
- Sara Gemelli and Gosse Minnema. 2024. Manospheres: exploring an italian incel community through the lens of nlp and frame semantics. In *Proceedings of the First Workshop on Reference, Framing, and Perspective @ LREC-COLING 2024*, pages 28–39. ELRA and ICCL.
- Debbie Ging. 2017. Alphas, betas, and incels: Theorizing the masculinities of the manosphere. *Men and Masculinities*, 22:638–657.
- Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.
- Martin Hammarstedt, Anne Schumacher, Lars Borin, and Markus Forsberg. 2022. Sparv 5 user manual. Technical report, Göteborg.
- Andrew Hardie. 2014. Statistical identification of keywords, lockwords and collocations as a two-step procedure. In *Proceedings of the ICAME 35 Conference*. CRAL (Centre for Research in Applied Linguistics).
- Nancy M. Henley, Michelle Miller, and Jo Anne Beazley. 1995. Syntax, semantics, and sexual violence. *Journal of Language and Social Psychology*, 14:60–84.
- Sylvia Jaki, Tom De Smedt, Maja Gwózdź, Rudresh Panchal, Alexander Rossa, and Guy De Pauw. 2019. Online hatred of women in the incels.me forum: Linguistic analysis and automatic detection. *Journal of Language Aggression and Conflict*, 7:240–268.
- Emily Klein and Jennifer Golbeck. 2024. A lexicon for studying radicalization in incel communities a lexicon for studying radicalization. In *Proceedings of the 16th ACM Web Science Conference, WebSci 2024*, pages 262–267. Association for Computing Machinery, Inc.
- Maja Lindmark and Frida Kindblom. 2021. Vad är problemet med att inte få ligga? en netnografisk studie om incels på flashback forum. Candidate thesis, University of Gothenburg, 3.

- Emma Lindström. 2024. The local effect of the incelosphere in Värmland: A qualitative interview study on the incelosphere from the perspective of professionals. Master's thesis, Karlstad University.
- Emilia Lounela and Shane Murphy. 2024. Incel violence and victimhood: Negotiating inceldom in on-line discussions of the plymouth shooting. *Terrorism and Political Violence*, 36:344–365.
- Alyssa Maryn, Jordan Keough, Ceilidh McConnell, and Deinera Exner-Cortens. 2024. Identifying pathways to the incel community and where to intervene: A qualitative study with former incels. *Sex Roles*, 90:910–922.
- Max Matza. 2023. Terrorism ruling first for canada 'incel' attack.
- December Maxwell, Sarah R. Robinson, Jessica R. Williams, and Craig Keaton. 2020. "a short story of a lonely guy": A qualitative thematic analysis of involuntary celibacy using reddit. *Sexuality and Culture*, 24:1852–1874.
- Gosse Minnema, Sara Gemelli, Chiara Zanchi, Tommaso Caselli, and Malvina Nissim. 2022a. Sociofillmore: A tool for discovering perspectives. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 240–250.
- Gosse Minnema, Gaetana Ruggiero, Marion Bartl, Sara Gemelli, Tommaso Caselli, Chiara Zanchi, Viviana Patti, Marco Te Brömmelstroet, and Malvina Nissim. 2022b. Responsibility framing under the magnifying lens of nlp: The case of gender-based violence and traffic danger. *Computational Linguistics in the Netherlands Journal*, 12:207–233.
- Moonshot. 2020. Incels: A guide to symbols and terminology. Technical report, Moonshot.
- Sophia Moskalenko, Juncal Fernández-Garayzábal González, Naama Kates, and Jesse Morton. 2022. Incel ideology, radicalization and mental health: A survey study. *The Journal of Intelligence, Conflict, and Warfare*, 4:1–29.
- Masuma Novak, Margda Waern, Lena Johansson, Anna Zettergren, Lina Ryden, Hanna Wetterberg, Therese Rydberg Sterner, Madeleine Mellqvist Fässberg, Pia Gudmundsson, and Ingmar Skoog. 2023. Six-year mortality associated with living alone and loneliness in swedish men and women born in 1930. *BMC Geriatrics*, 23:1–10.
- Levi Remijnse, Marten Postma, and Piek Vossen. 2021. Variation in framing as a function of temporal reporting distance. In *Proceedings of the 14th International Conference on Computational Semantics (IWCS)*, pages 228–238. Association for Computational Linguistics.
- Rachel M Schmitz, Emily Kazyak, Christine M Robinson, and Sue Spivey. 2016. Masculinities in cyberspace: An analysis of portrayals of manhood in men's rights activist websites. *Social Sciences 2016, Vol. 5, Page 18*, 5:18.
- Språkbanken Text. 2024a. Flashback: Livsstil.
- Språkbanken Text. 2024b. Svenskt frasnät (swefn).
- Märta Stenavi and Karin Bengston. 2020. Kvinnohat och våldshyllningar i digitala incelmiljöer. Technical report, Totalförsvarets forskningsinstitut (FOI).
- Joe Whittaker, William Costello, and Andrew G Thomas. 2024. Predicting harm among incels (involuntary celibates): The roles of mental health, ideological belief and social networking. Technical report, UK Government Commission for Countering Extremism (UK Home Office).
- Maria Wiklund. 2020. The misogyny within the manosphere. a discourse analysis in a swedish context. Master's thesis, Malmö University.
- Michael Miller Yoder, Chloe Perry, David West Brown, Kathleen M. Carley, and Meredith Pruden. 2023. Identity construction in a misogynist incels forum. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1–13. Association for Computational Linguistics (ACL).

Hotter and Colder: A New Approach to Annotating Sentiment, Emotions, and Bias in Icelandic Blog Comments

Steinunn Rut Friðriksdóttir

University of Iceland

srf2@hi.is

Dan Saatrup Nielsen

The Alexandra Institute

dan.nielsen@alexandra.dk

Hafsteinn Einarsson

University of Iceland

hafsteinne@hi.is

Abstract

This paper presents Hotter and Colder, a dataset designed to analyze various types of online behavior in Icelandic blog comments. Building on previous work, we used GPT-4o mini to annotate approximately 800,000 comments for 25 tasks, including sentiment analysis, emotion detection, hate speech, and group generalizations. Each comment was automatically labeled on a 5-point Likert scale. In a second annotation stage, comments with high or low probabilities of containing each examined behavior were subjected to manual revision. By leveraging crowdworkers to refine these automatically labeled comments, we ensure the quality and accuracy of our dataset resulting in 12,232 uniquely annotated comments and 19,301 annotations. Hotter and Colder provides an essential resource for advancing research in content moderation and automatically detecting harmful online behaviors in Icelandic. We release both the dataset¹ and annotation interface².

1 Introduction

The rapid growth of online communication platforms has led to an increase in harmful behaviors and, subsequently, an increased need for content moderation (Mathew et al., 2019). Inappropriate comments targeted at specific individuals or groups of people can even go so far as qualifying as hate speech, but more subtle ways of spreading these prejudiced ideas may, for instance, include fear speech, where attempts are made to incite fear about a target community (Saha et al.,

2023). Recent work has focused on detecting these toxic behaviors automatically, thereby lessening the cost and workload for human moderators (see Dehghan and Yanikoglu (2024), Nagar et al. (2023) and Mittal (2023) for instance).

This paper addresses limitations in previous work on sentiment analysis in Icelandic (Friðriksdóttir et al., 2024), using a new methodology to improve class imbalance and low annotator agreement in some tasks. Our approach first uses GPT-4o mini to analyze approximately 800,000 Icelandic blog comments across 25 tasks, including sentiment analysis, emotion detection, hate speech detection, and group generalizations. For most tasks, we employ focused binary annotation, targeting only the extreme cases (highly likely or highly unlikely to exhibit the behavior), rather than using rating scales which have been shown to present challenges in maintaining consistent annotation quality (Kiritchenko and Mohammad, 2017). The exception is sentiment analysis, where we maintain the standard negative, neutral, and positive categories.

This targeted approach allows us to efficiently identify rare but important cases (the proverbial needles-in-a-haystack) such as hate speech comments, which would be resource-intensive to locate through random sampling as used in previous work. To ensure dataset quality, we then employ crowd workers to manually verify the model’s predictions, focusing particularly on comments flagged as highly likely or highly unlikely to contain problematic content. This human verification step is crucial for maintaining accuracy and creating a high-consensus dataset.

Our contributions are as follows:

- We present Hotter and Colder, a dataset of 12,232 Icelandic blog comments annotated for 25 tasks including sentiment, emotions, hate speech, and group generalizations

¹<https://repository.clarin.is/repository/xmlui/handle/20.500.12537/352>

²https://github.com/icelandic-lt/annotation_if_sentiment

- We introduce a two-phase annotation methodology combining GPT-4o mini silver labels with targeted human verification to address class imbalance and improve annotation agreement
- We release both the annotated dataset and annotation platform to support research in content moderation for low-resource languages³

2 Methodology

Our approach combines AI and human efforts in a two-phase annotation process designed to create a high-quality dataset for tasks where the phenomena of interest are often rare. This scarcity poses a significant challenge for dataset creation - random sampling would require extensive human annotation effort to find sufficient positive examples while focusing only on suspected positive cases could bias the dataset. Our methodology aims to balance these concerns by using AI to efficiently identify potential cases across the full spectrum, followed by targeted human verification.

In the first phase (silver labeling), an LLM analyzes a large dataset of comments. For this initial screening, we use GPT-4o mini with a prompt designed for structured output (see Section 2.1). While the model was instructed to consider itself an expert in Icelandic blog analysis to maintain consistent task framing across annotations, we acknowledge this is a common but debatable prompting practice that warrants further investigation. For all tasks except sentiment analysis, the LLM uses a 5-point scale for labeling to capture nuanced assessments.

In the second phase (gold labeling), human annotators review selected comments, focusing primarily on those the LLM rated at the extremes of the scale (1 or 5). This design choice reflects our priority of establishing a foundational dataset with clear, agreed-upon examples of each phenomenon. While this approach may not capture all nuanced edge cases, it serves several important purposes: (1) it enables efficient identification of clear positive examples for rare phenomena, (2) it helps establish reliable baseline annotations for model evaluation, and (3) it aligns with findings that human annotators achieve higher agreement on clear cases (Kiritchenko and Mohammad, 2017). We acknowledge this as a limitation - fu-

ture work should explicitly target borderline cases to improve model robustness.

Human annotators perform binary (yes/no) annotations⁴ for a single task at a time to reduce task switching fatigue. The simplified binary choice for humans, compared to the LLM’s 5-point scale, reflects our focus on identifying clear instances while acknowledging that intermediate cases may require more nuanced future investigation.

This method of using a language model to identify potential candidates for gold labeling builds on established practices. For instance, when compiling their GoEmotions dataset, Demszky et al. (2020) used a BERT-based model to filter out comments that contained high levels of neutrality, leaving the more emotional comments for humans to annotate.

2.1 Silver Labeling Phase

To automate the initial labeling process, we created a prompt for the AI model that instructed the model to perform all of the 25 annotation tasks on a given blog comment in Icelandic⁵. The prompt included a JSON schema that instructed the model on how to label a given comment. The context provided to the model also included the previous comments and the beginning of the blog post on which the comments were posted. We used strictly structured outputs to guarantee that the GPT-4o mini model always labeled each comment for each of the 25 tasks and to make sure that it could only output values that aligned with the Likert scale⁶.

2.2 Data Selection

Following the previous work of Friðriksdóttir et al. (2024), the blog comments used in this work all derive from the Icelandic blog platform `blog.is`. As one of the oldest and still active blogging platforms in Iceland, this website offers a valuable collection of online communication, generating a wide range of debates between people with different perspectives, which is particularly useful for our purposes. However, it should be noted that the gender distribution of the site’s users appears to be quite skewed. `Blog.is` has no obvious demographics accessible for users. In

⁴Hick’s law states that increasing the number of choices will increase the time it takes a person to make a decision logarithmically (Hick, 1952).

⁵<https://gist.github.com/Haffill12/8813b738637fc9a678f524fdf9b5a5d9>

⁶See information on OpenAI’s website here.

³[links redacted]

his master thesis, however, Ásmundsson (2024) used a heuristic approach to determine the gender of the users based on their patronyms (traditionally, women’s last names in Icelandic end with *dóttir* (e. *daughter*) and men’s last names end with *son*). Similarly, we observed that out of 24,193 unique author names, 2,374 ended in “dóttir”, 7,539 ended in “son” and 14,280 user names did not match these endings.

2.3 Task Overview

The LLM was provided with the context of the blog post, previous comments, and the specific comment to be analyzed. The system prompt for the model was “You are an expert at analyzing Icelandic blog comments. Analyze the last comment shown and provide insights based on the given schema.” For a given input, the model generated its analysis according to a predefined JSON schema, ensuring consistency across all evaluated comments.

The analysis began with an overall sentiment classification (positive, negative, or neutral) of each comment. The LLM then evaluated a wide range of attributes, including toxicity, politeness, hate speech, social acceptability in various contexts, emotional content, sarcasm, constructiveness, encouragement, sympathy, trolling behavior, mansplaining, and group generalizations. For hate speech, the model identified specific target groups and aggression levels when present. The analysis of group generalizations included assessing sentiment, factual validity, and whether the mentioned groups were marginalized.

Most attributes were rated on a 5-point Likert scale, where 1 indicated strong disagreement and 5 indicated strong agreement with the presence or intensity of the attribute⁷. For some attributes, such as sentiment (“positive”, “neutral”, “negative”) and gender (“male”, “female”, “non-binary”, “n/a”), predefined categories were used instead.

We selected our emotion categories based on the foundational work of Ekman (1992); Ekman and Heider (1988), who identified seven basic emotions that appear to be universal across cultures: fear, happiness, sadness, surprise, disgust, anger, and contempt. To this set, we added indignation as it represents a distinct social emotion

particularly relevant to online discourse and content moderation. Social acceptability was assessed across various contexts, including conversations with strangers, acquaintances, and close friends, in educational settings with different age groups, and in parliamentary speeches.

The LLM also inferred the author’s gender and we further performed a majority vote over all annotations of a given username to assign a gender to the author’s name. We note that gender inference in online spaces presents significant challenges. While traditional Icelandic naming conventions can provide gender cues through patronymic suffixes (-son/-dóttir), we acknowledge several important limitations in our approach to gender inference:

1. Users may choose pseudonyms that do not reflect their actual gender, particularly given documented patterns of gender-based harassment online.
2. The relationship between usernames and actual gender identity is complex and cannot be reliably determined through automated analysis.
3. Some users may intentionally obscure their gender or choose gender-neutral identifiers.

We emphasize that the inferred gender labels should be treated as approximations of perceived rather than actual gender, particularly in analyses of gendered interaction patterns like mansplaining. Future work should explore alternative approaches to studying gendered communication patterns that do not rely on automated gender inference.

2.4 Human Annotation Process

To evaluate Icelandic blog comments, we developed a comprehensive annotation scheme covering various aspects of online discourse. Human annotators were provided with detailed instructions in Icelandic, emphasizing that their personal judgment was crucial and that there were no strictly right or wrong answers. Annotators were instructed to base their decisions on the content of the comments rather than the authors’ names, of which only initials and inferred gender were provided.

For most tasks, annotators were asked to make binary decisions (yes/no) about whether a com-

⁷Rubric: 1 - Strongly Disagree, 2 - Disagree, 3 - Neither Agree nor Disagree, 4 - Agree, 5 - Strongly Agree

ment exhibited specific characteristics. The exception was sentiment analysis, which used a three-way classification. Annotators could view preceding comments and the original blog post for context, although some images were no longer available. They were also given the option to skip annotation for comments containing minimal information or those in languages other than Icelandic.

2.4.1 Sentiment Analysis

Following the approach of Wankhade et al. (2022), we conducted sentiment analysis at the comment level. Annotators classified each comment as positive, negative, or neutral based on their personal interpretation. Positive sentiment was defined as expressing approval, happiness, satisfaction, or optimism. Negative sentiment indicated dissatisfaction, criticism, anger, or disappointment. Neutral sentiment was characterized by a lack of strong emotion or a balanced view, often seen in informational or factual statements.

2.4.2 Toxicity

We adopted the definition of toxicity in online discussions from Klein and Majdoubi (2024), describing it as behavior that is rude, disrespectful, or unreasonable, potentially making users feel unwelcome or discouraged from participating in the discussion. Annotators were instructed to identify comments containing insults, aggressive language, or content likely to incite conflict. This approach acknowledges the potential of toxic comments to disrupt constructive dialogue and decrease user engagement, as observed in studies of online forums (Young Reusser et al., 2024).

2.4.3 Hate Speech

Our hate speech annotation scheme was based on Basile et al. (2019) and aligned with Article 233 (a) of the Icelandic penal code, an approach also used by Friðriksdóttir et al. (2024). Annotators identified comments containing threats, defamation, or denigration based on protected characteristics such as nationality, color, race, religion, sexual orientation, disabilities, or gender identity.

2.4.4 Social Acceptance

To gauge social acceptability, annotators evaluated whether it would be appropriate to make the comment in question in various real-life contexts. These included interactions with strangers, acquaintances, and close friends, as well as in educational settings (for both young children and

teenagers) and in parliamentary speeches. This multi-context approach allowed for a nuanced understanding of perceived social norms across different situations.

2.4.5 Emotion Detection

Our emotion detection task was inspired by the work of Friðriksdóttir et al. (2024) and Demszky et al. (2020). We simplified the task by asking annotators to detect the presence of a single emotion at a time in a binary fashion. In other words, to answer whether or not a comment contained the given emotion. The emotions included were based on basic emotions identified by Ekman (1992) and Ekman and Heider (1988): fear, happiness, sadness, surprise, disgust, anger, and contempt. We also included indignation.

2.4.6 Sarcasm

Following the approach of Ptáček et al. (2014), we asked the annotators to label whether a given comment was sarcastic or ironic. In Icelandic, there is a tendency to lump these two meanings together in one (ice. *kaldhæðni*).

2.4.7 Constructiveness

We employed a simplified version of the annotation scheme from Kolhatkar et al. (2020), asking annotators to determine whether comments were constructive. This binary classification focused on identifying comments that provided useful feedback or contributed positively to the discussion.

2.4.8 Encouragement and Sympathy

Inspired by Sosea and Caragea (2022), we asked annotators to identify encouragement and sympathy in comments in a binary fashion. Encouragement was defined as inspirational words or support and sympathy was defined to be compassion, pity, or understanding of the situation of another person.

2.4.9 Additional Annotations

We included several other classification tasks to capture various aspects of online discourse:

Politeness: Annotators assessed whether comments were polite, providing a measure of civility in online interactions.

Trolling: Following the definition used by Friðriksdóttir et al. (2024), we asked annotators to identify comments that were intentionally

provocative, offensive, or off-topic, aimed at eliciting strong emotional responses or disrupting normal discussion.

Mansplaining: The term has been defined by Bridges (2017) as “a man explaining something to a woman in a tone perceived as condescending,” but has since been expanded to cover a broader range of communicative behaviors (Smith et al., 2022). Annotators were instructed to identify instances where comments exhibited unsolicited, patronizing explanations based on the assumption that the recipient is ignorant. Key characteristics of mansplaining include:

- Persistence even when the recipient demonstrates expertise.
- Maintenance of an oversimplified approach.
- Unwarranted confidence, sometimes even when factually incorrect.

While mansplaining can occur between individuals of any gender, annotators were instructed to use the label only for male-to-female interactions. The gendered term highlights the frequency of this dynamic in male-female conversations, particularly in fields where women may have equal or superior expertise. This annotation task aimed to reveal ongoing societal assumptions about gender, knowledge, and competence, illustrating how gender-based power dynamics continue to shape interpersonal and professional communications.

Group Generalizations: Annotators were asked to identify comments containing broad, often oversimplified statements about entire groups of people. These generalizations could be based on characteristics such as race, gender, nationality, or political views. Importantly, annotators were instructed to note that these generalizations could be positive, negative, or neutral in nature. This task aimed to capture instances where comments reflected biases, stereotypes, or assumptions about groups, providing insight into how these generalizations manifest in online discourse.

2.5 Agreement Measures

To evaluate annotation quality and reliability, we employed multiple agreement metrics. For tasks with two or more annotations per comment, we calculated pairwise agreement (PA) as the proportion of agreeing annotation pairs across all possible pairs. For assessing inter-annotator reliability, we utilized Krippendorff’s alpha (K’s α),

which accounts for chance agreement and can handle missing data — a common occurrence in crowdsourced annotations. To evaluate the GPT-4o mini’s performance against human judgments, we computed Cohen’s kappa (C’s κ) between the model’s predictions and the human consensus labels that were computed through a majority vote (examples with ties were dropped). For the sentiment analysis task, which involved three-way classification, we adapted these measures to account for the additional category whilst maintaining the same computational framework.

2.6 Annotation Interface

The annotation interface was designed to facilitate efficient and accurate labeling of blog comments while providing contextual information to annotators. The interface presents one comment at a time, along with metadata such as the author’s initials, inferred gender, and timestamp. To enhance context, annotators can optionally view the full blog post and previous comments in the thread where the same type of metadata is shown for each author. Tasks are presented sequentially, with clear instructions and the option to skip comments when necessary. To maintain engagement and provide feedback, the interface incorporates gamification elements such as progress tracking and achievement badges.

To ensure data quality, the interface implements several key features. First, it allows annotators to review task-specific guidelines at any point during the annotation process. Second, the interface offers an optional real-time feedback mechanism that compares human annotations to predictions from GPT-4o-mini, though annotators are explicitly instructed to rely on their own judgment rather than attempting to match the model’s output. This design balances the need for comprehensive contextual information with the goal of maintaining annotator focus and efficiency throughout the task.

3 Results

3.1 Distribution of AI labels

Before selecting comments for human annotations, we labeled all comments in the 25 different tasks using the GPT-4o mini model. The distribution of labels for each task that was labeled according to a Likert scale is shown in Figure 2 and the distribution of labels in the sentiment task is shown in Figure 3. For sentiment analysis, we ob-

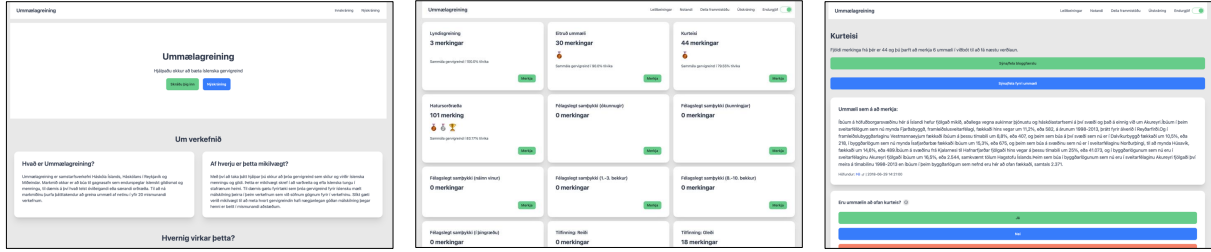


Figure 1: Key components of the annotation platform: (left) The landing page introducing the project and its importance; (middle) The task overview dashboard displaying user progress and available annotation tasks; (right) An example of a specific annotation task (politeness assessment) showing the comment to be annotated, contextual information, and annotation options.

serve a somewhat balanced distribution of labels with over 180,000 labels in each sentiment category. For tasks that were rated on a Likert scale, we see great variability in the label distributions. Some tasks, such as toxicity, social acceptability (teacher to young children in an educational environment, parliament speeches), emotion (anger, contempt, indignation), and constructiveness have a somewhat balanced distribution with a significant number of comments in each label category. Tasks such as politeness and social acceptability (strangers, acquaintances, close friends, teacher to teenagers in an educational environment) are skewed to the right and have few comments rated as not having the property of the task. Other tasks are skewed to the left with few comments having the property. For example, 6,672 comments were labeled as having hate speech with strong agreement. The most problematic tasks were “surprise” and “fear” with only 27 and 668 comments respectively labeled as having the properties with strong agreement.

Our sampling strategy balanced the need for cross-task analysis with the goal of maximizing dataset diversity. We began by creating a shared evaluation set of 100 comments selected uniformly at random from the full corpus. These comments were set as annotation candidates for all 25 tasks, providing a consistent benchmark for analyzing relationships between different aspects of online discourse, such as how toxicity relates to emotion or constructiveness.

For each task, we then selected an additional 1,100 comments that showed strong signals for that specific behavior based on the LLM’s ratings (600 comments rated “5” and 500 rated “1”). To maximize dataset diversity and reduce annotator fatigue, we excluded these task-specific comments

from the selection in other tasks. This decision reflects the distinct nature of our annotation tasks – a comment exhibiting strong hate speech, for instance, might be uninformative for tasks like encouragement or constructiveness. By presenting annotators with fresh content for each task, we aimed to maintain their engagement and avoid potential biases from repeated exposure to the same comments. Additionally, since we focus on extreme cases, reusing comments across tasks could lead to redundancy, as comments rated extreme in one dimension often represent neutral or irrelevant cases for other dimensions.

The resulting dataset of comment candidates⁸ for human evaluation contains 1,200 comments per task (100 shared + 1,100 task-specific). While this design limits comprehensive cross-task analysis to the shared set of 100 comments, it provides rich, focused data for developing robust classifiers for each individual task. Future work could explore the possibility of annotating a larger shared set of comments across all tasks, which would enable more comprehensive analysis of task relationships while potentially sacrificing some task-specific coverage.

3.2 Annotator Statistics

The dataset comprises annotations from 170 unique annotators with an average age of 37.61 years. The educational background of the annotators is diverse, with the majority holding advanced degrees: 36.5% have a master’s degree, 22.9% have a bachelor’s degree, and 5.9% have a PhD. The gender distribution is nearly balanced, with 47.6% male and 49.4% female annotators, while a small percentage identify as other (2.4%) or pre-

⁸Note that not all comments were fully annotated in all task categories.

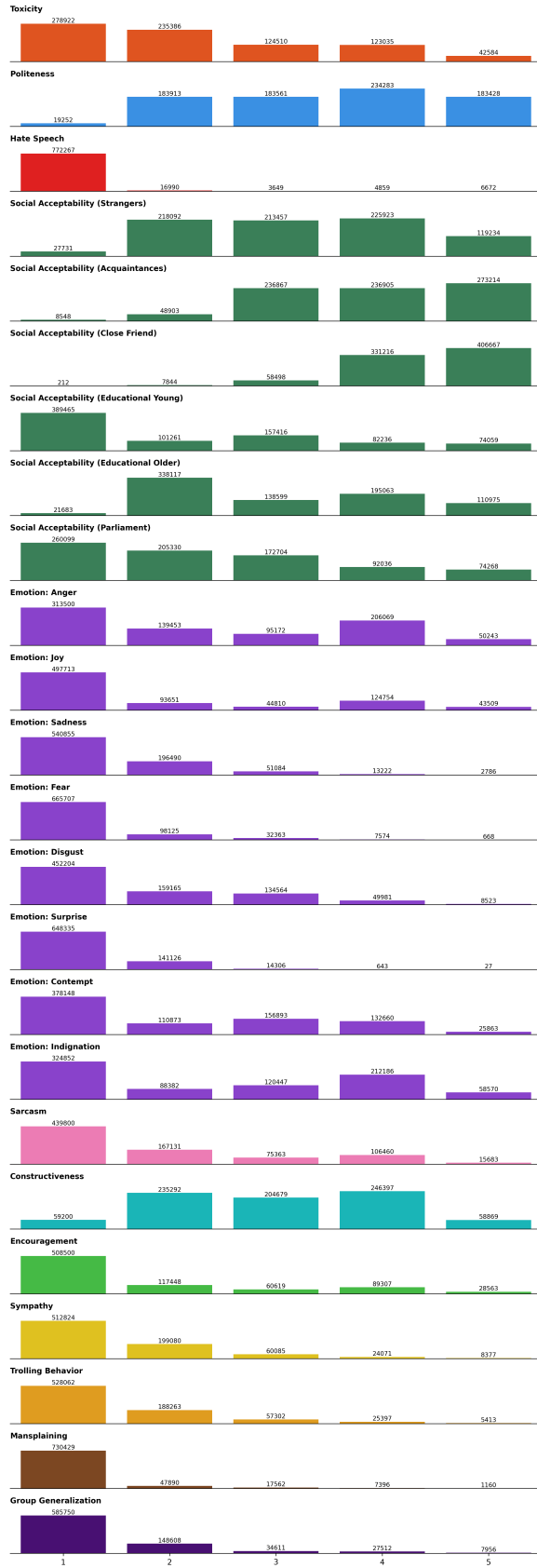


Figure 2: Distribution of AI labels on tasks that were rated from 1 to 5 on a Likert scale.

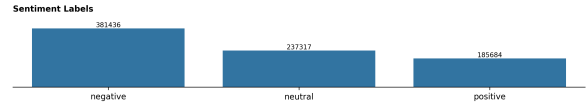


Figure 3: Distribution of AI labels for the sentiment analysis task.

fer not to say (0.6%). In terms of participation, there is a notable disparity between the average and median number of annotations per user (113.7 and 27.5 respectively), suggesting that while some annotators contributed extensively, the typical annotator provided a more modest number of annotations.

The recruitment and motivation of crowdworkers for annotation tasks can be a challenge. Most of our participants were recruited through targeted Facebook groups, with advertisements highlighting the potential societal benefits of training models to detect hate speech and toxic online behavior. This framing likely contributed to the relatively high number of annotations in these categories. However, task participation decreased for tasks presented later in the annotation sequence, leading to an uneven number of annotations across tasks and a potential annotator bias in those that had a lower number of total annotations. This suggests that fatigue or prioritization may have influenced the workers' engagement with certain tasks, particularly those positioned further down the task list. In future work, this issue could be mitigated by randomizing the order in which tasks are presented to each crowd worker, thereby ensuring a more balanced distribution of participation across tasks.

3.3 Agreement

Table 1 presents an overview of the annotation statistics and agreement measures for each task in our study. We report several metrics to provide a comprehensive view of the annotation quality and the performance of our AI model compared to human annotators.

To assess the reliability of the annotations, we calculated Krippendorff's alpha (Krippendorff, 2018, $K's \alpha$) for inter-annotator agreement. The results varied considerably across tasks, with some showing strong agreement (e.g., disgust: 0.92, sympathy: 0.83) and others showing weaker agreement (e.g., mansplaining: 0.07, fear: 0.24). This variability suggests that some concepts were

more challenging to annotate consistently than others. It may be noted that the instructions for mansplaining were more specific for the human annotators than for GPT-4o mini as they explicitly mentioned that the comment should be from a man to a woman. However, that is often an implicit understanding of the word.

To evaluate the performance of our AI model against human consensus, we computed Cohen’s kappa (Cohen, 1960, C’s κ) between the AI predictions and the aggregated human labels. The AI model showed moderate to substantial agreement with human annotators on several tasks, including politeness (0.82), social acceptability in educational settings (0.74), and emotion detection for anger and joy (both 0.68). However, the model struggled with more nuanced tasks such as mansplaining (0.17) and sarcasm detection (0.23).

Interestingly, some tasks exhibited a discrepancy between human inter-annotator agreement and AI-human agreement. For instance, the sympathy task had high human agreement (K’s $\alpha = 0.83$) but low AI-human agreement (C’s $\kappa = 0.24$), suggesting that while humans consistently identified sympathy, the AI model had difficulty capturing this concept accurately. However, it should be noted that while certainly a valid translation for “sympathy”, the Icelandic term “samúð” has a tendency to be linked exclusively to condolences made on the occasion of the death of a person’s relative or friend. It is therefore conceivable that our human annotators have a more narrow understanding of the word than that used by the AI model.

The sentiment analysis task, which involved a three-way classification, showed moderate agreement both among human annotators (K’s $\alpha = 0.64$) and between the AI and human consensus (C’s $\kappa = 0.59$).

The results highlight the varying degrees of difficulty in annotating different aspects of online discourse. While some tasks, particularly those related to basic emotions and clearly defined concepts, showed high agreement, others involving more nuanced or context-dependent judgments proved more challenging for both human annotators and our AI model. Most of the time, if a task has low inter-annotator agreement, the human-AI agreement will also be low, indicating that concepts like sarcasm and trolling are simply difficult to detect in text. It is, however, interesting to note the cases where inter-annotator agreement

is high but human-AI agreement is low. For instance, GPT-4o mini does not seem to have a good grasp of the emotions disgust and surprise.

4 Discussion

The gold standard, human annotated Hotter and Colder dataset is relatively small. While its main purpose is to serve as validation for the AI-labeled silver dataset, it can also be used as training data for few-shot learning models. The silver dataset offers considerable flexibility, supporting the training of models for individual tasks, such as the automated detection of hate speech. However, the utility of both datasets extends beyond single-task applications. Multi-Task Learning (MTL) allows a model to tackle multiple tasks simultaneously, drawing on shared representations and insights across tasks to improve overall performance. In sentiment analysis, for example, an MTL framework enables a more nuanced understanding of human communication. Tan et al. (2023) demonstrate how sarcasm detection can significantly enhance the performance of sentiment analysis models, particularly in identifying negative sentiment in sarcastic contexts. Our results indicate that sarcasm detection remains a challenge, likely contributing to the suboptimal performance of the model in the sentiment analysis task. Given that Icelandic humor often relies on sarcasm, this cultural factor may explain some of the difficulties the model encounters in this task. Consequently, it is plausible that an Icelandic sentiment analysis model would benefit from an MTL approach, particularly one that integrates sarcasm detection as a complementary task.

When working with multilingual LLMs, cultural norms exhibited by the model might not always match those of the country in question (Meadows et al., 2024). Rather, these models reflect the cultural, legal, and ideological values of their creators. Tao et al. (2024) showcased that GPT-4o mini generally mirrors values that are commonly found in English-speaking and Protestant European countries. While this cultural bias may not be inherently problematic for our purposes, it could lead to reduced agreement between human annotators and AI models in culture-specific annotations. For instance, ethical alignment performed during model training may influence the model’s ability to judge appropriateness in social contexts. A model might consistently

Task	Count	$A \geq 2$	AAPC	K's α	C's κ
Emotion disgust	355	32	1.09	0.86	0.53
Social acceptability acquaintances	395	44	1.11	0.77	0.71
Emotion contempt	342	37	1.11	0.77	0.63
Emotion surprise	359	23	1.07	0.75	0.35
Encouragement presence	448	69	1.17	0.74	0.66
Emotion joy	525	127	1.28	0.69	0.69
Emotion sadness	381	49	1.13	0.68	0.50
Emotion anger	547	106	1.22	0.67	0.72
Politeness	749	286	1.49	0.67	0.80
Social acceptability educational young	448	68	1.16	0.65	0.73
Group generalization presence	585	156	1.32	0.64	0.62
Social acceptability strangers	526	129	1.29	0.62	0.76
Hate speech presence	877	429	1.70	0.61	0.60
Sentiment	1099	837	2.64	0.61	0.61
Social acceptability educational older	390	51	1.14	0.58	0.75
Constructiveness	464	71	1.16	0.53	0.53
Sympathy	460	63	1.15	0.53	0.25
Toxicity	981	585	2.01	0.52	0.65
Social acceptability close friend	381	33	1.09	0.44	0.36
Emotion fear	384	48	1.13	0.43	0.60
Social acceptability parliament	404	58	1.15	0.39	0.51
Trolling behavior	511	111	1.25	0.38	0.47
Emotion indignation	354	26	1.08	0.33	0.56
Sarcasm	507	89	1.19	0.29	0.26
Mansplaining	572	136	1.29	0.28	0.21
Average	521.76	146.52	1.30	0.58	0.56

Table 1: Overview of the annotations by task. The count column represents the number of comments annotated for each task. The $A \geq 2$ represents the number of comments with two or more annotations. AAPC represents the average number of non-skipped annotations per comment. K's α corresponds to Krippendorff's α amongst the human annotators in the task. Finally, C's κ refers to Cohen's κ between the AI model and a human consensus label. The last row shows the total for the first two numerical columns and a macro average for the other columns.

classify toxic or hateful comments as unacceptable, even when human annotators might tolerate such comments in specific contexts, such as in private conversations among friends or informal parliamentary discourse. These nuances in cultural and ethical standards may hinder the model's performance in tasks requiring a deep understanding of social norms and context.

On the flip side of the coin, Hotter and Colder additionally offers invaluable insight into the sociolinguistic patterns of a small online community. Future research will i.a. include an analysis of how discourse changes in liaison with current events, which communities are most affected by toxic behaviors and hate speech, and the characteristics of toxic users.

5 Conclusion

This study presents Hotter and Colder, a dataset annotated for 25 tasks that examine various types of online behaviors. By leveraging both AI-based silver labeling and human-in-the-loop gold labeling, we ensure a comprehensive approach to annotating toxic behaviors, emotions, sentiments, and more in Icelandic blog comments. This dual-

phase annotation methodology enabled the identification of rare but critical instances of harmful speech while maintaining high annotator agreement across a variety of tasks.

The introduction of a Multi-Task Learning framework as a future direction holds promise for improving the detection of complex phenomena, such as sarcasm, which remains a challenge for both AI models and human annotators, particularly in culturally specific contexts. By integrating tasks such as sarcasm detection with sentiment analysis, future models may achieve greater accuracy and nuanced understanding in detecting various forms of harmful and toxic speech.

Hotter and Colder lays the foundation for future work on mitigating bias and improving ethical alignment in AI models for Icelandic, hopefully fostering safer and more inclusive online environments.

6 Ethical Considerations

In our efforts to recruit crowd workers, we appealed mostly to their desire to fight against toxic online behavior and to help aid in the eventual creation of an automatic content moderation tool.

Recruiting crowd workers without offering compensation for their work can be considered problematic. We acknowledge that this fact is the likely cause for the relatively unbalanced annotations across tasks. In our case, participants were informed during the recruitment process that a random participant would receive a prize. However, with sufficient financing, it would be more sustainable and fair towards the participants to pay each annotator based on their contributions.

Furthermore, the content in question is inherently problematic in nature. We instructed users to only participate in tasks they were comfortable with and warned them about potential triggers in the content. One user pointed out to us that only being able to label one task at a time for each comment can be unpleasant. For instance, a comment can both have a positive sentiment and exhibit hate speech at the same time. Furthermore, several of the comments will likely be of mixed valence but the annotators were only able to label the comments on either a binary or a 3-class labeling scheme. We acknowledge this limitation.

We also acknowledge that we studied gender from a binary perspective. We decided to go for that approach since non-binary gender identities can be significantly harder to infer based on usernames. We encourage future researchers to be more inclusive in their research.

We acknowledge the significant computational resources and associated carbon footprint involved in using GPT-4o mini to analyze 800,000 comments, especially given the final dataset size of approximately 12,000 annotated comments. While this approach may appear computationally inefficient at first glance, it served a crucial methodological purpose: identifying rare but important cases of problematic content that would have been extremely resource-intensive to locate through random sampling alone. Traditional approaches requiring human annotators to sift through hundreds of thousands of comments to find relatively rare instances of hate speech or other harmful content would have been prohibitively expensive and potentially more damaging to annotator well-being through extended exposure to toxic content. Future work should explore more environmentally sustainable approaches, such as using smaller, task-specific models for initial filtering or developing more efficient sampling strategies that could achieve similar results with less computa-

tional overhead.

7 Ethics Approval

Running this study as a crowdsourcing project was approved by the ethics board of the University of Iceland (SHV2024-080).

8 Acknowledgements

This work was supported by The Ludvig Storr Trust no. LSTORR2023-93030 and The Icelandic Language Technology Programme.

References

- Atli Snær Ásmundsson. 2024. Analyzing Social Behavior in Icelandic Blogging Communities. Sentiment Analysis and Related Tasks with IceBERT. Master’s thesis, University of Iceland.
- Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA. Association for Computational Linguistics.
- Judith Bridges. 2017. Gendering metapragmatics in online discourse: “mansplaining man gonna mansplain...”. *Discourse, Context & Media*, 20:94–102.
- Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.
- Somaiyeh Dehghan and Berrin Yanikoglu. 2024. Evaluating chatgpt’s ability to detect hate speech in turkish tweets. In *Proceedings of the 7th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2024)*, pages 54–59.
- Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. GoEmotions: A dataset of fine-grained emotions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4040–4054, Online. Association for Computational Linguistics.
- Paul Ekman. 1992. An argument for basic emotions. *Cognition & Emotion*, 6(3-4):169–200.
- Paul Ekman and Karl G Heider. 1988. The universality of a contempt expression: A replication. *Motivation and Emotion*, 12(3):303–308.

- Steinunn Rut Friðriksdóttir, Annika Simonsen, Atli Snær Ásmundsson, Guðrún Lilja Friðjónsdóttir, Anton Karl Ingason, Vésteinn Snæbjarnarson, and Hafsteinn Einarsson. 2024. Ice and fire: Dataset on sentiment, emotions, toxicity, sarcasm, hate speech, sympathy and more in Icelandic blog comments. In *Proceedings of the Fourth Workshop on Threat, Aggression & Cyberbullying @ LREC-COLING-2024*, pages 73–84, Torino, Italia. ELRA and ICCL.
- William E Hick. 1952. On the rate of gain of information. *Quarterly Journal of Experimental Psychology*, 4(1):11–26.
- Svetlana Kiritchenko and Saif Mohammad. 2017. Best-worst scaling more reliable than rating scales: A case study on sentiment intensity annotation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 465–470, Vancouver, Canada. Association for Computational Linguistics.
- Mark Klein and Nouhayla Majdoubi. 2024. The medium is the message: toxicity declines in structured vs unstructured online deliberations. *World Wide Web*, 27(3):31.
- Varada Kolhatkar, Nithum Thain, Jeffrey Sorensen, Lucas Dixon, and Maite Taboada. 2020. Classifying constructive comments. *arXiv preprint arXiv:2004.05476*.
- Klaus Krippendorff. 2018. *Content analysis: An introduction to its methodology*. Sage Publications.
- Binny Mathew, Ritam Dutt, Pawan Goyal, and Animesh Mukherjee. 2019. Spread of hate speech in online social media. In *Proceedings of the 10th ACM conference on web science*, pages 173–182.
- Gwenyth Isobel Meadows, Nicholas Wai Long Lau, Eva Adelina Susanto, Chi Lok Yu, and Aditya Paul. 2024. Localvaluebench: A collaboratively built and extensible benchmark for evaluating localized value alignment and ethical safety in large language models. *arXiv preprint arXiv:2408.01460*.
- Utkarsh Mittal. 2023. Detecting hate speech utilizing deep convolutional network and transformer models. In *2023 International Conference on Electrical, Electronics, Communication and Computers (ELEXCOM)*, pages 1–4. IEEE.
- Seema Nagar, Ferdous Ahmed Barbhuiya, and Kuntal Dey. 2023. Towards more robust hate speech detection: using social context and user data. *Social Network Analysis and Mining*, 13(1):47.
- Tomáš Ptáček, Ivan Habernal, and Jun Hong. 2014. Sarcasm detection on Czech and English Twitter. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 213–223, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.
- Punyajoy Saha, Kiran Garimella, Narla Komal Kalyan, Saurabh Kumar Pandey, Pauras Mangesh Meher, Binny Mathew, and Animesh Mukherjee. 2023. On the rise of fear speech in online social media. *Proceedings of the National Academy of Sciences*, 120(11):e2212270120.
- Chelsie J Smith, Linda Schweitzer, Katarina Lauch, and Ashlyn Bird. 2022. ‘Well, actually’: investigating mansplaining in the modern workplace. *Journal of Management & Organization*, pages 1–19.
- Tiberiu Sosea and Cornelia Caragea. 2022. EnsyNet: A dataset for encouragement and sympathy detection. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5444–5449, Marseille, France. European Language Resources Association.
- Yik Yang Tan, Chee-Onn Chow, Jeevan Kanesan, Joon Huang Chuah, and YongLiang Lim. 2023. Sentiment analysis and sarcasm detection using deep multi-task learning. *Wireless Personal Communications*, 129(3):2213–2237.
- Yan Tao, Olga Viberg, Ryan S Baker, and René F Kizilcec. 2024. Cultural bias and cultural alignment of large language models. *PNAS Nexus*, 3(9):pgae346.
- Mayur Wankhade, Annavarapu Chandra Sekhara Rao, and Chaitanya Kulkarni. 2022. A survey on sentiment analysis methods, applications, and challenges. *Artificial Intelligence Review*, 55(7):5731–5780.
- Alison I Young Reusser, Kristian M Veit, Elizabeth A Gassin, and Jonathan P Case. 2024. Responding to online toxicity: Which strategies make others feel freer to contribute, believe that toxicity will decrease, and believe that justice has been restored? *Collabra: Psychology*, 10(1).

Towards large-scale speech foundation models for a low-resource minority language

Yaroslav Getman

Department of Information and
Communications Engineering
Aalto University
Finland

yaroslav.getman@aalto.fi

Tamás Grósz

Department of Information and
Communications Engineering
Aalto University
Finland

tamas.grosz@aalto.fi

Katri Hiovain-Asikainen

UiT The Arctic University of Norway
katri.hiovain-asikainen@uit.no

Tommi Lehtonen

Finnish National Audiovisual Institute
(KAVI)
Finland

tommi.lehtonen@kavi.fi

Mikko Kurimo

Department of Information and
Communications Engineering
Aalto University
Finland

mikko.kurimo@aalto.fi

Abstract

Modern ASR systems require massive amounts of training data. While ASR training data for most languages are scarce and expensive to transcribe, a practical solution is to collect huge amounts of raw untranscribed speech and pre-train the ASR model in a self-supervised manner. Unfortunately, for many low-resource minority languages, even untranscribed speech data are scarce. In this paper, we propose a solution for the Northern Sámi language with 22,400 hours of speech extracted from the Finnish radio and television archives. We evaluated the model performance with different decoding algorithms and examined the models' internal behavior with interpretation-based techniques.

1 Introduction

Self-Supervised Learning (SSL) has caused a paradigm shift in Automatic Speech Recognition (ASR), enabling the development of highly accurate End-to-End models even with a limited amount of data. Low-resource languages also benefited from this advancement, as models pre-trained on other languages proved to be a good

foundation for the development of ASR models using small supervised corpora (Bogdanoski et al., 2023; Gilles et al., 2023). Northern Sámi, a language spoken by only about 20,000 people has also seen rapid advancements in speech technology (Hiovain-Asikainen and De la Rosa, 2023; Getman et al., 2024a).

While fine-tuning speech foundation models such as wav2vec 2.0 (Baevski et al., 2020) can now be considered standard procedure, choosing the right pre-trained system is still very critical. Several works have reported that monolingual pre-training tends to produce the best foundation (Evain et al., 2021; Lehečka et al., 2024; Parcollet et al., 2024), which could be impossible without access to large speech-only corpora. Alternatively, continuing the pre-training of an existing model could adapt it to new languages (Javed et al., 2022). In this work, we build speech foundation models for Northern Sámi with about 22,400 hours of speech from radio broadcasts, which puts them on par with most publicly available monolingual speech foundation models for high-resource languages (Evain et al., 2021; Wang et al., 2021; Javed et al., 2022; Malmsten et al., 2022; Getman et al., 2024b; Parcollet et al., 2024; Sawada et al., 2024).

In the past, various training methods have been explored for wav2vec 2.0. Still, its inference is

most commonly done via a greedy decoding algorithm. Here, we explore whether a more advanced technique called prefix beam search (Hannun et al., 2014) could lead to better results. The main issue with the standard greedy algorithm stems from the blank symbol, which usually receives a considerable portion of the probability mass (Jung et al., 2022), thus leading to spiky outputs and many deletion errors. To avoid this unwanted effect, prefix beam search merges multiple paths that would result in the same output, lessening the suppression effect of the blank output. While this technique was originally proposed to be used with recurrent models, its variants have been successfully utilized with large SSL models (Jung et al., 2022) and encoder-decoder-based architectures (Zhao et al., 2024) too, albeit those works also employ an external LM during the decoding procedure. In contrast, we only utilize prefix beam search to decode the wav2vec 2.0 model without any LM parts, as low-resource languages often lack in terms of text data too, which prevents the development of a good LM.

Besides the training and decoding algorithms, we also take a closer look at our models' mistakes and propose a new interpretation-based solution to learn more about the reasons for the misrecognition. One of our main observations revealed systematic, repeating mistakes, which we hypothesized were due to the dominance of the Internal LM developed by the model during the finetuning phase (Zeyer et al., 2021a). To validate this hypothesis, we utilized the Integrated Gradients (IG) technique (Sundararajan et al., 2017) to investigate whether the model behaves differently when it predicts various characters. Our experiments revealed that several characters which caused the problems were predominantly outputted by using mainly the long-term information embeddings while ignoring the current acoustic information. Furthermore, we have found that the model dedicated considerably more neurons towards detecting the rare Sámi-specific characters compared to the common Latin characters.

In summary, in this paper, we made the following contributions:

- Developed the first Northern Sámi speech foundation models ¹.

¹<https://huggingface.co/collections/GetmanY1/wav2vec2-sami-22k-66ead12fe465d6302b63d11b>

- Compared the greedy decoding algorithm with the prefix beam search algorithm without any LM component.
- Proposed a model interpretation technique to investigate why the model makes certain mistakes.

2 Methods

2.1 Continued Pre-Training

While standard pre-training of wav2vec 2.0 implies random initialization of the model weights, another training option is utilizing weights of an existing foundation model from a closely related language(s). Getman et al. (2024a) has demonstrated that continued pre-training on a small, 100-hour dataset can improve the downstream out-of-domain ASR performance. In this work, we take a step further and analyze whether this technique is useful even when a sufficient amount of unlabeled in-domain data is available.

Continued pre-training differs from pre-training from scratch only during the model initialization phase; otherwise, it follows the same standard training pipeline. A side effect of this approach is catastrophic forgetting (McCloskey and Cohen, 1989), which hinders the models' performance on language(s) they have been originally pre-trained on (Qian et al., 2024). However, one of the goals of this work is to develop monolingual foundations for a low-resource minority language rather than expand the mono- and multilingual models' capabilities to a new language.

2.2 Prefix Beam Search

End-to-end ASR models like wav2vec 2.0 are often trained with the Connectionist temporal classification (CTC) algorithm in the finetuning phase (Graves et al., 2006). While CTC offers a convenient way of training, the resulting models are well-known to suffer from various problems; namely, the blank label introduced by CTC usually obtains very high probabilities dominating the sequence of outputted symbols, and non-blank outputs display a peaky behavior (Zeyer et al., 2021b). These problems together mean that CTC-trained models often have high deletion errors, as the blank label could easily suppress the emission of actual characters, especially when the model is uncertain.

Prefix Beamsearch (Hannun et al., 2014) offers an alternative to the standard greedy decod-

ing algorithm by considering multiple paths that would result in the same output and combining the probabilities of these paths to gain a more accurate estimate of character emission probabilities. For example, if we consider a short window of 4 timesteps in which the model should recognize the character "a", then the greedy decoding would require that the output unit linked to "a" would get the maximum probability at least in one frame. In many cases, this assumption is not true. Thus, the character is deleted if the probability of the blank (\emptyset) is high. In the beam search algorithm, all possible combinations of \emptyset and "a" are considered, and the probabilities of these paths (e.g. $\emptyset\emptyset a\emptyset$, or $\emptyset aa\emptyset$, or $\emptyset\emptyset aa$, etc.) are added together, often surpassing the probability of purely \emptyset output, preventing the character deletion problem.

The algorithm was originally proposed for recurrent models, and RNN-T architectures, but here we demonstrate that it is applicable even with wav2vec 2.0 models, without any LM. In practice, we fix all the LM probabilities as 1 and feed the logit values of wav2vec 2.0 after a softmax layer to the decoding algorithm.

2.3 IG-based error analysis

For a long time, large foundation models, like any other deep neural network, were considered a black box. With the advancement made in the field of model explainability (Schwalbe and Finzel, 2021), it is now possible to peak inside these huge models and investigate their internal functions. In this work, we selected the technique called Integrated Gradients (IG) (Sundararajan et al., 2017) to learn more about the internal representations of our systems. IG belongs to the family of gradient-based posthoc interpretation tools, meaning that no modifications of the training algorithm or the model architecture are needed to gain insight. In essence, IG estimates the gradients of the relevant output units with respect to certain hidden neurons, and these values are called attributions. In Grósz et al. (2023), it was demonstrated that IG can be used to filter out the irrelevant neurons of various foundation models, without any significant performance loss. Inspired by these findings, here we employed IG to unveil how our models predict certain characters.

Our primary goal was to understand when the model makes decisions mainly based on acoustic information, and when the Internal LM

(ILM) (Zeyer et al., 2021a) becomes dominant. Several techniques have already been proposed to estimate the ILM developed during supervised training. In Zeyer et al. (2021a); Chen et al. (2023), the authors suggest masking out the encoder (acoustic) output to find the ILM scores or employing the so-called density ratio method. Unfortunately, these techniques are not applicable in our case as our model does not have a decoder part, and it is not autoregressive, thus we developed an alternative IG-based solution.

In our experiments, we choose to focus on two specific layers of the wav2vec 2.0 model; namely the feature embeddings of the CNN component, which can be considered as acoustic features, and the convolutional positional embedding layer’s output, where temporal information is introduced to the model. Using IG, we estimated the attributions of each neuron in these two layers per output units. Here we used the predicted (most probable) output at each timestep to estimate the attributions. Next, to approximate the importance of each layer, we calculated the sum of the absolute attributions of neurons inside the two layers. Our motivation for using the absolute values was simple; we did not want to lose valuable information if some neurons had both large negative and positive attributions at different times. Lastly, once the overall attribution of the two layers’ was known, the attribution ratio was calculated by dividing the positional embedding layer’s attribution by the feature embedding layer’s. In this context, an attribution ratio of 1 means that the positional embedding layer has the exact same information as the feature embeddings (i.e. it has no extra influence on the outputs), while a ratio of 2 means that the new temporal information introduced by the positional embedding layer is equally important compared to the acoustic one. Naturally, a ratio above 2 implies that the temporal information is valued more than the acoustic features, which is a sign of the ILM dictating the final output.

3 Data

For pre-training the Sámi models, we extracted 35,614 hours of radio broadcasts of Yle Saamen Radio. The broadcasts have been originally recorded by the Radio and Television Archive (RTVA) since 2009 and provided for research by the Finnish National Audiovisual Institute (KAVI).

Since the raw dataset also contained a considerable amount of non-speech events, including music and silence, we pre-processed it with a neural voice activity detector (VAD) (Bredin, 2023). After that, continuous speech segments longer than 30 seconds were split into shorter utterances. The final size of pre-training data was 22,415 hours, meaning that nearly 37% of the audio was recognized as non-speech.

For the ASR fine-tuning, we used the Sámi Parliament data ² featuring about 20 hours of transcribed speech. For testing the ASR models, 1 hour of out-of-domain read-aloud and spontaneous speech of varying audio quality was used.

4 Experiments

We pre-trained the foundation models with the Fairseq toolkit (Ott et al., 2019). Pre-training was done on 512 GPUs of the LUMI supercomputer ³ for 125,000 steps (approx. 115 epochs) for the Base models (95M parameters) and 167,000 steps (approx. 100 epochs) for the Large ones (317M parameters). The models were then fine-tuned on the Sámi Parliament data for 60 epochs with Huggingface Transformers (Wolf et al., 2020). In continued pre-training, we adapted models originally pre-trained on European Parliament plenary session recordings (Wang et al., 2021). The Base model was a monolingual Finnish foundation, while the Large one also included speech from two other Uralic languages (Hungarian and Estonian).

We evaluated the models with the standard ASR performance metrics such as word and character error rate (WER and CER) and compared them to existing ASR solutions, including Whisper (Radford et al., 2023) fine-tuned on 34 hours of spontaneous Northern Sámi (Hiovain-Asikainen and De la Rosa, 2023) and XLS-R (Babu et al., 2022) first fine-tuned on high-resource Finnish data and then adapted to Northern Sámi with the Sámi Parliament data (Getman et al., 2024a).

Table 1 summarizes the ASR results. Compared to the previously developed solutions, the Base-sized models provided lower WER but higher CER. In contrast, when switching to the Large models, more considerable improvements can be observed.

Next, we performed statistical significance tests

on both the word and character levels using the Matched Pair Sentence Segment approach. To run the tests, we employed the SCTK toolkit ⁴. Looking at the models with continued pre-training, models of both sizes gave significantly ($p \leq 0.001$) lower CER compared to pre-training from scratch, but only the Large one significantly ($p \leq 0.05$) outperformed its counterpart pre-trained from scratch on the word level.

Switching from greedy decoding to prefix beam search further improved the CER. On the word level, however, a significant improvement can be observed only for the Large model pre-trained from scratch, while it insignificantly changed the error rate in either direction for the rest of the models. A more detailed analysis of the results revealed that the prefix beam search always increased the number of substitutions and insertions but decreased the number of deletions compared to greedy decoding.

Overall, the best results were obtained by continued pre-training of the Large model. It gave a noticeable improvement on a character level over pre-training on the same data from scratch (14% relative CER reduction), which may suggest that continued pre-training allowed the model to benefit from acoustic patterns learned from other languages and combine them with the newly learned acoustic information of the target language. On the other hand, minor changes in the WER and the distribution of error rates in Figure 1 may indicate that the gained language knowledge was still not sufficient enough to properly recognize complete words.

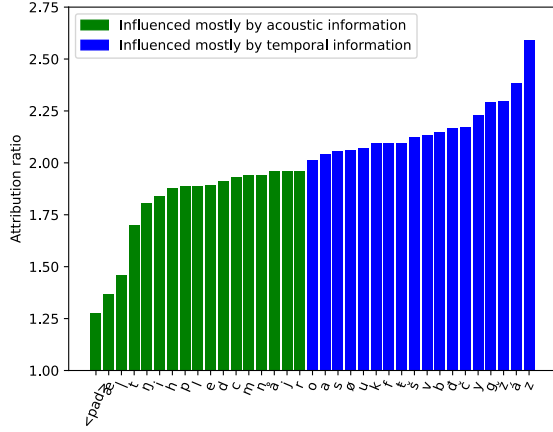
5 Analysis of the results

To better understand how our best model (*Large-22K CPT + Prefix Beam Search*) works, and why it makes certain mistakes, we first inspected the character-level confusion matrix on the test data, see Figure 2. Overall, most characters could be recognized with relatively good accuracy, and only a few rare characters like å, ä, x, ö have extremely low recognition rates. While these mistakes can be explained by the lack of training data, we also noticed other systematic problems on the word level. One such issue was related to the word "na" (in English: "well"), which was quite common in the training data. Interestingly, in the test set other similar words, like "ni" and "no" were almost al-

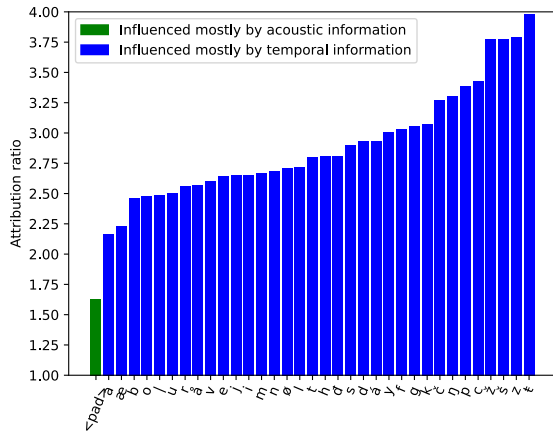
²<https://sametinget.kommunetv.no/archive>

³<https://www.lumi-supercomputer.eu/>

⁴<https://github.com/usnistgov/SCTK>



(a) Continued pre-training.



(b) Pre-training from scratch.

Figure 3: The attribution ratios between the positional and features embeddings per character.

After a closer look at the ratios per character (see Figure 3a), we identified two groups; in the first one, the ratio was below 2, suggesting that these were predicted mainly using acoustic features. This group includes characters such as "h", "i", "n", etc. On the other hand, we can see several characters, including "a", which were primarily predicted by the influence of the internal LM. These results imply that for some characters the acoustic component of the model was not good enough, and it would benefit from seeing additional training material with more diverse textual content in order to force the model to rely more on the acoustic information.

Next, we investigated the counterpart of the best model, trained from scratch (Large-22K), see Figure 3b. This model demonstrated a quite different behavior: all tokens except the blank la-

bel had an attribution ratio above 2, meaning that the system's output was determined mostly by the temporal information added by the positional embeddings. The average attribution ratio for non-blank characters was 2.7, signaling that the acoustic component had a considerably smaller attribution towards the output than the internal LM. Considering that the model was pre-trained only with a relatively small dataset, we can conclude that the acoustic component produced by the continued pre-training is more appropriate and extracts more relevant information. The purely Northern Sámi model's overreliance on temporal information indicates that it most probably obtained most of its knowledge by simply memorizing parts of the training transcripts during the fine-tuning phase, as large models are prone to do so (Huang et al., 2022; Wang et al., 2024). Validating this theory is out of the scope of this paper, but remains an important future task.

Lastly, we also investigated individual neurons in the two selected layers. Here, we aimed to find out which character needed the most actively contributing neurons. We looked at each neuron's attribution values per character. First, we calculated the average and standard deviation of the attributions in each layer. Our first observation at this stage was that the majority of the neurons had an attribution close to the mean (which was approximately 0 in all cases), and only a few neurons displayed large attributions similar to the findings of (Grósz et al., 2023). Based on these observations, we decided to separate the neurons into two groups; the highly contributing ones, whose accumulated attribution was farther than one standard deviation from the mean, and the rest categorized as low-contributing.

Figure 4 illustrates the amount of highly attributing neurons in each investigated layer of the best model. The first observation is that common characters like "r", "b" and "k" required only a few dedicated neurons, while special Sámi characters like "t" and "æ" were predicted based on a large number of neurons. In general, many latin characters required less than a 100 highly contributing neurons, while many Sámi charaters needed more units. This implies that the acoustic features of the CPT model were quite good for most Latin characters that were well represented in the original pre-training corpus, while some ("d", "c" and "t") required more units, perhaps due to non-standard

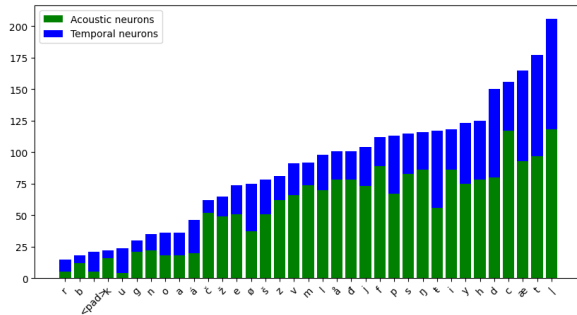


Figure 4: Number of highly attributing neurons in the best model. Acoustic neurons refer to units in the feature embedding layer, while Temporal ones can be found in the positional embedding layer.

pronunciation. Additionally, we can see that the model dedicated a larger portion of neurons to the Sámi specific outputs, implying that despite the language adaptation via CPT and finetuning, it still has difficulties recognizing them well.

6 Limitations

While experimental results suggest that prefix beam search is beneficial on the character level, its WERs proved to be quite similar to the greedy decoding algorithm’s. As the lower CER suggests better quality output, testing its readability by humans and comparing it to the greedy alternative remains an important future task. Additionally, we should mention that here, we utilized the prefix search without any modifications, but it might benefit from adjustments in terms of hyperparameters and vocabulary usage of wav2vec 2.0, especially regarding the word separator symbol.

While our model interpretation experiments have revealed interesting facts about the internal functions of the models, they should be rigorously tested and validated. On the one hand, interpretation techniques are known to be fragile (Ghorbani et al., 2019). Thus, our experiments should be repeated with other attribution estimation methods to ensure that our observations hold. Furthermore, we made several simplifications in this work, including the decision to accumulate the attributions over time, thus ignoring their changes in different contexts. In the future, we intend to investigate how the attributions’ trajectories change over time and in different contexts to gain a deeper understanding of when temporal information is valued more than acoustic information. Lastly, all of our findings should be validated by the use of

a reliable ILM estimation method. Unfortunately, currently, no such technique is available for non-autoregressive models.

7 Conclusions

In this work, we presented the first speech foundation models for Northern Sámi. In addition to standard greedy decoding, we tested prefix beam search, which showed a slight improvement in terms of CER by reducing the number of deletions. Although continued pre-training of a multilingual foundation did not bring a considerable improvement in downstream ASR performance compared to pre-training from scratch, deeper IG-based analysis demonstrated differences in the internal behavior of these two models and revealed that the one pre-trained from scratch was heavily influenced by the temporal information (internal LM), while its counterpart with continued pre-training relied more on its acoustic component when predicting certain characters.

References

- Arun Babu, Changan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli. 2022. Xls-r: Self-supervised cross-lingual speech representation learning at scale. In *Interspeech 2022*, pages 2278–2282.
- Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Processing Systems*, volume 33, pages 12449–12460. Curran Associates, Inc.
- Konstantin Bogdanoski, Kostadin Mishev, Monika Simjanoska, and Dimitar Trajanov. 2023. Exploring asr models in low-resource languages: Use-case the macedonian language. In *Deep Learning Theory and Applications*, pages 254–268, Cham. Springer Nature Switzerland.
- Hervé Bredin. 2023. pyannote.audio 2.1 speaker diarization pipeline: principle, benchmark, and recipe. In *INTERSPEECH 2023*, pages 1983–1987.
- Zhipeng Chen, Haihua Xu, Yerbolat Khassanov, Yi He, Lu Lu, Zejun Ma, and Ji Wu. 2023. Knowledge distillation approach for efficient internal language model estimation. In *INTERSPEECH 2023*, pages 1339–1343.
- Solène Evain, Ha Nguyen, Hang Le, Marcely Zanon Boito, Salima Mdhaffar, Sina Alisamir, Ziyi Tong,

- Natalia Tomashenko, Marco Dinarelli, Titouan Parcollet, Alexandre Allauzen, Yannick Estève, Benjamin Lecouteux, François Portet, Solange Rossato, Fabien Ringeval, Didier Schwab, and Laurent Besacier. 2021. LeBenchmark: A Reproducible Framework for Assessing Self-Supervised Representation Learning from Speech. In *Proc. Interspeech 2021*, pages 1439–1443.
- Yaroslav Getman, Tamas Grosz, Katri Hiovain-Asikainen, and Mikko Kurimo. 2024a. Exploring adaptation techniques of large speech foundation models for low-resource ASR: a case study on Northern Sámi. In *Interspeech 2024*, pages 2539–2543.
- Yaroslav Getman, Tamas Grosz, and Mikko Kurimo. 2024b. What happens in continued pre-training? analysis of self-supervised speech models with continued pre-training for colloquial finnish asr. In *Interspeech 2024*, pages 5043–5047.
- Amirata Ghorbani, Abubakar Abid, and James Zou. 2019. Interpretation of neural networks is fragile. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):3681–3688.
- Peter Gilles, Léopold Edem Ayité Hillah, and Nina Hosseini Kivanani. 2023. Asrlux: Automatic speech recognition for the low-resource language luxembourgish. In *Proceedings of the 20th International Congress of Phonetic Sciences*. Guarant International.
- Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. *Proceedings of the 23rd International Conference on Machine Learning*, page 369–376.
- Tamás Grósz, Anja Virkkunen, Dejan Porjazovski, and Mikko Kurimo. 2023. Discovering Relevant Sub-spaces of BERT, Wav2Vec 2.0, ELECTRA and ViT Embeddings for Humor and Mimicked Emotion Recognition with Integrated Gradients. In *Proceedings of the 4th Multimodal Sentiment Analysis Challenge and Workshop: Mimicked Emotions, Humour and Personalisation (MuSe ’23)*. ACM.
- Awani Y. Hannun, Andrew L. Maas, Daniel Jurafsky, and Andrew Y. Ng. 2014. First-Pass Large Vocabulary Continuous Speech Recognition using Bi-Directional Recurrent DNNs.
- Katri Hiovain-Asikainen and Javier De la Rosa. 2023. Developing tts and asr for lule and north sami languages. In *2nd Annual Meeting of the ELRA/ISCA SIG on Under-resourced Languages (SIGUL 2023)*, pages 48–52.
- W. Ronny Huang, Steve Chien, Om Dipakbhai Thakkar, and Rajiv Mathews. 2022. Detecting unintended memorization in language-model-fused asr. In *Interspeech 2022*, pages 2808–2812.
- Tahir Javed, Sumanth Doddapaneni, Abhigyan Raman, Kaushal Santosh Bhogale, Gowtham Ramesh, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh M. Khapra. 2022. Towards building asr systems for the next billion users. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10813–10821.
- Minkyu Jung, Ohhyeok Kwon, Seung Byum Seo, and Soonshin Seo. 2022. Blank collapse: Compressing ctc emission for the faster decoding. *ArXiv*, abs/2210.17017.
- Jan Lehečka, Josef V. Psutka, Lubos Smidl, Pavel Ircing, and Josef Psutka. 2024. A comparative analysis of bilingual and trilingual wav2vec models for automatic speech recognition in multilingual oral history archives. In *Interspeech 2024*, pages 1285–1289.
- Martin Malmsten, Chris Haffenden, and Love Börjesson. 2022. Hearing voices at the national library – a speech corpus and acoustic model for the swedish language.
- Michael McCloskey and Neal J. Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. volume 24 of *Psychology of Learning and Motivation*, pages 109–165. Academic Press.
- Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.
- Titouan Parcollet, Ha Nguyen, Solène Evain, Marcely Zanon Boito, Adrien Pupier, Salima Mdhaffar, Hang Le, Sina Alisamir, Natalia Tomashenko, Marco Dinarelli, Shucong Zhang, Alexandre Allauzen, Maximin Coavoux, Yannick Estève, Mickael Rouvier, Jérôme Gouliau, Benjamin Lecouteux, François Portet, Solange Rossato, Fabien Ringeval, Didier Schwab, and Laurent Besacier. 2024. Lebenchmark 2.0: A standardized, replicable and enhanced framework for self-supervised representations of french speech. *Computer Speech & Language*, 86:101622.
- Mengjie Qian, Siyuan Tang, Rao Ma, Kate Knill, and M.J.F. Gales. 2024. Learn and don’t forget: Adding a new language to asr foundation models. In *Interspeech 2024*, pages 2544–2548.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine Mcleavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 28492–28518. PMLR.
- Kei Sawada, Tianyu Zhao, Makoto Shing, Kentaro Mitsui, Akio Kaga, Yukiya Hono, Toshiaki Wakatsuki, and Koh Mitsuda. 2024. Release of pre-trained

- models for the Japanese language. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 13898–13905, Torino, Italia. ELRA and ICCL.
- Gesina Schwalbe and Bettina Finzel. 2021. A comprehensive taxonomy for explainable artificial intelligence: a systematic survey of surveys on methods and concepts. *Data Mining and Knowledge Discovery*, pages 1–59.
- Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70, ICML’17*, page 3319–3328. JMLR.org.
- Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. 2021. VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. In *Proceedings of the 59th Annual Meeting of the ACL and the 11th ICNLP (Volume 1: Long Papers)*, pages 993–1003.
- Lun Wang, Om Thakkar, Zhong Meng, Nicole Rafidi, Rohit Prabhavalkar, and Arun Narayanan. 2024. Efficiently train asr models that memorize less and perform better with per-core clipping. In *Interspeech 2024*, pages 1320–1324.
- 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 EMNLP: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Albert Zeyer, Andr’e Merboldt, Wilfried Michel, Ralf Schlüter, and Hermann Ney. 2021a. Librispeech transducer model with internal language model prior correction. In *Interspeech*.
- Albert Zeyer, Ralf Schlüter, and Hermann Ney. 2021b. Why does CTC result in peaky behavior? *CoRR*, abs/2105.14849.
- Zeyu Zhao, Peter Bell, and Ondřej Klejch. 2024. Exploring Dominant Paths in CTC-Like ASR Models: Unraveling the Effectiveness of Viterbi Decoding. In *2024 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSP)*, pages 868–872.

OpusDistillery: A Configurable End-to-End Pipeline for Systematic Multilingual Distillation of Open NMT Models

Ona de Gibert¹ Tommi Nieminen¹ Yves Scherrer^{1,2} Jörg Tiedemann¹

¹University of Helsinki, Dept. of Digital Humanities

²University of Oslo, Dept. of Informatics

¹firstname.lastname@helsinki.fi

²firstname.lastname@ifi.uio.no

Abstract

In this work, we introduce OpusDistillery, a novel framework to streamline the Knowledge Distillation (KD) process of multilingual NMT models. OpusDistillery’s main features are the integration of openly available teacher models from OPUS-MT and Hugging Face, comprehensive multilingual support and robust GPU utilization tracking. We describe the tool in detail and discuss the individual contributions of its pipeline components, demonstrating its flexibility for different use cases. OpusDistillery is open-source and released under a permissive license, aiming to facilitate further research and development in the field of multilingual KD for any sequence-to-sequence task. Our code is available at <https://github.com/Helsinki-NLP/OpusDistillery>.

1 Introduction

Neural Machine Translation (NMT) has continuously improved, offering higher-quality translations and supporting an ever-increasing number of languages. However, these advancements come with significant computational costs. The resources required for both training and, more critically, using these models can be quite expensive. As a response to this trend, there has been a growing effort in the field to optimize these large systems by producing smaller models that are easier to deploy in practical settings. Knowledge Distillation (KD) (Hinton et al., 2015) is a compression technique that allows to build such systems. In KD, a powerful large model, referred to as the *teacher*, is *distilled* into a more compact model, faster and smaller in size, known as the *student*, that tries to match the performance of the teacher by mimicking its output.

In this work, we introduce OpusDistillery, a novel open-source toolkit for performing distillation of open NMT models in multilingual scenarios. We leverage publicly available tools and release our code in our Github repository under the Mozilla Public License 2.0. We intend our pipeline to serve researchers as well as industry players in NMT or any sequence-to-sequence task.

2 Background and Motivation

Our tool implements both standard Sequence-Level Knowledge Distillation (Seq-KD) and its enhanced version, interpolated Seq-KD. Seq-KD, first introduced by Kim and Rush (2016), trains a student model on the sentence-level outputs produced by a teacher model. This process involves two main steps: (1) generating a synthetic dataset by forward translating the source text using the teacher model, and (2) training the student model on this generated data. Despite its simplicity, Seq-KD has been shown to outperform more sophisticated methods for multilingual NMT (Gumma et al., 2023).

Building on Seq-KD, Kim and Rush (2016) further introduced Sequence-Level Interpolation. It enhances Seq-KD by using beam search to generate multiple translations (K-translations) and selecting the most similar sentence to the ground truth for distillation, based on smoothed sentence-BLEU (Chen and Cherry, 2014). This interpolated approach has been demonstrated to surpass the performance of standard Seq-KD; however, the ground truth may not always be available, as distillation can also be performed using monolingual data only.

The challenge of applying KD in multilingual settings is still underexplored. Several studies have attempted to address this task (Tan et al., 2018; Sun et al., 2020; Dabre and Fujita, 2020; Diddee et al., 2022; Do and Lee, 2023), yet there is no standard framework available. To the best of

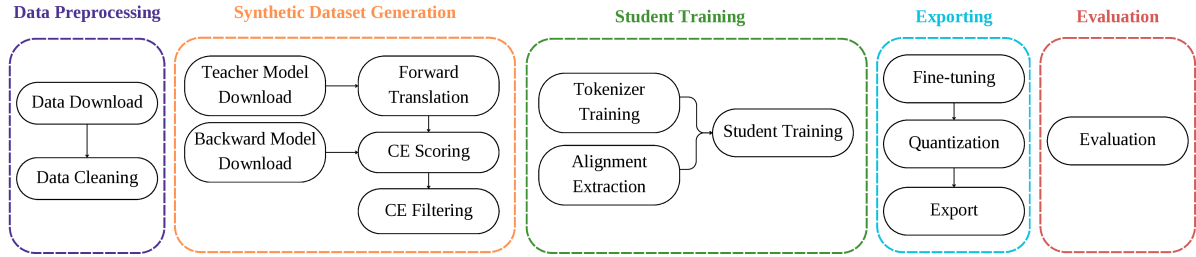


Figure 1: Overview of the OpusDistillery pipeline. *CE* stands for Cross-Entropy.

our knowledge, there exists only one other open toolkit to perform multilingual Seq-KD. *Stopes* (Andrews et al., 2022) is a framework of modular pipelines developed within the NLLB project that allows to recreate their distilled models for reproducibility purposes, but provides little flexibility.

Our motivation for developing OpusDistillery is driven by the need to address this limitation. First, our pipeline provides a versatile toolkit that is easy to configure to perform systematic distillation for NMT in any kind of multilingual setting. Second, we emphasize the use of external, openly available pre-trained teacher models, similar to the approach in Galiano-Jiménez et al. (2023). We advocate for the reuse of public models as a practical and economical solution. This approach not only leverages the continuous publication of new models in open-source repositories such as Hugging Face (HF)¹, but also significantly reduces the costs associated with training from scratch.

3 The OpusDistillery Pipeline

OpusDistillery is an extension of the Firefox Translation Training pipeline (FTT)². The FTT tool trains bilingual NMT teacher models and distills them to produce student models. It was originally developed within the Bergamot project³ for training efficient NMT models that can run locally in a web browser on CPU. The final student is a quantized model, fast at decoding and ready to be fed to the Bergamot-translator application.⁴

The pipeline works by feeding a YAML configuration file to Snakemake (Mölder et al., 2021), a workflow management system that enables the definition of computational pipelines through rules specifying their input and output files. When the

expected output files of a particular rule are absent, Snakemake systematically backtracks to identify and execute the necessary preceding rules in sequence to produce the required outputs. The tool uses the Marian toolkit (Junczys-Dowmunt et al., 2018) for training and SentencePiece (Kudo and Richardson, 2018) for segmentation.

3.1 Main features

Our work implements the use of public pre-trained models as teachers, multilinguality support and the tracking of GPU utilisation.

Use of Open Models as Teachers OpusDistillery allows to distill an open-source pre-trained model. We have added support for using OPUS-MT models⁵ and models from the HF hub. We chose to implement OPUS-MT models because of their broad selection, which includes both bilingual and multilingual variants, as well as their free availability. We have added rules for subword segmentation since OPUS-MT models use their own SentencePiece tokenizers. Furthermore, we support HF systems, allowing the user to choose from a wide range of pre-trained models available on the hub. This seamless integration with our pipeline ensures flexibility and ease of use, enabling users to leverage the diverse and continuously updated models within both ecosystems.

Multilinguality Support Multilingual NMT has been shown to be highly beneficial, especially for low-resource languages that lack sufficient training data (Arivazhagan et al., 2019). OpusDistillery enables the training and distillation of NMT models in any multilingual scenario. This covers two aspects: the ability to use any combination of bilingual and multilingual teachers, as well as the flexibility to train either bilingual or multilingual students. Regarding multilinguality,

¹<https://huggingface.co/>

²<https://github.com/mozilla/firefox-translations-training>

³<https://browser.mt/>

⁴<https://github.com/browsermt/bergamot-translator>

⁵<https://github.com/Helsinki-NLP/OPUS-MT-train>

we have included support for many-to-one (m2o), one-to-many (o2m) and many-to-many (m2m) settings.

GPU Tracking With the goal of moving towards a greener NLP field and for the sake of transparency, we have added GPU utilisation tracking along all steps so that users can report the amount of hours and energy consumed by their experiments. The GPU tracking records the output of `roc-smi` (for AMD GPUs) or `nvidia-smi` (for Nvidia GPUs), depending on the environment, every 10 seconds; monitoring both energy consumption and GPU usage.

3.2 Configuration Files

The pipeline takes a YAML definition file as input, containing all the relevant information for the current experiment. The essential descriptors are the teacher model(s) we want to distill from, as well as the data for training and evaluation. For multilingual scenarios and OPUS-MT models, we have to specify whether the teacher and the student model are multilingual at the target side. In that case, the corresponding language tag will be automatically added. Specific training arguments for SentencePiece and Marian can be overwritten in the configuration file, as for example, a specific architecture for the student model.

```
experiment:
  dirname: baseline
  name: eng-zle
  langpairs:
    - en-uk
    - en-ru
    - en-be

  opusmt-teacher: "best"
  opusmt-backward: "best"

  one2many-teacher: True
  one2many-backward: False
  one2many-student: True

datasets:
  train:
    - tc_Tatoeba-Challenge-v2023-09-26
  devtest:
    - flores_dev
  test:
    - flores_devtest
```

Figure 2: Sample YAML configuration file for OpusDistillery.

3.3 Main Steps

Our pipeline can be divided in five major steps: data preparation, synthetic dataset generation, student training, exporting and evaluation. A high-level overview of the steps is shown in Figure 1. A detailed summary can be consulted in Table 2.

Data Preparation This step includes downloading monolingual and parallel data from public repositories like MTDData (Gowda et al., 2021) and OPUS (Tiedemann and Thottingal, 2020), or using custom datasets. We have added support for using the Tatoeba Challenge data (Tiedemann, 2020), a collection of all datasets available in OPUS, deduplicated and shuffled. Next, data cleaning is performed, an essential step to filter noisy internet data (Kreutzer et al., 2022), with options for basic filtering (e.g., removing sentences by length) and advanced filtering using OpusFilter (Aulamo et al., 2020).

Synthetic Dataset Generation After preparing the data, the pipeline generates the synthetic dataset via forward translation with the teacher. Users can specify pre-trained models, or choose the best available OPUS-MT model⁶. By default, translations are generated using interpolated Seq-KD following Bogoychev et al. (2020). We produce the 8-best translations and keep the most similar output to the ground truth based on smoothed sentence-BLEU (Chen and Cherry, 2014). Reducing the beam to 1 removes the interpolation step and reduces the procedure to standard Seq-KD. Optionally, Cross-Entropy (CE) filtering Junczys-Dowmunt (2018) can reduce noise by removing the 5% lowest-scoring translations with a backward model.

Student Training The student model is trained on the filtered dataset with guided alignment. This step includes training the tokenizer with SentencePiece, extracting word alignments using eflomal (Östling and Tiedemann, 2016),⁷ and generating lexical shortlists for faster decoding. The pipeline supports running multiple experiments efficiently, training compact models based on the

⁶The top-scoring model on a given benchmark (our current implementation uses the Flores-200 (Goyal et al., 2022) and the OPUS-MT Dashboard (Tiedemann and De Gibert, 2023) as a reference point).

⁷The experiments for this paper were run with fast_align (Dyer et al., 2013) that was part of the earlier implementation, which is now replaced by eflomal due to its better performance.

tiny architecture from Bogoychev et al. (2020). The student’s resulting size has 16.9M parameters and occupies 65MB, 12.6 times smaller than transformer-big and 3.8 times smaller than transformer-base architectures.

Exporting The exporting step creates the final student. First, the student is fine-tuned by emulating 8bit quantization during training to make the model more robust. Then, the fine-tuned student is quantized to 8 bits to further reduce its size. Finally, the export step which saves the model so it is ready for deployment. On average, the exported model translates 3119,3 words per second on a single AMD MI250x GPU.

Evaluation The last step is to evaluate all of our models trained (student, fine-tuned, quantized). Evaluation is performed using sacreBLEU (Post, 2018), ChrF (Popović, 2015), and COMET metrics (Rei et al., 2020).

We can illustrate the pipeline steps for a given configuration file as a Directed Acyclic Graph (DAG). OpusDistillery automatically generates the DAG. Figure 3 illustrates the final steps of the pipeline before evaluation. When dealing with multiple languages, the graphs become very complex quickly, as there are so many steps involved. A toolbox and workflow management system like the one we are presenting in this work is very useful for handling such convoluted procedures.

4 Experiments

To showcase the versatility and capabilities of the presented pipeline, we conduct a series of experiments. We train multilingual student models up until the student training step, without exporting, to showcase the impact of different components. Specifically, we trained student models using the complete pipeline and perform ablation studies by excluding CE filtering, alignment, and both.

Languages Following Do and Lee (2023), we perform our experiments focusing on selected language groups from and into English. For each group, we distill a student model from multiple teachers. We test three language families paired with English, each of them containing three languages and with diverse linguistic characteristics:

- Finno-Ugric languages (fiu):
Finnish (fi), Estonian (et), Hungarian (hu).

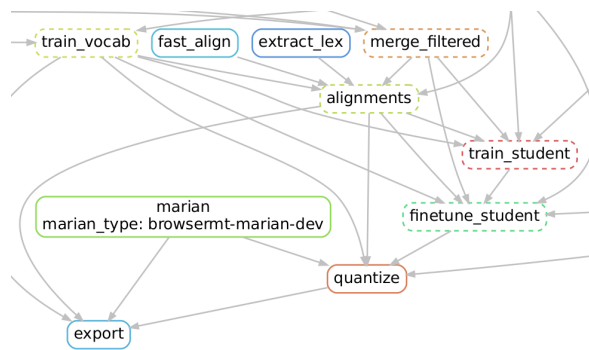


Figure 3: DAG of the OpusDistillery for the final steps before evaluation.

- Romance languages (rom):
Catalan (ca), Spanish (es), Occitan (oc).
- East Slavic languages (zle):
Ukrainian (uk), Russian (ru), Belarussian (be).

Data We use the parallel Tatoeba Translation Challenge dataset, sampling up to 10 million sentence pairs per language pair when available. Occitan, being a low-resource language, had a smaller dataset of approximately 200k sentences. We applied default cleaning and used the Flores-200 development and test sets for evaluation.

Teacher models For each language pair, we selected the best OPUS-MT teacher available using the implemented feature of best teacher selection. Each teacher model was also used as a backward model for the opposite translation direction for CE scoring. Their performance is reported in Table 1 for reference.

4.1 Results

Results are shown in Table 1. “Student” refers to the student model trained with all the steps in the pipeline, including CE filtering and guided alignment. Overall, student models generally perform 5 BLEU points lower than teacher models due to their reduced capacity. However, our objective in this work is to introduce the tool and demonstrate its application. OpusDistillery will enable future research to optimize multilingual student models further.

Performance across student models was consistent, with minimal variation. In some cases, removing CE filtering produced better results, though its overall impact was minimal. Students

	Finno-Ugric-English			Romance-English			East Slavic-English		
	et-en	fi-en	hu-en	ca-en	es-en	oc-en	be-en	ru-en	uk-en
Teacher	38.59	35.72	34.60	45.40	29.86	46.64	18.10	35.21	39.23
Type	big-bi	big-bi	big-bi	big-m2o	big-m2o	big-m2m	big-m2o	big-m2o	big-m2o
Student	28.92	26.86	27.79	40.89	25.41	32.67	15.36	30.31	33.51
w/o CE-filtering	29.97	27.65	28.23	41.17	25.24	32.17	15.80	30.12	33.80
w/o Alignment	29.95	27.32	29.05	40.72	25.48	32.78	15.83	30.39	33.66
w/o CE & A	29.36	27.54	28.42	40.93	25.42	32.49	15.80	30.00	33.15
	English-Finno-Ugric			English-Romance			English-East Slavic		
	en-et	en-fi	en-hu	en-ca	en-es	en-oc	en-be	en-ru	en-uk
Teacher	28.27	27.58	29.58	41.52	28.45	31.60	11.23	32.66	32.14
Type	big-bi	big-bi	big-bi	big-bi	big-bi	base-o2m	big-o2m	big-o2m	big-o2m
Student	22.56	19.55	23.13	38.79	25.28	27.73	10.19	26.54	25.95
w/o CE-filtering	23.09	20.06	23.51	38.70	24.58	27.98	10.32	26.27	27.02
w/o Alignment	23.20	19.95	23.99	39.05	25.26	28.35	10.43	26.58	27.29
w/o CE & A	22.98	19.89	23.42	38.58	24.84	26.72	10.34	25.97	26.48

Table 1: Results of our distillation experiments in BLEU. We include the performance of the teacher as a reference, as well as its size (transformer-big or transformer-base) and its multilinguality: bilingual (bi), many-to-one languages (m2o), one-to-many (o2m) and many-to-many (m2m).

trained without guided alignment slightly outperformed the baseline. Omitting both CE filtering and alignment resulted in comparable performance, suggesting that these steps can be skipped without significant quality loss while reducing the number of pipeline steps.

5 Conclusions and Future Work

In this work, we have presented OpusDistillery, an end-to-end pipeline to perform systematic multilingual distillation of open NMT models. Through our experiments, we demonstrated its effectiveness and versatility by training English-centric models for three distinct language groups using the Tatoeba Challenge dataset. We explored the individual contributions of the CE filtering and guided alignment steps, revealing that simplifying the pipeline can slightly enhance student model performance.

OpusDistillery is open source and distributed under a permissive license. We hope that our research benefits the community by enabling them to perform distillation of publicly available models and to contribute to the development of more efficient and accessible language technologies.

In future work, we plan to extend the pipeline to better accommodate multilingual scenarios by integrating additional tools, such as employing BicleanerAI (Zaragoza-Bernabeu et al., 2022) and incorporating monolingual data, which is now not implemented. Furthermore, we aim to explore

the use of Large Language Models (LLMs) to enhance performance. Additionally, we intend to implement alternative distillation strategies, such as word-level distillation (Kim and Rush, 2016).

Ethics Statement

With the goal of moving towards a greener NLP field, the OpusDistillery pipeline automatically reports GPU and energy usage. This allows us to measure the carbon footprint used in this work. The four main steps of the pipeline that use GPU are listed below, together with their average GPU hours, energy consumed (kWh), and GPU usage (%):

- Translation: 10.37 h 15.17 kWh 86.74 %
- CE scoring: 0.57 h 0.98 kWh 78.27 %
- Training: 32.88 h 49.87 kWh 77.10 %
- Evaluation: 0.05 h 0.02 kWh 0.45 %

As expected, training accounts for the highest energy consumption, while scoring and evaluation require the least. The GPU usage of the evaluation step is rather low, since the experiments were run only using sacreBLEU. We anticipate that the recent implementation of COMET will improve the utilization of the GPU during evaluation, leading to both a more efficient use of resources and a more comprehensive performance assessment.

Acknowledgements

This project has received funding from the European Union’s Horizon Europe research and innovation programme under Grant agreement No 101070350 and from UK Research and Innovation (UKRI) under the UK government’s Horizon Europe funding guarantee [grant number 10052546]. The contents of this publication are the sole responsibility of its authors and do not necessarily reflect the opinion of the European Union. This work was also supported by the GreenNLP project funded by the Research Council of Finland. The authors wish to thank CSC – IT Center for Science, Finland for computational resources and support.

References

- Pierre Andrews, Guillaume Wenzek, Kevin Heffernan, Onur Çelebi, Anna Sun, Ammar Kamran, Yingzhe Guo, Alexandre Mourachko, Holger Schwenk, and Angela Fan. 2022. [stopes - modular machine translation pipelines](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 258–265, Abu Dhabi, UAE. Association for Computational Linguistics.
- Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George F. Foster, Colin Cherry, Wolfgang Macherey, Zhifeng Chen, and Yonghui Wu. 2019. [Massively multilingual neural machine translation in the wild: Findings and challenges](#). *CoRR*, abs/1907.05019.
- Mikko Aulamo, Sami Virpioja, and Jörg Tiedemann. 2020. [OpusFilter: A configurable parallel corpus filtering toolbox](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 150–156, Online. Association for Computational Linguistics.
- Nikolay Bogoychev, Roman Grundkiewicz, Alham Fikri Aji, Maximiliana Behnke, Kenneth Heafield, Sidharth Kashyap, Emmanouil-Ioannis Farsarakis, and Mateusz Chudyk. 2020. [Edinburgh’s submissions to the 2020 machine translation efficiency task](#). In *Proceedings of the Fourth Workshop on Neural Generation and Translation*, pages 218–224, Online. Association for Computational Linguistics.
- Boxing Chen and Colin Cherry. 2014. A systematic comparison of smoothing techniques for sentence-level bleu. In *Proceedings of the ninth workshop on statistical machine translation*, pages 362–367.
- Raj Dabre and Atsushi Fujita. 2020. [Combining sequence distillation and transfer learning for efficient low-resource neural machine translation models](#). In *Proceedings of the Fifth Conference on Machine Translation*, pages 492–502, Online. Association for Computational Linguistics.
- Harshita Diddee, Sandipan Dandapat, Monojit Choudhury, Tanuja Ganu, and Kalika Bali. 2022. [Too brittle to touch: Comparing the stability of quantization and distillation towards developing low-resource MT models](#). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 870–885, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Heejin Do and Gary Geunbae Lee. 2023. [Target-oriented knowledge distillation with language-family-based grouping for multilingual nmt](#). *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 22(2).
- Chris Dyer, Victor Chahuneau, and Noah A Smith. 2013. A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648.
- Aarón Galiano-Jiménez, Felipe Sánchez-Martínez, Víctor M. Sánchez-Cartagena, and Juan Antonio Pérez-Ortiz. 2023. [Exploiting large pre-trained models for low-resource neural machine translation](#). In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 59–68, Tampere, Finland. European Association for Machine Translation.
- Thamme Gowda, Zhao Zhang, Chris Mattmann, and Jonathan May. 2021. [Many-to-English machine translation tools, data, and pretrained models](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 306–316, Online. Association for Computational Linguistics.
- Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc’Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.
- Varun Gumma, Raj Dabre, and Pratyush Kumar. 2023. An empirical study of leveraging knowledge distillation for compressing multilingual neural machine translation models. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 103–114.
- Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. [Distilling the knowledge in a neural network](#).
- Marcin Junczys-Dowmunt. 2018. Dual conditional cross-entropy filtering of noisy parallel corpora. In

- Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 888–895.
- Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, et al. 2018. Marian: Fast neural machine translation in c++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121.
- Yoon Kim and Alexander M Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327.
- Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, et al. 2022. Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.
- Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71.
- Felix Mölder, Kim Philipp Jablonski, Brice Letcher, Michael B Hall, Christopher H. Tomkins-Tinch, Vanessa V. Sochat, Jan Forster, Soohyun Lee, Sven O. Twardziok, Alexander Kanitz, Andreas Wilm, Manuel Holtgrewe, Sven Rahmann, Sven Nahnsen, and Johannes Köster. 2021. [Sustainable data analysis with snakemake](#). *F1000Research*, 10.
- Robert Östling and Jörg Tiedemann. 2016. Efficient word alignment with markov chain monte carlo. *The Prague Bulletin of Mathematical Linguistics*, 106(1):125.
- Maja Popović. 2015. chrF: character n-gram f-score for automatic mt evaluation. In *Proceedings of the tenth workshop on statistical machine translation*, pages 392–395.
- Matt Post. 2018. [A call for clarity in reporting BLEU scores](#). In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.
- Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. [COMET: A neural framework for MT evaluation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.
- Haipeng Sun, Rui Wang, Kehai Chen, Masao Utiyama, Eiichiro Sumita, and Tiejun Zhao. 2020. Knowledge distillation for multilingual unsupervised neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3525–3535.
- Xu Tan, Yi Ren, Di He, Tao Qin, Zhou Zhao, and Tie-Yan Liu. 2018. Multilingual neural machine translation with knowledge distillation. In *International Conference on Learning Representations*.
- Jörg Tiedemann. 2020. The tatoeba translation challenge—realistic data sets for low resource and multilingual mt. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1174–1182.
- Jörg Tiedemann and Ona De Gibert. 2023. The opusmt dashboard—a toolkit for a systematic evaluation of open machine translation models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 315–327.
- Jörg Tiedemann and Santhosh Thottingal. 2020. Opusmt—building open translation services for the world. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*. European Association for Machine Translation.
- Jaume Zaragoza-Bernabeu, Gema Ramírez-Sánchez, Marta Bañón, and Sergio Ortiz-Rojas. 2022. Bicleaner ai: Bicleaner goes neural. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 824–831.

A Detailed Overview of OpusDistillery Main Steps

Main Step	Step	Resource	Optional	Configurable
Data Processing	Data Download	CPU	✗	✓
	Data Cleaning	CPU	✗	✓
Synthetic Dataset Generation	Teacher Model Download	CPU	✗	✓
	Forward Translation	GPU	✗	✗
	Backward Model Download	CPU	✓	✓
	Cross-Entropy Scoring	GPU	✓	✗
	Cross-Entropy Filtering	CPU	✓	✓
Student Training	Tokenizer Training	CPU	✗	✓
	Alignment Extraction	CPU	✓	✗
	Student Training	GPU	✗	✓
Exporting	Fine-tuning	GPU	✓	✓
	Quantization	CPU	✓	✗
	Export	CPU	✓	✗
Evaluation	Evaluation	GPU	✓	✗

Table 2: Summary of OpusDistillery main steps. For each step, we report the compute resource used (CPU or GPU), whether the step is optional, and whether it is configurable or hard-coded.



Mind the Gap: Diverse NMT Models for Resource-Constrained Environments

Ona de Gibert^{1*} Dayyán O’Brien² Dušan Variš³ Jörg Tiedemann¹

¹University of Helsinki ²University of Edinburgh ³Charles University

*Corresponding author: ona.degibert@helsinki.fi

Abstract

We present fast Neural Machine Translation models for 17 diverse languages, developed using Sequence-level Knowledge Distillation. Our selected languages span multiple language families and scripts, including low-resource languages. The distilled models achieve comparable performance while being 10x times faster than transformer-base and 35x times faster than transformer-big architectures. Our experiments reveal that teacher model quality and capacity strongly influence the distillation success, as well as the language script. We also explore the effectiveness of multilingual students. We release publicly our code and models in our Github repository: <https://github.com/hplt-project/bitextor-mt-models>.

1 Introduction

Neural Machine Translation (NMT) has seen significant advancements with the advent of Large Language Models (LLMs; Zhu et al., 2024). Although LLMs often perform exceptionally well on high-resource languages, their performance on low-resource languages lags behind (Stap and Araabi, 2023; Kocmi et al., 2023; Robinson et al., 2023). Nevertheless, recent advancements suggest that this gap may be narrowing (Enis and Hopkins, 2024).

Despite their high quality performance, LLMs come with substantial computational costs, requiring significant amount of training data, high-end hardware and extensive energy consumption (Rae et al., 2021). These limitations make LLMs unsuitable for many real-world scenarios where resources are constrained, such as on-device translation, low-latency requirements, or environments with privacy concerns.

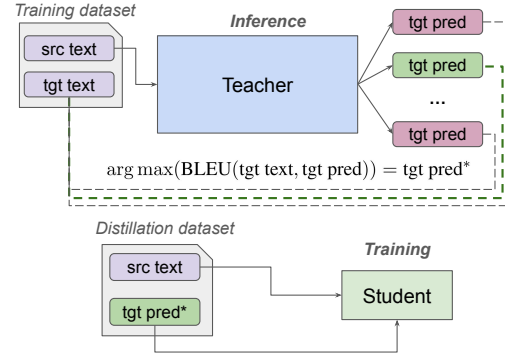


Figure 1: Conceptual overview of interpolated Sequence-Level Knowledge Distillation.

The traditional sequence-to-sequence (seq2seq) Transformer architecture (Vaswani et al., 2017), though not as versatile as LLMs, offers considerable advantages in terms of computational efficiency. These models can be optimized to run faster, consume less memory, and require fewer resources, making them a practical solution for many NMT applications (Kim et al., 2019; Aji and Heafield, 2020).

In this work, we leverage Knowledge Distillation (KD) (Hinton et al., 2015; Kim and Rush, 2016) to train compact seq2seq NMT models. KD allows the transfer of knowledge from a large, high-performing *teacher* model to a smaller, more efficient *student* model.

We present fast NMT models for 17 diverse languages with English as the target language. The selected languages vary widely in terms of script, language family, and resource availability, including low-resource languages like North Azerbaijani and high-resource languages like Hindi.

In our experiments, we address the following Research Questions (RQ): *RQ1: How does the capacity gap affect the distillation quality?*, *RQ2: To what extent does script influence the transfer of knowledge?* and *RQ3: Can we train multilingual students effectively?*

2 Related Work

We use Sequence-level KD (Seq-KD, Kim and Rush, 2016), which has proven to be effective to do KD for NMT (Gumma et al., 2023; Team et al., 2024). In Seq-KD, the teacher model is used to forward-translate all the sentences in the training data to create a distilled dataset. In the interpolated Seq-KD variant, the teacher generates K-candidate translations, selecting the one with the highest smoothed sentence BLEU (Chen and Cherry, 2014) with the reference. Then, the student model is trained on the synthetically generated data. Figure 1 illustrates this procedure. In this way, the lightweight student retains much of the teacher’s performance while being optimized for speed and efficiency.

Several studies explore how to build compact NMT models. With the motivation of testing the time-efficiency of NMT systems, a shared task on NMT efficiency was organized for several years within the Workshop on Neural Generation and Translation (Hayashi et al., 2019; Heafield et al., 2020, 2021). Research has focused on various aspects, including compressing multilingual systems (Tan et al., 2018), investigating different architectures for student models (Bogoychev et al., 2020), and understanding the effectiveness of KD (Zhou et al., 2020). One widely adopted approach is the thin and deep architecture (Gala et al., 2023; Gumma et al., 2023), characterized by a deep encoder and a shallow decoder (Mohammadshahi et al., 2022; Kasai et al., 2020), which has become a standard for compressing NMT models. We follow that approach in this work.

3 Methodology

Next, we describe the selected languages, datasets, tools, and teacher and student architectures used for our experiments.

Languages The 17 selected languages are listed in Table 1. To highlight their diversity, we provide the language family (spanning 13 distinct families) and the script, representing seven different scripts: Arabic (Arab), Latin (Latn), Hebrew (Hebr), Devanagari (Deva), Japanese (Jpan), Cyrillic (Cyr), Hangul (Hang). We also include the taxonomy class proposed by Joshi et al. (2020) to classify languages according to their available resources. It ranges from 1 (resources for that language are limited) to 5 (rich-resource languages).

Language	Family	Class	Data (M)
Arabic (arb_Arab)	Semitic	5	10.44
Basque (eus_Latn)	Isolate	4	6.40
Catalan (cat_Latn)	Romance	4	29.23
Galician (glg_Latn)	Romance	3	7.78
Hebrew (heb_Hebr)	Semitic	3	28.90
Hindi (hin_Deva)	Indo-Iranian	4	13.62
Japanese (jpn_Jpan)	Japonic	5	15.81
Kazakh (kaz_Cyrl)	Turkic	3	21.28
Korean (kor_Hang)	Koreanic	4	7.56
Latvian (lvs_Latn)	Baltic	3	24.73
Lithuanian (lit_Latn)	Baltic	3	34.70
Slovak (slk_Latn)	Slavic	3	53.66
Swahili (swh_Latn)	Bantu	2	6.27
Malay (zsm_Latn)	Austronesian	3	42.65
N. Azerbaijani (azj_Latn)	Turkic	1	44.46
N. Uzbek (uzn_Latn)	Turkic	3	17.55
Vietnamese (vie_Latn)	Austro-Asiatic	4	2.83

Table 1: Overview of the selected languages, including their script, language family, class as defined by Joshi et al. (2020) and training data (in millions of sentences).

Datasets We use the Tatoeba Challenge dataset, a compilation of all datasets available in OPUS (Tiedemann et al., 2024), de-duplicated and shuffled. Other datasets include: MaCoCu (Bañón et al., 2022, 2023) for Catalan; CLUVI (Universidade de Vigo, 2012) for Galician; SAWA (De Pauw et al., 2009) and Gourmet (Sánchez-Martínez et al., 2020) for Swahili. We use a combination of OpusCleaner (Bogoychev et al., 2023) and OpusFilter (Aulamo et al., 2020) for cleaning the corpora. We list the clean training data sizes for each language pair in Table 1. For development and evaluation, we use Flores-200 (Goyal et al., 2022).

Tools We train our models with interpolated Seq-KD with three different tools: we follow recipes from the Bergamot project¹, the Firefox Translations training pipeline² and its extended multilingual version, OpusDistillery (de Gibert et al., 2025). All tools perform a forward translation of the training data to create the distilled dataset, generating an 8-best list of candidate translations, as illustrated in Figure 1. Using the distilled dataset, we train a new, shared 32k subword vocabulary with SentencePiece (Kudo and Richardson, 2018), alignments with fast_align (Dyer et al., 2013) and lexical shortlists for faster

¹<https://github.com/browsermt/students/tree/master/train-student>

²<https://github.com/mozilla/firefox-translations-training>

decoding with `extract.lex`³. Then, we train the student with guided alignment using the Marian NMT toolkit (Junczys-Dowmunt et al., 2018). Finally, we quantize the student models using an 8-bit integer representation, which significantly reduces memory usage while maintaining translation quality.

OPUS-MT teacher models All teachers are OPUS-MT transformers (tf). We use one single teacher for each student model. Five teachers are tf-base (~ 70 M parameters) while the remaining are tf-big (~ 209 M params). We show the size of each teacher in Table 3. We train our own tf-big teachers for Galician and Swahili. For the other languages, we use the OPUS-MT dashboard (Tiedemann and De Gibert, 2023) to choose the best available teacher.

Tiny student models Our student models adopt the tiny architecture proposed by Bogoychev et al. (2020), consisting of a transformer encoder with 6 layers and a lightweight RNN-based decoder with the Simpler Simple Recurrent Unit (SSRU, Kim et al., 2019) with 2 layers. In a pilot study, we initially trained both small and tiny student models, with a detailed comparison of their architectures provided in Table 2. Results from this study showed that the translation quality loss in tiny models was minimal compared to the small models. Consequently, we opted to focus exclusively on the tiny models, which offer substantial inference speedups. After training, we quantize the model. **On average, the tiny architecture is 10x times faster than tf-base and 35x times faster than tf-big architectures.**

We train bilingual student models for all language pairs except for the Baltic and Turkic families, for which we train multilingual many-to-one students.

Evaluation We use COMET⁴ (Rei et al., 2020) and spBLEU (Goyal et al., 2022) for evaluation. COMET is a neural metric that demonstrates the highest correlation with human judgments in translation quality assessment. It covers all tested languages. Additionally, we use SacreBleu (Post, 2018) to compute spBLEU, which refers to the BLEU (Papineni et al., 2002) metric on the tokenized text with SentencePiece.

³<https://github.com/marian-nmt/extract-lex>

⁴We use the model `Unbabel/wmt22-comet-da`.

	Teachers		Students	
	big	base	small	tiny
N_{enc}	6	6	6	6
N_{dec}	6	6	2	2
d_{emb}	1024	512	512	256
d_{ff}	4096	2048	2048	1536
h	16	8	8	8
Params (M)	213	65	39	17
Size (MB)	798	277	42	17
Speed (tok/s)	814.8	2758.5	18649.5	28854.7

Table 2: Comparison of tf architectures used for teachers (big, base) and students (small, tiny). The table lists the number of encoder and decoder layers (N_{enc} and N_{dec}), embedding dimensions (d_{emb}), feed-forward dimensions (d_{ff}), number of attention heads (h), parameters in millions, model size in MB, and decoding speed in tokens per second. Speed values are averaged across all models on 32 CPU cores.

4 Results

Tables 3 and 4 summarize the results of our distillation experiments in COMET scores for bilingual and multilingual settings, respectively. We report spBLEU scores in Tables 5 and 6 in the Appendix.

On average, the students exhibit a drop of 2.9 COMET points compared to their teachers. In general, we observe that our students maintain competitive performance, with high scores for several languages, including Catalan, Galician, Hebrew, Slovak, and Malay. These results indicate that, despite the reduction in model size and complexity, these students still capture a significant portion of the teacher’s knowledge. However, for languages like Arabic, Korean and Japanese, the scores drop significantly. For Japanese, Table 5 reveals that the teacher model performs the worst among all selected languages, with a spBLEU score of 19.2. **This suggests that a low-performing teacher is not capable of knowledge transfer.** Therefore, we exclude Japanese from our analysis in the next section.

We expect that our students do not outperform their teachers, due to the capacity limitations of the students when compared to their larger teachers, known as the capacity gap problem (Jafari et al., 2021). However, our Catalan student achieves a COMET score 1.1 point higher than its teacher, correlating with a 90% human agreement that it outputs better translations (Kocmi et al., 2024).

Language		ara	cat	eus	glg	heb	hin	jpn	kor	slk	swl	vie	zsm
Teacher	Params (M)	76.4	69.4	235.4	209.1	238.1	75.9	77.5	209.2	235.5	209.1	63.9	237.1
	Performance	83.7	84.3	83.8	87.6	86.2	81.9	80.3	85.3	85.1	82.9	79.2	85.6
Student	Compression	4.5	4.1	13.9	12.4	14.1	4.5	4.6	12.4	13.9	12.4	3.8	14.0
	Performance	76.7	85.4	80.6	84.4	85.2	81.8	62.8	78.9	85.2	79.3	79.8	85.6
	Δ	-7.0	+1.1	-3.3	-3.2	-1.0	-0.1	-17.6	-6.4	+0.1	-3.6	+0.6	+0.1

Table 3: COMET score results of our bilingual distillation experiments. For the teacher models, we report parameters in millions and performance. We provide results for the students, as well as their compression ratio. Δ shows the difference in COMET scores with the teacher.

Family		Baltic		Turkic		
Language		lit	lvs	azn	kaz	uzj
Teacher	Params (M)	236.9	236.9	238.8	238.8	238.8
	Performance	83.5	84.0	82.0	81.7	81.7
Student	Compression	14.02	14.02	14.13	14.13	14.13
	Performance	82.7	83.7	80.2	78.5	78.9
	Δ	-0.8	-0.3	-1.8	-3.2	-2.7

Table 4: COMET score results of our multilingual distillation experiments.

We also find an improved score for Vietnamese, Slovak and Malay, though these improvements were less significant.

5 Discussion

In this section, we address the research questions (RQs) posed in the introduction based on the results of our distillation experiments.

RQ1: How does the capacity gap between the teacher and student models affect the distillation quality? The capacity gap between the teacher and student models is a critical factor in distillation quality. We find that larger teachers (tf-big) lead to a more significant performance drop, with an average COMET reduction of 2.2 compared to tf-base teachers, which exhibit an average of 1.1 COMET. **This directly correlates with the capacity gap problem: the smaller the gap in model size, the better the distillation.** The compression ratios for tf-big teachers are 3.2 times larger, underscoring the complexity of transferring knowledge from a high-capacity teacher to a smaller student.

RQ2: To what extent does script influence the transfer of knowledge? We compare Latin vs. non-Latin scripts because English (the target language in all models) is in the Latin script. Students trained for Latin script languages have an average of 1.2 COMET, while non-Latin script languages have a similar average of 3.5 COMET. **This difference indicates that script plays a role**

in the transfer of knowledge during distillation.

With a fixed vocabulary size, a shared script between source and target lets SentencePiece build longer, more semantically rich subwords. In contrast, non-Latin script languages yield shorter subwords, making knowledge transfer more difficult and reducing translation quality.

RQ3: Can we train multilingual students effectively? The student models for the language families in Table 4 maintain relatively high scores. For example, Lithuanian and Latvian demonstrate that multilingual training can compensate for some of the limitations of model compression, particularly for closely related languages. The Turkic family has a combination of scripts that may hinder knowledge transfer. **Even with the reduced size of the tiny model, we are able to fit multiple languages into a single student.**

6 Conclusions and Future Work

In this paper, we introduced fast MT models for 17 diverse languages, leveraging interpolated SeqKD to compress large teacher models into more efficient students. Our experiments reveal that low-performing teachers struggle to transfer knowledge effectively. We also demonstrate that the capacity gap between teacher and student models, as well as language script, significantly affect distillation performance. Additionally, our results highlight the effectiveness of multilingual distillation for related languages.

For future work, we plan to develop student models for additional languages. We also aim to expand our approach by distilling from a broader range of teacher models available on the HuggingFace Hub⁵ and to further investigate cross-script knowledge transfer.

⁵<https://huggingface.co/>

Acknowledgements

This project has received funding from the European Union’s Horizon Europe research and innovation programme under Grant agreement No 101070350 and from UK Research and Innovation (UKRI) under the UK government’s Horizon Europe funding guarantee [grant number 10052546]. The contents of this publication are the sole responsibility of its authors and do not necessarily reflect the opinion of the European Union.

References

- Alham Fikri Aji and Kenneth Heafield. 2020. Compressing neural machine translation models with 4-bit precision. In *Proceedings of the Fourth Workshop on Neural Generation and Translation*, pages 35–42, Online. Association for Computational Linguistics.
- Mikko Aulamo, Sami Virpioja, and Jörg Tiedemann. 2020. Opusfilter: A configurable parallel corpus filtering toolbox. In *2020 Annual Conference of the Association for Computational Linguistics*, pages 150–156. The Association for Computational Linguistics.
- Marta Bañón, Malina Chichirau, Miquel Esplà-Gomis, Mikel L. Forcada, Aarón Galiano-Jiménez, Cristian García-Romero, Taja Kuzman, Nikola Ljubešić, Rik van Noord, Leopoldo Pla Sempere, Gema Ramírez-Sánchez, Peter Rupnik, Vít Suchomel, Antonio Toral, and Jaume Zaragoza-Bernabeu. 2023. Catalan-english parallel corpus MaCoCuca-en 1.0. Slovenian language resource repository CLARIN.SI.
- Marta Bañón, Miquel Esplà-Gomis, Mikel L. Forcada, Cristian García-Romero, Taja Kuzman, Nikola Ljubešić, Rik van Noord, Leopoldo Pla Sempere, Gema Ramírez-Sánchez, Peter Rupnik, Vít Suchomel, Antonio Toral, Tobias van der Werff, and Jaume Zaragoza. 2022. Macocu: Massive collection and curation of monolingual and bilingual data: focus on under-resourced languages. In *EAMT 2022 - Proceedings of the 23rd Annual Conference of the European Association for Machine Translation*, pages 303–304. European Association for Machine Translation.
- Nikolay Bogoychev, Roman Grundkiewicz, Alham Fikri Aji, Maximiliana Behnke, Kenneth Heafield, Sidharth Kashyap, Emmanouil-Ioannis Farsarakis, and Mateusz Chudyk. 2020. Edinburgh’s submissions to the 2020 machine translation efficiency task. In *The 4th Workshop on Neural Generation and Translation*, pages 218–224. Association for Computational Linguistics (ACL).
- Nikolay Bogoychev, Jelmer van der Linde, Graeme Nail, Barry Haddow, Jaume Zaragoza-Bernabeu, Gema Ramírez-Sánchez, Lukas Weymann, Tudor Nicolae Mateiu, Jindřich Helcl, and Mikko Aulamo. 2023. Opuscleaner and opustrainer, open source toolkits for training machine translation and large language models. *arXiv preprint arXiv:2311.14838*.
- Boxing Chen and Colin Cherry. 2014. A systematic comparison of smoothing techniques for sentence-level bleu. In *Proceedings of the ninth workshop on statistical machine translation*, pages 362–367.
- Guy De Pauw, Peter Waiganjo Wagacha, and Gilles-Maurice de Schryver. 2009. The sawa corpus: a parallel corpus english-swahili. In *Proceedings of the First Workshop on Language Technologies for African Languages*, pages 9–16.
- Chris Dyer, Victor Chahuneau, and Noah A Smith. 2013. A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 644–648.
- Maxim Enis and Mark Hopkins. 2024. From llm to nmt: Advancing low-resource machine translation with claude. *arXiv preprint arXiv:2404.13813*.
- Jay Gala, Pranjal A Chitale, AK Raghavan, Varun Gumma, Sumanth Doddapaneni, Janki Atul Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, Pratyush Kumar, et al. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *Transactions on Machine Learning Research*.
- Ona de Gibert, Tommi Nieminen, Yves Scherrer, and Jörg Tiedemann. 2025. OpusDistillery: A Configurable End-to-End Pipeline for Systematic Multilingual Distillation of Open NMT Models. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, Tallinn, Estonia.
- Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc’Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.
- Varun Gumma, Raj Dabre, and Pratyush Kumar. 2023. An empirical study of leveraging knowledge distillation for compressing multilingual neural machine translation models. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 103–114.
- Hiroaki Hayashi, Yusuke Oda, Alexandra Birch, Ioannis Konstas, Andrew Finch, Minh-Thang Luong, Graham Neubig, and Katsuhito Sudoh. 2019. Findings of the third workshop on neural generation and

- translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 1–14, Hong Kong. Association for Computational Linguistics.
- Kenneth Heafield, Hiroaki Hayashi, Yusuke Oda, Andrew Finch, Graham Neubig, Xian Li, and Alexandra Birch. 2020. Findings of the fourth workshop on neural generation and translation. In *The 4th Workshop on Neural Generation and Translation*, pages 1–9. Association for Computational Linguistics (ACL).
- Kenneth Heafield, Qianqian Zhu, and Roman Grundkiewicz. 2021. Findings of the wmt 2021 shared task on efficient translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 639–651.
- Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network.
- Aref Jafari, Mehdi Rezagholizadeh, Pranav Sharma, and Ali Ghodsi. 2021. Annealing knowledge distillation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2493–2504, Online. Association for Computational Linguistics.
- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.
- Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast neural machine translation in c++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121.
- Jungo Kasai, Nikolaos Pappas, Hao Peng, James Cross, and Noah Smith. 2020. Deep encoder, shallow decoder: Reevaluating non-autoregressive machine translation. In *International Conference on Learning Representations*.
- Yoon Kim and Alexander M Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327.
- Young Jin Kim, Marcin Junczys-Dowmunt, Hany Hassan, Alham Fikri Aji, Kenneth Heafield, Roman Grundkiewicz, and Nikolay Bogoychev. 2019. From research to production and back: Ludicrously fast neural machine translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 280–288, Hong Kong. Association for Computational Linguistics.
- Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thammie Gowda, Roman Grundkiewicz, Barry Haddow, Philipp Koehn, Benjamin Marie, Christof Monz, Makoto Morishita, Kenton Murray, Makoto Nagata, Toshiaki Nakazawa, Martin Popel, Maja Popović, and Mariya Shmatova. 2023. Findings of the 2023 conference on machine translation (wmt23): LLMs are here but not quite there yet. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1–42, Singapore. Association for Computational Linguistics.
- Tom Kocmi, Vilém Zouhar, Christian Federmann, and Matt Post. 2024. Navigating the metrics maze: Reconciling score magnitudes and accuracies. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1999–2014, Bangkok, Thailand. Association for Computational Linguistics.
- Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71.
- Alireza Mohammadshahi, Vassilina Nikoulina, Alexandre Bérard, Caroline Brun, James Henderson, and Laurent Besacier. 2022. Small-100: Introducing shallow multilingual machine translation model for low-resource languages. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8348–8359.
- 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.
- Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.
- Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, H. 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, Mari-beth 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 M. 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 J. Johnson, Blake A. Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Edward Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorraine Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. 2021. Scaling language models: Methods, analysis & insights from training gopher. *CoRR*, abs/2112.11446.
- Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. Comet: A neural framework for mt evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702.
- Nathaniel Robinson, Perez Ogayo, David R. Mortensen, and Graham Neubig. 2023. ChatGPT MT: Competitive for high- (but not low-) resource languages. In *Proceedings of the Eighth Conference on Machine Translation*, pages 392–418, Singapore. Association for Computational Linguistics.
- Felipe Sánchez-Martínez, Víctor M Sánchez-Cartagena, Juan Antonio Pérez-Ortiz, Mikel L Forcada, Miquel Espla-Gomis, Andrew Secker, Susie Coleman, and Julie Wall. 2020. An english-swahili parallel corpus and its use for neural machine translation in the news domain. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 299–308.
- David Stap and Ali Araabi. 2023. ChatGPT is not a good indigenous translator. In *Proceedings of the Workshop on Natural Language Processing for Indigenous Languages of the Americas (Americas-NLP)*, pages 163–167, Toronto, Canada. Association for Computational Linguistics.
- Xu Tan, Yi Ren, Di He, Tao Qin, Zhou Zhao, and Tie-Yan Liu. 2018. Multilingual neural machine translation with knowledge distillation. In *International Conference on Learning Representations*.
- NLLB Team et al. 2024. Scaling neural machine translation to 200 languages. *Nature*, 630(8018):841.
- Jörg Tiedemann, Mikko Aulamo, Daria Bakshandaeva, Michele Boggia, Stig-Arne Grönroos, Tommi Nieminen, Alessandro Raganato, Yves Scherrer, Raúl Vázquez, and Sami Virpioja. 2024. Democratizing neural machine translation with opus-mt. *Language Resources and Evaluation*, 58(2):713–755.
- Jörg Tiedemann and Ona De Gibert. 2023. The opus-mt dashboard—a toolkit for a systematic evaluation of open machine translation models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 315–327.
- 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.
- Grupo de investigación TALG Universidade de Vigo. 2012. Cluvi parallel corpus.
- Chunting Zhou, Jiatao Gu, and Graham Neubig. 2020. Understanding knowledge distillation in non-autoregressive machine translation. In *International Conference on Learning Representations*.
- Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2024. Multilingual machine translation with large language models: Empirical results and analysis. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2765–2781.

A spBLEU results

Language		ara	cat	eus	glg	heb	hin	jpn	kor	slk	swh	vie	zsm
Teacher	Params (M)	76.4	69.4	235.4	209.1	238.1	75.9	77.5	209.2	235.5	209.1	63.9	237.1
	Performance	37.6	45.1	33.6	44.5	46.5	32.1	19.2	30.1	43.2	41.0	28.7	44.4
Student	Compression	4.5	4.1	13.9	12.4	14.1	4.5	4.6	12.4	13.9	12.4	3.8	14.0
	Performance	29.7	43.3	26.7	39.5	41.2	29.8	7.4	22.2	38.4	35.4	29.9	41.4
	Δ	-7.9	-1.8	-6.9	-5.0	-5.3	-2.3	-11.8	-7.9	-4.8	-5.6	+1.2	-3.0

Table 5: spBLEU score results of our bilingual distillation experiments. For the teacher models, we report parameters in millions and performance. We provide results for the students, as well as their compression ratio. Δ shows the difference in spBLEU scores with the teacher.

Family Language		Baltic		Turkic		
		lit	lvs	azn	kaz	uzj
Teacher	Params (M)	236.9	236.9	238.8	238.8	238.8
	Performance	34.0	36.2	24.2	30.0	32.0
Student	Compression	14.02	14.02	14.13	14.13	14.13
	Performance	31.3	32.9	20.2	24.3	26.1
	Δ	-2.9	-3.3	-4.0	-5.7	-5.9

Table 6: spBLEU score results of our multilingual distillation experiments.

Testing relevant linguistic features in automatic CEFR skill level classification for Icelandic

Isidora Glišić
University of Iceland
Reykjavík, Iceland
isidora@hi.is

Caitlin Laura Richter
Reykjavík University
Reykjavík, Iceland
caitlinr@ru.is

Anton Karl Ingason
University of Iceland
Reykjavík, Iceland
antoni@hi.is

Abstract

This paper explores the use of various linguistic features to develop models for automatic classification of language proficiency on the CEFR scale for Icelandic, a low-resourced and morphologically complex language. We train two classifiers to assess skill level of learner texts. One is used as a baseline and takes in the original unaltered text written by a learner and uses predominantly surface features to assess the level. The other uses both surface and other morphological and lexical features, as well as context vectors from transformer (IceBERT). It takes in both the original and corrected versions of the text and takes into account errors/deviation of the original texts compared to the corrected versions. Both classifiers show promising results, with baseline models achieving between 62.2-67.1% accuracy and dual-version between 75-80.3%.

(NLP) and highlight the importance of broadening research efforts beyond high-resourced languages.

The Icelandic L2 Error Corpus (IceL2EC), published in 2022 (Ingason et al., 2022), has served as a foundational dataset for analyzing features associated with the CEFR skill level. In particular, manually corrected text versions and error annotation have shown a high value in predicting proficiency levels through machine learning approaches (Glišić, 2023). To explore automatic assessment further, this study builds on IceL2EC and includes additional unpublished texts sourced from the University of Iceland. Using this combined dataset, several models were trained to test the efficacy of various features for the assessment of the CEFR skill level. We present the results of baseline models using K-nearest neighbors (KNN) algorithm which uses the learners' original texts and basic linguistic features, and "dual-version" models (logistic regression - LR) which integrate the corrected versions of the data, and more complex features.

1 Introduction

Language skill level assessment is a critical component in language education and testing, and accurate and scalable methods for assessing skill levels can facilitate personalized learning, enhance testing systems, and contribute to linguistic research. However, automating this process presents significant challenges, especially for low-resourced languages such as Icelandic. In this paper, we present findings from an ongoing study focused on using linguistic features to train models for automatic skill level assessment in Icelandic as a second language (L2) texts on the CEFR scale, a widely adopted framework in language education. By focusing on Icelandic we aim to contribute to the growing body of work on underrepresented languages in natural language processing

Key research questions addressed in this study include: (1) How accurately can linguistic features predict writing skill levels in Icelandic L2 texts, and (2) To what extent do corrected texts and advanced models like IceBERT (Snæbjarnarson et al., 2022) contribute to improved classification performance? We incorporate surface features, morphological and lexical elements, and IceBERT-derived context vectors to provide a comprehensive approach to automatic skill level assessment.

The paper is organized as follows: Section 2 provides a background on L2 Icelandic, the CEFR scale, and automatic skill level detection. Sections 3 and 4 detail our models and evaluation metrics, while Section 5 presents the experimental results.

2 Background

Icelandic stands out among lesser spoken languages for its relatively robust digital resources (Nikulásdóttir et al., 2020; Nikulásdóttir et al., 2022). However, the resources available for Icelandic as a learner language (L2) are sparse mainly due to L2 Icelandic being a relatively recent phenomenon and collecting written data for L2 Icelandic is challenging. However, in recent years the number of foreign nationals in Iceland has surged. In the mid-1990s, only 2% of Iceland’s population were first-generation immigrants; by early 2023, this number reached approximately 17.3%. (Hagstofa Íslands, 2023). This demographic shift has heightened the importance of developing resources for L2 Icelandic. An essential aspect of teaching and assessing a second language is measuring learner skill level. The CEFR standardizes skill level assessment with a six-level scale (A1 to C2), focusing on communicative competencies rather than specific linguistic structures (Council of Europe, 2018). Icel2EC, developed under a government-sponsored language technology initiative (see Nikulásdóttir et al., 2020), is the primary resource available for investigating CEFR-labeled learner errors and interlanguage features; data for a new learner corpus is currently being collected to build on these foundations.

Automatic classification of skill level in written texts remains challenging due to the subjective nature of language proficiency scales like the CEFR. A critical component in skill assessment involves the selection of linguistic features that effectively capture learners’ proficiency. Thus, with learner corpora and error tagging, researchers can identify relevant linguistic patterns that correspond to specific CEFR levels. In English, for example, accuracy rates for automatic CEFR classification range from 62.7% to 83.8% (Kerz et al., 2021). Using features derived from lexical, morphological, and syntactic patterns, classifiers like logistic regression and more advanced approaches have achieved promising results in multilingual proficiency assessment tasks, as seen in studies with L2 German, Swedish, and Estonian (Kerz et al., 2021; Vajjala and Lõo, 2014). Importantly, model evaluation metrics must consider the proximity between CEFR levels, recognizing that misclassifications between adjacent levels (e.g., C1 and B2) are less severe than those between distant levels (e.g., C1 and A1). Additionally, language proficiency as-

essment carries significant implications, as its results can influence the learner’s educational and professional opportunities. In this context, predicting a higher level is generally less harmful to the learner than predicting a lower one.

3 Model training

This study establishes preliminary models for automated skill level classification. Baseline models, utilizing only original texts, are compared with dual-version models that use both original and corrected texts. Feature-based approaches yield high prediction accuracy, especially for morphologically rich languages (see Weiss et al., 2021, Reynolds, 2016), and this study combines surface, morphologic, and lexical features, as suggested in recent research (see Pilán and Volodina, 2018, Yekrangi, 2022, Curto et al., 2015), as well as combining context vectors from transformers and perplexity score, typically used to evaluate the performance of language models. For Icelandic proficiency classification, we adapt representative models for these approaches, whose established performance in other languages provides additional context for the results we observe in Icelandic. In this section we introduce the dataset, models and features selected for our task.

3.1 Dataset

Training data consists of Icel2EC, the first published corpus of L2 Icelandic which has 101 student essays categorized by skill level, manually corrected and annotated for errors. Initial CEFR level labels were made based on the students’ academic progress and assessment by a human annotator (Glišić and Ingason, 2022). To validate these levels, inter-annotator agreement was reached with five experienced Icelandic L2 instructors, and the final level assignments reflect the averaged ratings from this team. The corpus includes writing assignments of varying lengths, from 150–200 word beginner texts to several thousand words advanced essays, leading to an uneven distribution of data across skill levels. To create a more balanced dataset for model training, 83 additional unpublished texts from the Practical Diploma Program in Icelandic (A1/A2) were added. Additionally, the texts were cleaned by removing all non-Icelandic sentences, and longer texts (in particular full BA and MA theses) were chunked into 40–50 sentences segments to fit

BERT maximum token length, resulting in a total of 276 texts with a more even training support across levels.

Level	Texts	Total Words	Sentences
A1	73	7,820	913
A2	38	10,204	913
B1	37	21,960	1,229
B2	31	22,457	1,052
C	97	84,730	3,873

Table 1: Distribution of data for each level

Given that the ongoing project on Icelandic CEFR alignment currently emphasizes levels A1 through B2, the advanced levels C1 and C2 were merged into a single advanced category, labeled "C." The final dataset thus spans a five-level scale, with A1 and A2 representing beginner, B1 and B2 intermediate, and C advanced levels, as depicted in Table 1.

3.2 Feature selection for baseline

Baseline models used only features that can be computed from shallow analysis of the text. Minimal feature sets were selected from those suggested by (Yekrani, 2022), inspired by older formulas for assessing text complexity. The total length of the text, along with features like type-token ratio that are considered excessively influenced by it according to consensus in cross linguistic literature (McCarthy and Jarvis, 2010), were excluded from baseline models as confounds.

Baseline-Minimal uses two features: average word length, the number of letters per token; and HD-D, the hypergeometric distribution of lexical (word) diversity, an alternative to type-token ratio (McCarthy and Jarvis, 2010).

Baseline-Lemma requires lemmatisation (stemming) (Ingason et al., 2008) and a frequency list of the language’s vocabulary (Arnardóttir and Ingason, 2023), but no further language processing technology or resources. PoS-tagging and lemmatization for all models tested was conducted with ABL Tagger (Steingrímsson et al., 2019) and the Nefnir lemmatizer (Ingólfssdóttir et al., 2019). Some features expect CEFR aligned vocabularies, but lacking one for Icelandic, this implementation assigns the 1000 most frequent words to A1, and so on, following the teaching resource RÚV Orð¹.

¹<https://ord.ruv.is/>

The features included are: average word length and HD-D as in Baseline-Minimal; ATTR; CLI; average vocabulary level of tokens’ lemmas, advanced vocabulary percentage, the percentage of the text’s lemmas not in CEFR A or B inventories; and Dale-Chall readability score (DCRS), a formula combining the proportion of "difficult" tokens (lemmas not in A-B1) with the average number of words per sentence.

3.3 Feature selection for dual version models

The dual-version models incorporate two versions of the data — original and corrected — and include surface features, morphological features derived from PoS-tagging and lemmatization, lexical diversity metrics (word frequencies, tf-idf weighted words), and NLP-based features like contextual embeddings from transformers and text perplexity extracted from originals. Key features that highlight differences between text versions are cosine similarity and average error count per sentence.

Dual-ling uses linguistic features primarily inspired by Pilán and Volodina’s feature set (2018), with several adaptations. Key features include average sentence length, percentage of long words (over six characters), average error counts per sentence, and cosine similarity between original and corrected texts; morphological features include proportions of pronouns, past participles, conjunctions, articles, and subjunctive forms; lexical features include average lemma count, average vocabulary level of lemmas, and tf-idf weighted terms for uni-, bi-, and trigrams in both original text and PoS tags.

Dual-expand is supplemented by incorporating IceBERT, an Icelandic language model based on the BERT architecture, which estimates word probability given its context (Snæbjarnarson et al., 2022). The IceBERT-igc feature selection pipeline was applied to derive embeddings and extract relevant features from the dataset, and the model was used to calculate the perplexity of the original texts.

4 Evaluation

Both baseline and dual-version models were tested on several algorithms, including linear regression, SVM, KNN, LR, and MLP. After initial testing, K-nearest neighbors (K=10) was viewed for baseline evaluations, while logistic regression was selected

for the dual-version evaluations. Each model and feature set was assessed using an 80/20 train-test split, with stratified sampling. It was repeated 1000 times with different random splits, and the reported metrics represent averages of these runs. Accuracy, as the percentage of correct classifications, is sensitive to data distribution and may overlook false positives, disproportionately affecting smaller classes. Additionally, accuracy does not account for the “distance of prediction,” where predicting C1 instead of B2, for example, is a less severe error than predicting A1 instead of B2. For a more comprehensive evaluation, F1 scores were also calculated to provide a balance between precision and recall. Alongside exact accuracy, we assessed adjacent accuracy, i.e. also viewing predictions within one level above or below the true level (e.g., A2 predicted as either A1, A2, or B1) as correct. This metric reflects the CEFR scale’s flexibility and the frequent disagreements between human evaluators.

5 Results

Baseline models showed varied performance, with the highest exact accuracy achieved by the Baseline-Lemma KNN model, which recorded 67.1% exact accuracy and 89.7% with adjacent accuracy. Interestingly, the linear regression model, although performing lower on exact matches at 63.5%, had the highest adjacent accuracy among all models, achieving 97.6%. This suggests that while linear regression may struggle with precise classification, it is particularly effective at capturing a close approximation to the true level. All tested models varied in performance between CEFR levels, with levels A1 and C showing better performance across the board, as seen in Tables 2 and 3.

Class	Precision	Recall	F1	Support
A1	0.79	0.97	0.87	15
A2	0.59	0.36	0.43	8
B1	0.33	0.30	0.30	7
B2	0.42	0.23	0.28	6
C	0.75	0.84	0.79	20

Table 2: Average Performance Statistics for the Baseline-Lemma KNN Model

Table 4 presents a comparative overview of the average accuracy (exact and adjacent) across models. The Dual-Ling model, which combines orig-

Class	Precision	Recall	F1	Support
A1	0.89	1.00	0.94	16
A2	0.75	0.50	0.60	6
B1	0.78	0.70	0.74	10
B2	1.00	0.11	0.20	9
C	0.62	1.00	0.77	15

Table 3: Average Performance Statistics for Dual-expand LR Model

inal and corrected text features without IceBERT embeddings, achieved the highest exact accuracy at 80.3% and 96.4% when including the one-level deviation. In addition, the introduction of lexical features, especially tf-idf weights, notably improved the models’ performance, with tf-idf for PoS tags alone contributing an average 4% boost in accuracy.

Model	Exact(%)	Adjacent(%)
Baseline-Minimal	62.2	85.9
Baseline-Lemma	67.1	89.7
Dual-Ling	80.3	96.4
Dual-Expand	75.0	94.6

Table 4: Comparative accuracy of KNN and LR models, exact and adjacent

6 Discussion

Findings from this study show that baseline models can achieve moderate classification accuracy, with the Baseline-Lemma KNN model reaching the highest baseline performance (67.1% exact, 89.7% adjacent). Models performed best at A1 and C levels, likely due to both their highest data support as well as distinctiveness, while B1 and B2 had lower F1 scores, reflecting their similarity to adjacent levels and greater classification difficulty. This level of performance aligns with accuracy rates reported for other languages, suggesting that even simple feature sets can yield effective results for Icelandic, despite the limited resources available for L2. We also note that these results are robust to variation in the feature set, as e.g. other two-feature models with different lexical diversity measures in place of HD-D perform about as well as Baseline-Minimal, without clear preference among a few reasonable alternatives at least within present statistical power.

Enhanced models using dual versions and more sophisticated linguistic features outper-

formed baseline models, with the exception of the linear regression baseline model's strong adjacent accuracy (97.6%). The Dual-ling model demonstrated the highest exact accuracy at 80.3% and 96.4% for adjacent. This supports the hypothesis that including corrected versions can improve classifier accuracy by providing insights into corrective changes that reveal interlanguage patterns. Additionally, incorporating lexical features, specifically tf-idf weights for both lexical terms and PoS tags, proved influential in boosting accuracy, underscoring the importance of lexical diversity and usage patterns in prediction. The absence of IceBERT embeddings in the top-performing Dual-ling model suggests that raw contextual embeddings may not be essential for achieving strong performance in this task. However, it remains a question for future research whether using IceBERT for classification or extracting embeddings comparatively could improve results in a more balanced dataset or with a larger corpus.

7 Conclusion

We have demonstrated that integrating surface and deeper linguistic features is notably effective in skill level classification, showing that a blend of lexical, morphological, and contextual data can meaningfully reflect learner proficiency. We found that baseline models performed moderately well, with the Baseline-Lemma KNN model achieving the highest exact accuracy (67.1%) and 89.7% when adjacent accuracy was considered. The Dual-Ling model, relying on both original and corrected text features, achieved the highest overall performance with 80.3% exact and 96.4% adjacent accuracy. These findings have significant implications for future automated tools assessing Icelandic learners' skill levels. However, a challenge with the CEFR lies in its broad descriptors, which lack specific grammatical and lexical competencies for each level, making it difficult to map concrete linguistic features directly to skill levels. The forthcoming Icelandic learner corpus, specifically designed for skill level analysis with balanced data, marks an important step forward. It promises to provide an empirically grounded dataset for further development of automated tools, enabling more accurate skill level assessments.

References

- Pórunn Arnardóttir and Anton Karl Ingason. 2023. Frequency lists for icelandic 23.06. CLARIN-IS.
- Council of Europe. 2018. *Common European Framework of Reference for Languages: Learning, Teaching, Assessment. Companion Volume with New Descriptors*. Council of Europe Publishing, Strasbourg.
- Pedro Curto, Nuno Mamede, and Jorge Baptista. 2015. Automatic text difficulty classifier - assisting the selection of adequate reading materials for european portuguese teaching. pages 36–44.
- Isidora Glišić. 2023. Towards automated icelandic skill level evaluation: A deep dive into I2 error corpus patterns and classification. Master's thesis, University of Iceland, September. Master's Thesis.
- Isidora Glišić and Anton Karl Ingason. 2022. The nature of icelandic as a second language: An insight from the learner error corpus for icelandic. pages 23–33.
- Hagstofa Íslands. 2023. Yfirlit mannfjölda. Accessed: 2023-06-04.
- Anton Karl Ingason, Sigrún Helgadóttir, Hrafn Loftsson, and Eiríkur Rögnvaldsson. 2008. A mixed method lemmatization algorithm using a hierarchy of linguistic identities (holi). In *Advances in natural language processing: 6th international conference, GoTAL 2008 Gothenburg, Sweden, August 25-27, 2008 Proceedings*, pages 205–216. Springer.
- Anton Karl Ingason, Lilja Björk Stefánsdóttir, Pórunn Arnardóttir, Xindan Xu, Isidora Glišić, and Dagbjört Guðmundsdóttir. 2022. The icelandic I2 error corpus (IceL2EC) 1.3 (22.10). CLARIN-IS.
- Svanhvít Lilja Ingólfssdóttir, Hrafn Loftsson, Jón Friðrik Daðason, and Kristín Bjarnadóttir. 2019. Nefnir: A high accuracy lemmatizer for Icelandic. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 310–315, Turku, Finland. Linköping University Electronic Press.
- 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.
- Philip M McCarthy and Scott Jarvis. 2010. "mtld, vocd-d, and hd-d: A validation study of sophisticated approaches to lexical diversity assessment". *Behavior research methods*, 42(2):381–392.
- Anna Björk Nikulásdóttir, Pórunn Arnardóttir, Starkaður Barkarson, Jón Guðnason, Þorsteinn Daði

- Gunnarsson, Anton Karl Ingason, Haukur Páll Jónsson, Hrafn Loftsson, Hulda Óladóttir, Eiríkur Rögnvaldsson, et al. 2022. Help yourself from the buffet: National language technology infrastructure initiative on clarin-is. In *CLARIN Annual Conference*, pages 109–125.
- Anna Björk Nikulásdóttir, Jón Guðnason, Anton Karl Ingason, Hrafn Loftsson, Eiríkur Rögnvaldsson, Eiríkur Freyr Sigurðsson, and Steinþór Steingrímsson. 2020. Language technology programme for icelandic 2019-2023. In *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, pages 3414–3422.
- Ildikó Pilán and Elena Volodina. 2018. Investigating the importance of linguistic complexity features across different datasets related to language learning. In *Proceedings of the Workshop on Linguistic Complexity and Natural Language Processing*, pages 49–58, Santa Fe, New-Mexico. Association for Computational Linguistics.
- Robert Reynolds. 2016. Insights from Russian second language readability classification: complexity-dependent training requirements, and feature evaluation of multiple categories. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 289–300, San Diego, CA. Association for Computational Linguistics.
- Vésteinn Snæbjarnarson, Haukur Barri Símonarson, Pétur Orri Ragnarsson, Svanhvít Lilja Ingólfssdóttir, Haukur Jónsson, Vilhjalmur Thorsteinsson, and Hafsteinn Einarsson. 2022. A warm start and a clean crawled corpus - a recipe for good language models. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4356–4366, Marseille, France. European Language Resources Association.
- Vésteinn Snæbjarnarson, Haukur Simonarson, Pétur Ragnarsson, Svanhvít Lilja Ingólfssdóttir, Haukur Jónsson, Vilhjálmur Þorsteinsson, and Hafsteinn Einarsson. 2022. A warm start and a clean crawled corpus – a recipe for good language models.
- Steinþór Steingrímsson, Örvar Kárason, and Hrafn Loftsson. 2019. Augmenting a BiLSTM tagger with a morphological lexicon and a lexical category identification step. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1161–1168, Varna, Bulgaria. INCOMA Ltd.
- Sowmya Vajjala and Kaidi Lõo. 2014. Automatic CEFR level prediction for Estonian learner text. In *Proceedings of the third workshop on NLP for computer-assisted language learning*, pages 113–127, Uppsala, Sweden. LiU Electronic Press.
- Zarah Weiss, Xiaobin Chen, and Detmar Meurers. 2021. Using broad linguistic complexity modeling for cross-lingual readability assessment. In *Proceedings of the 10th Workshop on NLP for Computer Assisted Language Learning*, pages 38–54, Online. LiU Electronic Press.
- Aryan Yekrani. 2022. "leveraging simple features and machine learning approaches for assessing the cefr level of english texts". Master's thesis, "University of Eastern Finland".

MORSED: Morphological Segmentation of Danish and its Effect on Language Modeling

Rob van der Goot¹
Mikkel Wildner Kildeberg¹

Anette Jensen
Nicolaj Larsen¹

Emil Allerslev Schledermann¹
Mike Zhang² Elisa Bassignana¹

¹IT University of Copenhagen

²Aalborg University

robv@itu.dk

Abstract

Current language models (LMs) mostly exploit subwords as input units based on statistical co-occurrences of characters. Adjacently, previous work has shown that modeling morphemes can aid performance for Natural Language Processing (NLP) models. However, morphemes are challenging to obtain as there is no annotated data in most languages. In this work, we release a wide-coverage Danish morphological segmentation evaluation set. We evaluate a range of unsupervised token segmenters and evaluate the downstream effect of using morphemes as input units for transformer-based LMs. Our results show that popular subword algorithms perform poorly on this task, scoring at most an F_1 of 57.6 compared to 68.0 for an unsupervised morphological segmenter (Morfessor). Furthermore, evaluate a range of segmenters on the task of language modeling.¹

1 Introduction

Although there is no exact consensus on the definition of morphemes (e.g. Nida, 1948; Bolinger, 1948), they are commonly described as the smallest meaning-carrying units in natural language (Sinclair, 1996). Morphemes are useful for linguistic analysis, language understanding, language learning and potentially as input units for NLP models. Traditionally, characters or words were used as inputs for NLP models, but contextualized Language Models (LMs) popularized subwords (Devlin et al., 2019), which are often based on a trained vocabulary obtained with statistical methods. Morphemes, however, are a promising

Input:	frakkeskåner	lærte
MorSeD:	frakke-skån-er	lær-te
TinyBERT:	fra-kk-es-kan-er	l-æ-rte
BPE:	frakke-skå-ner	lærte
WordPiece:	fra-kke-sk-åne-r	lærte
Unigram:	fra-kke-skån-er	lærte
Morfessor:	frakke-skån-er	lært-e

Figure 1: Two examples from our dataset, with the input words, gold morpheme annotation (morsed), and the outputs of: a baseline English language model segmenter (TinyBERT), three Danish statistical segmenters, and a Danish unsupervised morphological segmenter (Morfessor).

alternative as they are of similar granularity but are linguistically motivated. In NLP, morphemes have been successfully used in machine translation models (Clifton and Sarkar, 2011; Popović, 2012), RNN LMs (Blevins and Zettlemoyer, 2019; Schwartz et al., 2020), for static word embeddings (Üstün et al., 2018), and as an auxiliary task in character-level models (Matthews et al., 2018).

Although there have been large multilingual benchmarking efforts for morphological tagging (Zeman et al., 2018) and reinflection (Cotterell et al., 2018), data for morphological segmentation is more scarce, Especially for mid-resource languages, like Danish (Joshi et al., 2020). Therefore, we create a small yet high-coverage benchmark to evaluate unsupervised segmenters for Danish morphological segmentation and provide an extensive evaluation of existing models.

There has been some work that incorporating morphemes as input to LMs. For English, Hofmann et al. (2021) showed that derivational segmentation aids LM interpretation of complex words, and Bostrom and Durrett (2020) showed that using units that closer resemble morphemes improves language modeling (although the mor-

¹Data and code are available on <https://bitbucket.org/robvanderg/morsed>

TYPE	DESCRIPTION
Root Morphemes	The root of a word is its stem, the shortest meaning-bearing part. A root is also called a free morpheme, as it makes sense on its own and often has a concrete meaning.
Compounds	New words in Danish can be formed by combining existing words, creating new meanings. These are compound words and are considered complex. Many compounds are formed solely from root morphemes, which are often nouns, but also adverbs and adjectives.
Compounds with Linking	Some roots in compound words are connected using linking letters, commonly "-e" and "-s." Linking letters are often used when the first root is a verb.
Prefixes	A prefix is a derivative added to the beginning of a word, altering its meaning but not its word class. Prefixes cannot form words on their own.
Suffixes	A suffix is also a derivative, added to the end of a word, typically changing its word class. Like prefixes, they cannot form words on their own.
Inflections	Inflectional morphemes are mainly associated with nouns, verbs, and adjectives. They add information such as gender, definiteness, tense, and mood, but do not form words independently.

Table 1: Description of each type of morphological segmentation we use in our study.

phemes are of relatively low accuracy). Limisiewicz et al. (2024) use morphemes in a multilingual LM. They transform unsupervised morphemes to byte sequences which are used as input sequence to an LM, but they do not evaluate the quality of the morphemes. Our work differs by focusing on Danish, including a wider range of morphemes, evaluating more segmenters, evaluating morpheme performance, and obtaining inputs closer to true morphemes.

Our contributions are: ① We present MORSED, an evaluation dataset for Danish morphological segmentation, including morpheme-level categories and labels. ② We evaluate various segmenters on the task of morphological segmentation: 3 subword algorithms and an unsupervised morphological segmenter. ③ We examine the impact of training data and vocabulary size on tokenizers by training them on 11 different data sources. ④ We assess our tokenizers for language model training using small discriminative transformer-based models.

2 MORSED

Here, we introduce MORSED, to the best of knowledge the first publicly available dataset annotated for morphological segmentation of Danish. We follow the guidelines and categories defined in Jensen (2021). Our main annotator (author of Jensen (2021)) has 35 years of experience as a Danish teacher, with a degree in Teaching and a postgraduate diploma in Adult Literacy Education. The dataset contains 800 words.² The

²Morphological segmentation/labeling datasets are typically smaller than other NLP datasets, even for English. We

words were selected by our main annotator, focusing on diversity and good coverage for each category. In Table 1, we describe each type of morphological segmentation.

A second native Danish annotator without a linguistic background annotated 300 words from MORSED by following the same guidelines. Since inter-annotator scores (e.g., Cohen’s Kappa) are challenging to compute for segmentation tasks, we use F_1 score on the morpheme level for comparison. The resulting F_1 is 0.991, indicating well-defined guidelines and a clear task definition.

3 Setup

Segmentation Methods. We adopt (1) BPE (Shibata et al., 1999), which merges frequent character pairs into subwords until a fixed vocabulary size is reached; (2) WordPiece (Sennrich et al., 2016), which iteratively builds subwords based on likelihood, optimizing for unseen words; (3) Unigram (Kudo, 2018), which applies a probabilistic model to select the best subword units from an initial large set; and (4) Morfessor (Virpioja et al., 2013), which uses methods for unsupervised learning to perform morphological segmentation. We compare these segmenters to the Leave-As-Is (LAI) baseline, which simply returns the word unchanged.

Raw Text Data. For training the segmenters and the LMs, we use raw text data. We collect data from 8 different resources (Table 2). We filter the

believe that due to the diversity of selected words and the relatively morphological simplicity of Danish, the variety of phenomena within each category is well-represented in our data.

DATASET	DOMAIN	SOURCE
Bookshop	Books	Tiedemann (2012)
CC-100	Webscrape	Wenzek et al. (2020)
CulturaX	Webscrape	Nguyen et al. (2023)
Gigaword	Mixed	Strömberg-Derczynski et al. (2021)
OpenSubtitles ⁵	Subtitles	Lison and Tiedemann (2016)
Reddit	Social	Chang et al. (2020)
Twitter	Social	archive.org/details/twitterstream
Wiki	Wiki	Attardi (2015)

Table 2: List of datasets. From the multi-lingual datasets, we only consider the Danish part.

data using the FastText language classifier (Joulin et al., 2017)³ and shuffle the lines before taking the first 40M characters from each source. With these, we create two multi-domain datasets of 40M and 320M characters respectively by evenly mixing the 8 individual datasets.

Language model evaluation Due to computational constraints, we choose to train a model with the same architecture as TinyBERT (Jiao et al., 2020). We did a hyperparameter search with its default tokenizer on the English data from the BabyLM challenge (Warstadt et al., 2023) to find reasonable settings (details are available in the repository).⁴ We use the Adam optimizer, with a learning rate of 1×10^{-3} , a batch size of 512, and 1 epoch over the mixed 320M dataset (Section 2), of which we keep 1% separate for evaluation.

We use a 15% masking strategy during training and evaluation, because perplexity is affected by the segmentation. We use Bits Per Character (BPC) to evaluate the language models. Bits per character represents the average number of bits needed to encode each character in the dataset. Furthermore, we use accuracy on the token level. Even though the accuracy is affected by the segmentation, it is highly interpretable, and since none of our models is tuned to optimize on this metric we expect it to correlate to language model quality.

4 Results

4.1 Morphological Segmentation.

Although there is a variety of metrics available for evaluating morphological segmentation (Virpioja et al., 2011), we opt for the interpretable precision, recall, and F_1 score based on found morphemes (not split points). We start with finding the best

³We keep all text with a confidence above .6 for Danish.

⁴We did this on English, as there is more consensus on which tokenizer/data to use.

⁵<http://www.opensubtitles.org/>

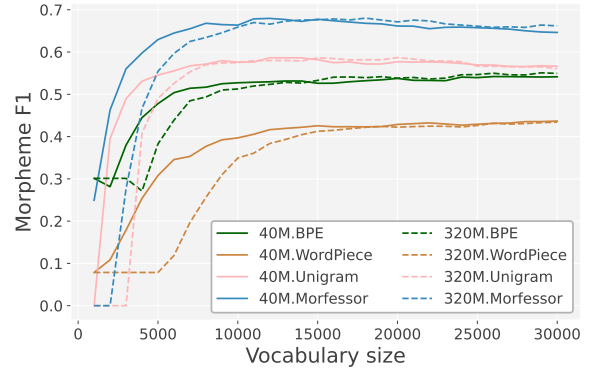


Figure 2: F_1 of each algorithm for different vocabulary sizes for the multi-domain dataset.

vocabulary size of each segmenter on the mixed datasets, as it has the broadest coverage, and then we compare the effect of training on each individual data source.

We evaluate a vocabulary size of 1K-30K subwords with intervals of 1K (Figure 2). Results show that performance for all algorithms follows a similar trend; performance improves strongly in the beginning (i.e., small vocabulary size), until a size of around 10K, after which performance remains in a similar range. For Morfessor and Unigram performance slowly drops, while for BPE and WordPiece it remains rather stable. Morfessor outperforms the other segmenters by a large margin, scoring a maximum F_1 of 67.96, showing that the task is still far from unsolved.⁶ The segmenters trained on 320M characters often perform slightly worse compared to the 40M character training data (especially for smaller vocabulary sizes). In the following sections, we use 40M characters for training segmenters, and use the best vocabulary size for each method: BPE 26K, WordPiece 30K, Unigram 11K, Morfessor 12K.

Next, we compare the effect of the data source on the performance of the segmenters (Figure 3). Results show that while the mixed dataset leads to robust performance across segmenters, different segmenters have different best-performing datasets. As MORSED is composed of well-formed, general-domain words, we would expect that corpora that resemble this (i.e., books, subtitles, wiki, subtitles corpora) would lead to better performance. This trend is loosely reflected in the scores, as the Twitter and Reddit dataset per-

⁶It should be noted that higher scores can be obtained in (partially) supervised settings (Kohonen et al., 2010).

MODEL	Root	Comp.	Link.	Pref.	MORSED		Prec.	Rec.	F1	Acc.	MELFO F1	Lang. Modeling	
					Suff.	Infl.						↓BPC	Acc.
TinyBERT	48.40	16.64	7.76	20.32	29.43	15.12	27.60	29.27	28.41	14.00	11.74	9.84	3.12
LAI	100.00	0.66	15.83	4.42	1.12	12.10	23.33	57.45	33.18	32.25	3.68		
BPE	90.42	45.85	24.45	30.93	10.80	9.37	47.91	62.39	54.20	46.50	25.79	5.25	4.11
WordPiece	83.23	23.93	9.85	13.96	8.94	8.35	38.88	49.81	43.67	26.00	12.87	3.62	27.37
Unigram	82.37	54.82	46.20	39.65	17.29	21.16	53.02	63.13	57.63	46.12	35.20	5.96	5.41
Morfessor	87.93	68.41	50.09	56.86	22.40	44.03	65.00	71.20	67.96	59.75	44.06	6.98	54.04

Table 3: Metrics for Language Modeling and Morphological Segmentation. For the language modeling experiments, we show BPC and accuracy (Acc.). For the morphological segmentation experiments on MORSED, we show performance in F_1 on Root morphemes (Root), Compounds (Comp.), Linking elements (Link.), Prefixes (Pref.), Suffixes (Suff.), Inflections (Infl.) and average performance over the whole dataset: Precision (Prec.), Recall (Rec.), F1 on morphemes, Accuracy (Acc.) on the word level.

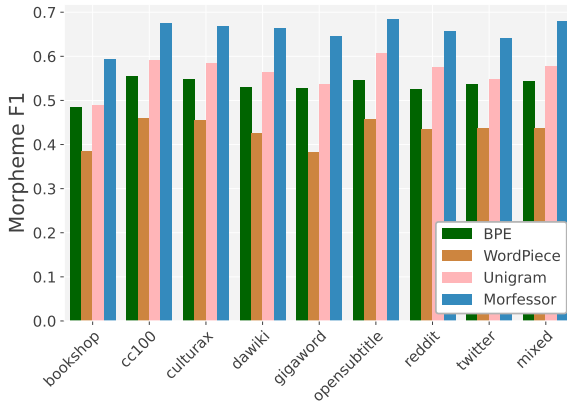


Figure 3: Comparison of the effect of data source, all with 40M characters, and the best vocabulary size for each algorithm.

form relatively poor. However, the Bookshop and FTSpeech also leads to quite low performance, which is probably due to topic bias, FTSpeech contains parlemental data and Bookshop contains quite some technical data (e.g., legal and political topics), which leads to a larger coverage of domain-specific words, but lower performance on MORSED.

4.2 Language Modeling.

For each segmentation algorithm, we used the segmenter trained on the mixed dataset (40M) with the best size from the morphological segmentation results (Section 4) for evaluation on language modeling (Table 3, Language Modeling column). The BPC scores of the Danish tokenizers outperform the original TinyBERT tokenizer (9.2) trained on the Danish corpus. Across the Danish tokenizers, the BPC scores show minimal variance, with the WordPiece tokenizer achieving the best score of 3.62. Morfessor shows a higher BPC

score than the other tokenizers (6.98). We hypothesize that, since BPC correlates directly with cross-entropy, Morfessor’s more granular “sub-word” units (morphemes) lead to less probability mass being concentrated on the most likely token. This results in higher entropy, as the model distributes the probability mass across a larger set of possible tokens, reducing certainty in its predictions. Manual inspection of the output distributions revealed that the Morfessor based language model more often has the correct candidate ranked high, but its confidence scores are less well aligned (i.e. more often scores ≈ 0.5 for incorrect predictions, and lower scores for the best candidate when it is correct). Therefore, we also calculate the subword (i.e. morpheme) accuracy, where only the highest ranking candidate is used. Our results show that the Morfessor tokenizer achieves the highest accuracy by a large margin, indicating that it performs best among all models.

5 Analysis

Quantitative. Our results show that recall is higher than precision for all methods (Table 3). This indicates that most models under-segment. The difference between accuracy and F_1 score (between 6-8 absolute points) shows that there are cases where a word is segmented partially correct.

Models perform especially well on root morphemes, which are not segmented in our task definition (Section 2). A clear trend is that Morfessor and Unigram underperform on root morphemes, but perform better on the other categories. This is because of their smaller optimal vocabulary size (12,000 and 11,000 versus 26,000 for BPE and 30,000 for WordPiece), which leads to oversplitting on the root morphemes. Overall, Morfessor outperforms all other segmenters on all classes ex-

cept root morphemes and suffixes. For the latter, TinyBERT performs better on some word-endings that overlap with English (e.g. ‘-er’, ‘-ing’), which are kept attached to the words by Morfessor.

Qualitative. To get a more fine-grained picture of the difficulties for the segmentation models, we spot-check cases where at least three of the segmenters were incorrect. Our analysis reveals that tokenizers frequently missegment in the categories *compounds* and *compounds with linking elements*. The segmentation of morphemes such as “-e” and “-s” is especially challenging, underscoring tokenizers’ difficulties with complex morphological structures such as “sygeplejeskole” (syg-e-plej-eskole; en: “nursing school”), “gulerod” (gul-e-rod; en: “carrot”) and “landsholdstrup” (land-s-hold-s-trup; en: “national team”). Furthermore, as morpheme length increases, the error rate increases, highlighting the tokenizers’ limitations in handling more complex word formations.

MELFO data After our experiments, we managed to get access to morphological segmentation data from the MELFO (Mobil e-læring for ordblinde) project⁷. This data is not publicly available, but we used it to evaluate the robustness of each segmenter on another dataset with different guidelines and annotators. Upon manual inspection, we found that the main difference between the datasets is the choice of words (there are 8 overlapping words) and that the segmentation of MORSED leads to more splits and smaller elements (e.g. fri-tid-s-hjem versus fritid-s-hjem). The results show a similar trend (i.e. ranking of models), but lower performances overall, which is partially due to tuning (of vocabulary size) on MORSED, but also due to the structure of the data: MELFO has a longer average word length (12 characters versus 8) and a larger average amount of morphemes per word (2.6 versus 1.9).

6 Conclusion

We introduced MORSED, a broad-coverage, expert-annotated dataset for subword segmentation in Danish. We used MORSED to show that an unsupervised segmenter outperforms statistical-based subword segmenters on the task of morphological segmentation for Danish by 10.3 points absolute F_1 score on our novel Danish benchmark.

⁷https://laes.hum.ku.dk/centerets_forskning/melfo/

We also show that the tokenizer that performs best at morphological segmentation also performs well on language modeling (accuracy).

Acknowledgments

We would like to thank Arzu Burcu Güven for her feedback. We thank Bart Jongejan for sharing the MELFO data. We acknowledge the IT University of Copenhagen HPC resources made available for conducting the research reported in this paper. Mike Zhang is supported by a research grant (VIL57392) from VILLUM FONDEN. Elisa Bassignana is supported by a research grant (VIL59826) from VILLUM FONDEN.

References

- Giuseppe Attardi. 2015. Wikiextractor. <https://github.com/attardi/wikiextractor>.
- Terra Blevins and Luke Zettlemoyer. 2019. Better character language modeling through morphology. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1606–1613, Florence, Italy. Association for Computational Linguistics.
- Dwight L Bolinger. 1948. On defining the morpheme. *Word*, 4(1):18–23.
- Kaj Bostrom and Greg Durrett. 2020. Byte pair encoding is suboptimal for language model pretraining. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4617–4624, Online. Association for Computational Linguistics.
- Jonathan P. Chang, Caleb Chiam, Liye Fu, Andrew Wang, Justine Zhang, and Cristian Danescu-Niculescu-Mizil. 2020. ConvoKit: A toolkit for the analysis of conversations. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 57–60, 1st virtual meeting. Association for Computational Linguistics.
- Ann Clifton and Anoop Sarkar. 2011. Combining morpheme-based machine translation with post-processing morpheme prediction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 32–42, Portland, Oregon, USA. Association for Computational Linguistics.
- Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Arya D. McCarthy, Katharina Kann, Sabrina J. Mielke, Garrett Nicolai, Miikka Silfverberg, David Yarowsky, Jason Eisner, and Mans Hulden. 2018. The CoNLL–SIGMORPHON 2018 shared task: Universal morphological inflection. In *Proceedings of the*

- CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection, pages 1–27, Brussels. 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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Valentin Hofmann, Janet Pierrehumbert, and Hinrich Schütze. 2021. Superbizarre is not superb: Derivational morphology improves BERT’s interpretation of complex words. 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 3594–3608, Online. Association for Computational Linguistics.
- Anette Jensen. 2021. *Morfemer*. Gyldendal Uddannelsen.
- Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. TinyBERT: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4163–4174, Online. Association for Computational Linguistics.
- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.
- Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.
- Oskar Kohonen, Sami Virpioja, and Krista Lagus. 2010. Semi-supervised learning of concatenative morphology. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 78–86, Uppsala, Sweden. Association for Computational Linguistics.
- Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.
- Tomasz Limisiewicz, Terra Blevins, Hila Gonen, Orevaghene Ahia, and Luke Zettlemoyer. 2024. MYTE: Morphology-driven byte encoding for better and fairer multilingual language modeling. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15059–15076, Bangkok, Thailand. Association for Computational Linguistics.
- Pierre Lison and Jörg Tiedemann. 2016. OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).
- Austin Matthews, Graham Neubig, and Chris Dyer. 2018. Using morphological knowledge in open-vocabulary neural language models. 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 1435–1445, New Orleans, Louisiana. Association for Computational Linguistics.
- 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 preprint*, abs/2309.09400.
- Eugene A Nida. 1948. The identification of morphemes. *Language*, 24(4):414–441.
- Maja Popović. 2012. Morpheme- and POS-based IBM1 and language model scores for translation quality estimation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 133–137, Montréal, Canada. Association for Computational Linguistics.
- Lane Schwartz, Francis Tyers, Lori Levin, Christo Kirov, Patrick Littell, Chi-kiu Lo, Emily Prud’hommeaux, Hyunji Hayley Park, Kenneth Steimel, Rebecca Knowles, et al. 2020. Neural polysynthetic language modelling. *ArXiv preprint*, abs/2005.05477.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.
- Yusuxke Shibata, Takuya Kida, Shuichi Fukamachi, Masayuki Takeda, Ayumi Shinohara, Takeshi Shinohara, and Setsuo Arikawa. 1999. Byte pair encoding: A text compression scheme that accelerates pattern matching. *Technical Report DOI-TR-161, Department of Informatics, Kyushu University*.

- John Sinclair. 1996. The search for units of meaning. *Textus*, 9(1):75–106.
- Leon Strömberg-Derczynski, Manuel Ciosici, Rebekah Baglini, Morten H. Christiansen, Jacob Aarup Dalsgaard, Riccardo Fusaroli, Peter Juel Henriksen, Rasmus Hvingelby, Andreas Kirkedal, Alex Speed Kjeldsen, Claus Ladefoged, Finn Årup Nielsen, Jens Madsen, Malte Lau Petersen, Jonathan Hvithamar Rystrom, and Daniel Varab. 2021. The Danish Gigaword corpus. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 413–421, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).
- Ahmet Üstün, Murathan Kurfalı, and Burcu Can. 2018. Characters or morphemes: How to represent words? In *Proceedings of the Third Workshop on Representation Learning for NLP*, pages 144–153, Melbourne, Australia. Association for Computational Linguistics.
- Sami Virpioja, Peter Smit, Stig-Arne Grönroos, Mikko Kurimo, et al. 2013. Morfessor 2.0: Python implementation and extensions for morfessor baseline. *Aalto University publication series SCIENCE + TECHNOLOGY*.
- Sami Virpioja, Ville T. Turunen, Sebastian Spiegler, Oskar Kohonen, and Mikko Kurimo. 2011. Empirical comparison of evaluation methods for unsupervised learning of morphology. *Traitement Automatique des Langues*, 52(2):45–90.
- Alex Warstadt, Aaron Mueller, Leshem Choshen, Ethan Wilcox, Chengxu Zhuang, Juan Ciro, Rafael Mosquera, Bhargavi Paranjabe, Adina Williams, Tal Linzen, and Ryan Cotterell. 2023. Findings of the BabyLM challenge: Sample-efficient pretraining on developmentally plausible corpora. In *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 1–34, Singapore. Association for Computational Linguistics.
- Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.
- Daniel Zeman, Jan Hajič, Martin Popel, Martin Potthast, Milan Straka, Filip Ginter, Joakim Nivre, and Slav Petrov. 2018. CoNLL 2018 shared task: Multilingual parsing from raw text to Universal Dependencies. In *Proceedings of the CoNLL 2018 Shared*
- Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–21, Brussels, Belgium. Association for Computational Linguistics.

Opinion Units: Concise and Contextualized Representations for Aspect-Based Sentiment Analysis

Emil Häglund

Department of Computing Science
Umeå University, Sweden
emilh@cs.umu.se

Johanna Björklund

Department of Computing Science
Umeå University, Sweden
johanna@cs.umu.se

Abstract

We introduce *opinion units*, a contribution to the field Aspect-Based Sentiment Analysis (ABSA) that extends aspect-sentiment pairs by including substantiating excerpts. The goal is to provide fine-grained information without sacrificing succinctness and abstraction. Evaluations on review datasets demonstrate that large language models (LLMs) can accurately extract opinion units through few-shot learning. The main types of errors are providing incomplete contexts for opinions and mischaracterising objective statements as opinions. The method reduces the need for labelled data and allows the LLM to dynamically define aspect types. As a practical evaluation, we present a case study on similarity search across academic datasets and public review data. The results indicate that searches leveraging opinion units are more successful than those relying on traditional data-segmentation strategies, showing robustness across datasets and embeddings.

1 Introduction

We propose *opinion units* as a representation for subjective viewpoints in text. An opinion unit consists of (i) an aspect such as price, quality, or location, (ii) an excerpt, which may be lightly paraphrased to only include relevant text, that contextualises the opinion (iii) and a sentiment such as positive, negative or neutral. The structured nature of opinion units makes them suitable for applications requiring fine-grained *aspect-based sentiment analysis* (ABSA), such as the mining and retrieval of opinions. ABSA goes beyond the surface level of traditional sentiment analysis. Instead of assigning a sentiment to an entire text, ABSA identifies opinions expressed about particular features

of, for instance, a product, service or event. This multi-faceted analysis provides valuable insights for those seeking to understanding public opinion on a particular topic. For example, for retailers, ABSA of customer reviews or interactions can suggest areas for improvement, personalise marketing strategies, and gauge overall customer satisfaction.

Initial ABSA research focused on classifying reviews into predefined aspect- and sentiment categories (Zhang et al., 2022). Over time, this came to include the extraction of aspect- and sentiment keywords (Zhang et al., 2022; Gao et al., 2021). While the reduction of a text to keywords is helpful for many applications, it also lead to information loss. In contrast, opinion units offer a structured representation that retains more of the original nuance. The emergence of generative LLMs, with their capacity for longer sequence-to-sequence outputs, enable the flexible extraction of phrases required for creating opinion units. For concisely expressed opinions, as in the short-sentence examples used in keyword-extraction benchmarks like SEMEVAL Res-15 & 16 (Pontiki et al., 2016), opinion units closely resembles keyword extraction. However, in real-world reviews, customer opinions often involve descriptions and motivations spanning longer passages. Phrase extraction provides a more natural and expressive method for capturing these nuanced opinions. For instance keyword extraction would overlook subtlety in a sentence like: “The outdoor area is delightful, especially in the evening, with its soft lighting and comfortable chairs creating a cozy atmosphere”. Moreover, phrases provide better interpretability for end users, allowing them to identify which sections of the raw text influenced the decisions made by downstream applications.

The extraction of opinion units can serve as a standalone chunking strategy for applications requiring detailed information. However, it can also be made as preprocessing step before keyword extraction (Siddiqi and Sharan, 2015), because the

Last Sunday we went to **brunch** and I had a **muffin**. It was **amazing**! We loved our waiter Stephanie she was so **friendly** however the service **could have been a little quicker**. But on the whole, we had a **great time**!

- **Muffin**: I had a muffin. It was **amazing**. {positive}
- **Staff friendliness**: We loved our waiter Stephanie, she was so **friendly**. {positive}
- **Service speed**: The service **could have been a little quicker**. {negative}
- **Overall brunch experience**: On the whole, we had a **great time**. {positive}

Figure 1: Four opinion units extracted from a review, each representing an opinion in the text and consisting of an aspect label, an excerpt from the text, and a sentiment label. The colour purple indicates aspects, and orange indicates sentiment terms.

atomic nature of opinion units—each representing a single opinion about one aspect—simplifies analysis. This is advantageous compared to analysing “raw text”, which often contains intertwined opinions and unrelated non-opinionated content.

In this article, we explore how opinion units can be extracted from subjective commentary, specifically customer reviews, by large language models (LLMs). The models are prompted in a way that allows them to dynamically generate aspect categories not explicitly mentioned in the text, and to choose and paraphrase motivating text excerpts that retain only the most relevant information. An example of how opinion units are formed is given in Figure 1 and a formal definition is provided in Section 3. The main benefit opinion units is that they provide a structured representation of the opinions expressed in a text, while retaining much of the nuance through the supportive excerpt.

Language models excel at many of the tasks involved in the generation of opinion units, including information extraction, text summarization, entity recognition, and sentiment analysis. Previous work has successfully applied LLMs to extract *propositions*, that is, atomic factual statements, to facilitate question answering in a dense retrieval setting where both the query and documents are transformed into embeddings (Chen et al., 2024). We transfer this method to the ABSA domain, demonstrating that LLMs can effectively identify opinion aspects, extract concise snippets of text expressing the opinion, and accurately classify the sentiment of the excerpt. An important advantage of extracting opinion units with LLMs stems from the few-shot approach. Unlike traditional ABSA methods

that often rely on pre-defined categories or require labeled training data, LLMs can extract opinion units without such constraints. This opens doors for broader application across diverse domains and allows for more efficient and scalable analysis.

In the following sections, we first investigate the ability of LLMs in generating opinion units by evaluating GPT-4-turbo, GPT-3.5-turbo, and Llama2-70B. This evaluation is conducted on subsets of SEMEVAL restaurant review sentence dataset (Pontiki et al., 2016) as well as a Yelp dataset (Yelp, 2015) containing complete restaurant reviews. Furthermore, we categorize the errors produced by the LLMs, where providing incomplete context, missing aspects and the conflation of objective statements with opinions turn out to be the most serious sources of error. Finally, we demonstrate the effectiveness of opinion units in dense similarity search, where words are represented by embeddings. In particular, we show that opinion units outperform the competing chunking strategies of sentence and passage chunking. These positive results suggest that opinion units are potentially useful also for dense retrieval, retrieval-augmented generation and clustering applications. For example, in topic modeling, opinion units can reveal which topics customers discuss in reviews.

The experiments conducted in this article serve to answer the following research questions:

- RQ1.** To what extent can LLMs extract accurate opinion units?
- RQ2.** What are the types and frequencies of errors made by the LLMs in this process?
- RQ3.** How does the performance of opinion units in dense similarity search for opinions compare to other data-segmentation strategies?

2 Related Work

This section recalls related work on ABSA, summarisation, and information retrieval.

2.1 Aspect-Based Sentiment Analysis

Aspect-based sentiment analysis is a specialized area within the broader field of sentiment analysis. Its focus is on identifying and extracting sentiment in relation to specific aspects in a given text (Zhang et al., 2022). The analysis typically involves establishing some or all of the following sentiment elements: The aspect category c which is the general concept to which the sentiment pertains; the aspect term a which is the entity being referred

to; the opinion term o which conveys the aspect sentiment; and the sentiment polarity p which is the valence of the emotion expressed (Zhang et al., 2022). Given the sentence “the tiramisu was amazing”, these elements could be mapped accordingly: c = ‘dessert’, a = ‘tiramisu’, o = ‘amazing’, and p = ‘positive’. We note that the construction of opinion units involves all four sentiment elements: The opinion label corresponds to the aspect category, although in our case it is generated on the fly by the LLM rather than chosen from a set of predefined categories. The excerpt in opinion units includes both aspect and opinion terms. Finally, each opinion unit includes a sentiment polarity.

Earlier works concentrated on solutions for isolated sentiment elements, such as aspect term extraction (Liu et al., 2015; Li and Lam, 2017) or aspect category detection (Zhou et al., 2015; Luo et al., 2019). Later studies extract several factors at once, capturing both the opinion aspect and expression (Peng et al., 2020; Gao et al., 2021). The main challenge in these tasks is the accurate pairing of aspect-sentiment elements (Zhang et al., 2022).

We are now seeing significant advancements in the implementation of multifaceted analysis tasks. A salient example is sequence-to-sequence models which output the result of the analysis as a natural-language statement. This approach has been shown to outperform classification methods and exhibits particular strengths in scenarios with limited training data thanks to few-shot and zero-shot learning (Ma et al., 2019; Zhang et al., 2022).

2.2 Summarisation

Opinion mining benefits from both extractive and abstractive summarization (Anand Babu and Badugu, 2023). The former produces a summarisation by concatenating informative segments from the source document, whereas the latter generates a summary based on the semantics of the source, which at a superficial level can be very different from the original text. Extractive summarisation is needed because it provides evidence in the source material for the generated opinion units (Priya and Umamaheswari, 2020), but to keep the excerpts short and self-contained, a degree of abstractive summarisation is also necessary.

Yang et al. (2019) evaluate ChatGPT on abstractive summarization. Even with a zero-shot approach, the model performs on par with smaller LMs fine-tuned for the task. This stands in con-

trast to the case for aspect-based sentiment analysis discussed above, where the smaller, fine-tuned models were more successful (Zhang et al., 2023). A related task is key-point extraction (Bar-Haim et al., 2020a,b, 2021), where the objective is to extract salient viewpoints from a text. Also here LLM-enabled aspect-based approaches have been successfully applied (Tang et al., 2024) and reduce the number of partially overlapping key points.

2.3 Information Retrieval

Dense retrievers are a common type of modern retrieval systems where a dual-encoder architecture transforms documents and queries into dense embeddings for similarity comparison (Ni et al., 2022). These similarity functions, also used for embedding-based clustering (Chandrasekaran and Mago, 2021), have limitations in understanding complex semantics and can be misled by irrelevant information (Chen et al., 2024). Chen et al. (2024) explored using propositions, factual statements distilled from text using LLMs (GPT-4), as retrieval units for Wikipedia passage retrieval and retrieval-augmented LLM question answering. Using propositions to segment and index the retrieval corpus outperformed traditional methods like sentence or fixed-length passage chunking. In their context of fact retrieval, each proposition represented a single atomic fact with relevant context, phrased concisely in natural language (Chen et al., 2024). Corpus segmentation using propositions is described as an orthogonal strategy that can be used in conjunction with other methods for improving dense retrieval such as supervised retrievers (Chen et al., 2024), data augmentation (Wang et al., 2022) or mixed-strategy retrieval (Ma et al., 2023).

Propositions offers a high information density with complete context. Comparatively, passage chunking constitutes a coarse information unit, often containing unrelated and multiple aspects. This lack of conciseness can distract downstream applications such as retrieval relying on similarity comparison (Yu et al., 2023). Sentence chunking provides more fine-grained information. However, sentences can include multiple aspect and lack necessary context when dependencies span multiple sentences (Yang et al., 2019).

3 Opinion units

As stated in Section 1, an opinion unit is composed of three elements: i) an aspect label, ii) a text ex-

Challenge	Example of review and extracted opinion units	Benefits of opinion units
Passages expressing multiple opinions	<i>The food is great but the drinks sucked.</i> ► Food: The food is great {positive} ► Drinks: The drinks sucked {negative}	Unlike passage and sentence chunking, opinion units separate aspects which avoids noisy and non-concise segments.
Opinions spanning multiple sentences	<i>We had margaritas. They tasted absolutely wonderful!</i> ► Margaritas: We had margaritas. They tasted absolutely wonderful . {positive}	Opinion units provide full context spanning several sentence. Sentence chunking provides incomplete context and passage chunking could be incomplete or include noise, depending on the length of the relevant passage.
Lack of contextual information	<i>The restroom was not ADA compliant.</i> ► Disabled persons accessibility: The restroom was not ADA compliant . {negative}	The opinion label generated by the LLM provides helpful context for later processing steps. In the example, ADA stands for Americans with Disabilities Act which ensures equal access for people with disabilities.
Insufficient sentiment understanding and filtering	<i>The portion size was perfect... for an ant.</i> ► Portion size: The portion size was perfect... for an ant . {negative}	LLMs are more adept at understanding sentiments or irony compared to word embeddings at inference time. Opinion units can be filtered by sentiment.

Figure 2: Examples and summary of four challenges when segmenting opinionated texts for downstream applications where opinion units provide advantages compared to passage- and sentence chunking.

cerpt substantiating a subjective viewpoint on the aspect, and iii) a sentiment label that quantifies the sentiment expressed according to some set scale. Additionally, we outline four key principles that together characterize opinion units. These are inspired by the factual propositions of Chen et al. (2024) described in Section 2.3, but are tailored for the ABSA domain. The principles are as follows:

Atomicity. Every opinion unit should represent exactly one opinion (i.e., aspect-sentiment pair).

Injectivity. No two opinion units should represent the same opinion.

Completeness. Collectively, the set of extracted opinion units should encompass all the opinions expressed in the text.

Contextuality. The excerpt associated with each opinion unit should give sufficient contextual information to motivate the inferred sentiment. If needed, the excerpt may refer to other aspects or sentiments.

When used for data segmentation in applications such as customer-satisfaction surveys or brand studies, LLM-enabled generation of opinion unit overcomes a number of challenges (see Figure 2). First of all, opinion units can handle sentences and passages with multiple opinions, and as well as opinions spanning multiple sentences. In these cases, traditional segmentation strategies such as sentence and passage chunking (which we benchmark against in Section 4), create irrelevant or uninformative chunks. Opinion units, in contrast, isolate opinions and adapt the excerpt length to match the coverage of the aspect in the source text.

Another benefit is that the aspect label gener-

ated by the LLM facilitate the clustering of opinion units that refer to the same concept, even though the terms and wording used in the source text may vary. Similarly, the sentiment label can be used to filter opinion units based on sentiment polarity. This approach leverages the LLM’s high performance in sentiment analysis (Zhang et al., 2023) while ensuring efficient inference (see Section 5.2). Incorporating other metadata than sentiment, or a finer sentiment scale would also be possible and could be beneficial for specific applications. For chunking strategies like passage- or sentence chunking, the presence of multiple opinions or non-opinionated text within a single chunk can make sentiment labeling less straightforward and precise.

Finally, the LLM can be prompted to disregard sections of the source text that do not express opinions, which is valuable because also subjectively written texts can have strictly objective passages. For example, in the context of restaurant reviews, as statement such as “I went with my two friends and sat in a corner booth” may not have much bearing on the writer’s assessment of the food. In passage- or sentiment chunking, these non-opinionated texts cannot be avoided and add noise to the analysis process.

4 Method

The experimental evaluation of opinion units comprises two parts. First, we evaluate the performance of three LLMs (GPT-4 turbo, GPT-3.5 turbo, and Llama2-70B) in generating well-formed opinion units. Second, we perform a case study on opinion retrieval, comparing data segmentation based on opinion units to traditional chunking strategies.

4.1 Generation of Opinion Units

We generate opinion units using LLMs in a few-shot approach. The prompt template, provided in Figure 3, instructs the LLM to perform ABSA, extracting the three components of an opinion unit. An example review with opinion units is provided in the template. The examples are designed to address issues discussed in Section 3, such as non-opinionated text and opinions spanning multiple sentences. If the generated opinion units deviate from the format defined in the prompt template—for instance, by producing an incorrect JSON object—the generation is repeated (this happens approximately 5% of the time). For all LLMs we use a temperature of 1.0.

Perform aspect-based sentiment analysis for the restaurant review provided as the input. Return each aspect-sentiment pair with a label and a corresponding excerpt from the text. Also mark the sentiment of aspects as negative or positive.

Aspect-sentiment pairs should not mix opinions on different aspects. Make sure to include all aspects. An aspect should be independent and not have to rely on other aspects to be understood.

If an opinion in the review is about the restaurant or experience in general then label this aspect as "overall experience". Opinions not related to the restaurant should not be included.

Example input: I just left Mary's with my lovely wife. The gorgeous outdoor patio seating was fantastic with a nice view of the ocean. We came for brunch and were blown away! We split dozen oysters. They were the best I had in my life! FRESH! Delicious! The avocado toast was excellent as were the crab cakes. Altogether, we had a great experience. Almost 5 stars! but the staff could have been a little friendlier and the tables cleaner.

Example output:

[["Outdoor patio seating", "The gorgeous outdoor patio seating was fantastic with a nice view of the ocean", "positive"],
["View", "a nice view of the ocean", "positive"],
["Brunch", "We came for brunch and were blown away", "positive"],
["Oysters", "We split a dozen oysters. They were the best I had in my life! FRESH! Delicious!", "positive"],
["Avocado toast", "the avocado toast was excellent", "positive"],
["Crab cakes", "the crab cakes were excellent", "positive"],
["Overall experience", "Altogether, we had a great experience. Almost 5 stars!", "positive"],
["Staff friendliness", "the staff could have been a little friendlier", "negative"],
["Table cleanliness", "the tables could have been cleaner", "negative"]]

Input: Review to be processed
Output:

Figure 3: Prompt template: opinion unit generation

4.2 Opinion Unit Evaluation

To assess the correctness of the generated opinion units, we conduct evaluations on subsets of SEMEVAL Res15 and Res16, which consist of restaurant-review sentences (Pontiki et al., 2016), as well as full Yelp restaurant reviews (Yelp, 2015). We compare the performance of GPT-3.5-turbo, GPT-4-turbo and Llama2-70B. For these subsets, we created solution keys of correct opinion units by manually identifying aspects and their sentiments in each text. For the SEMEVAL subset, sentiment labels followed the ASTE annotations provided by (Zhang et al., 2021). In the solution keys, we selected approved LLM-generated aspect labels and excerpts to serve as examples of correct opinion unit components. For the SEMEVAL subset we

select reviews from the Res15 and Res16 test sets that, according to (Zhang et al., 2021)’s annotations, include multiple aspects. The subset used for SEMEVAL evaluation consists of 565 opinion units in the solution key, stemming from 238 review sentences. A similar size subset was randomly subsampled from the Yelp dataset, constituting 505 opinion units from 96 reviews.

We evaluate opinion units according to the principles outlined in Section 3. These principles include, ensuring that each unit reflects a single opinion, provides enough context to motivate its sentiment and that the sentiment classification and identified aspects align with the solution key. We classify errors into the categories listed below; an opinion unit is considered correct only if it avoids all these errors. The evaluation was conducted by the two authors and was not blind to which LLM generated the opinion units. Disagreements that arose during the evaluation were revisited and resolved through careful re-examination in accordance with the established error and evaluation guidelines.

Atomicity error. An opinion unit lacks *atomicity*, providing context for multiple opinions.

Injectivity error. Collectively, opinion units are redundant, lacking *injectivity*.

Missing aspect. Collectively, the opinion units lack *completeness*, meaning that not all opinions in the review were captured.

Missing context. An opinion unit is not *contextualized*, i.e., does not provide sufficient contextual information to motivate the inferred sentiment.

Non-opinion. A non-opinionated excerpt from the text is incorrectly classified as an opinion.

Sentiment error. The sentiment label is incorrect.

Aspect-label error. The aspect label does not adequately describe the opinion.

Hallucination. The LLM invents aspects or excerpts that are not part of the review.

To quantify the results, we use three metrics: Precision, the ratio of correct generated units to total generated units; recall, the ratio of correct generated units to total opinion units in the solution key; and F1-score, the harmonic mean of precision and recall. In the scoring, certain cases were handled with special consideration. For the short SEMEVAL reviews, the LLMs in addition to individual aspects, sometimes created instances of “overall experience” which combined multiple aspects as a characterization of the overall experience.

When considered reasonable reflections of overall sentiment, these were excluded from scoring and did not impact the precision and recall values.

Our evaluation inherently involves a degree of subjectivity. For example, differing human assessments may arise about whether an extracted phrase provides full context or if an aspect label is descriptive enough to capture the opinion. This subjectivity, though typical for many NLP annotations (Röttger et al., 2021) and perhaps especially for unstructured generative LLM outputs, makes the evaluation unsuitable as a strict benchmark, like ABSA benchmarks for classification and keyword extraction (Pontiki et al., 2016). Despite these limitations, we believe this evaluation to be crucial for understanding the performance of opinion unit generation in isolation and not just through its impact on downstream tasks. Additionally, the error classification offers important insights for future work on using LLMs for opinion extraction.

4.3 Case Study: Opinion Retrieval

Whereas the experiment just described tests the viability of LLM-extracted opinion units, the following case study evaluates the method’s usefulness. For this opinion retrieval task, opinion units were generated using GPT-3.5-turbo, selected for its balance of performance (as demonstrated in Section 5.1) and cost-efficiency.

Retrieval Tasks. We designed 50 similarity search tasks for restaurant reviews. The goal of the retrieval system is to return reviews that contain opinions that are similar to the opinion provided as the query. We categorized the 50 tasks into 10 general tasks and 40 detailed tasks. General tasks correspond to common and overarching opinions found in restaurant reviews, such as overall experience, value for money, and staff friendliness. For instance, Task 1 has the query: “All in all, we had a great time.” For returned reviews to be considered correct, they must express satisfaction with the overall experience. Task 4 seeks reviews that highlight staff friendliness, using the query: “The staff were very friendly. Detailed tasks focus on specific aspects mentioned in fewer reviews. For example, the query for Task 24 is: “The food was cold when we received it.” Returned reviews must detail negative experiences related to receiving cold food at the restaurant. Out of the 50 tasks, half entail a positive sentiment, and the remaining a negative sentiment. The returned reviews were assessed by a team of 4

evaluators who were blind to the chunking strategies used. On average, each returned review received 2.3 annotations. Conflicts were resolved through majority voting; in cases of equal votes, an additional evaluator was consulted for final assessment. The reviews were presented in a randomized order to eliminate a potential source of bias. The full list of review tasks, including queries and task descriptions are available online. Implementations of opinion unit generation, retrieval and passage and sentence chunking are also provided¹.

Evaluation Groups. We compare dense retrieval based on opinion units to the conventional approaches of passage- and sentence chunking (Chen et al., 2024). In sentence chunking, each sentence serves as a retrievable unit, whereas in passage chunking, we employ Langchain’s RecursiveCharacterTextSplitter with parameters `size=200` and `overlap=20`. The retrievable units in passage chunking are on average longer (avg. 28.2 words in Yelp dataset) compared to sentence chunking (avg. 12.9 words) and opinion units (14.9 words). In addition to standard opinion units, we also use opinion units with sentiment filtering as a retrieval unit (denoted *opinion + sf* in results tables). In this approach, only opinion units labeled with the specific sentiment demanded by the task are considered by the retrieval system. For each retrieval strategy, we extract 20 unique reviews. Precision @5, 10, and 20 is used to evaluate results by measuring the percentage of relevant reviews among the top k returned reviews for each task.

The primary dataset used for evaluating the opinion retrieval case study is the Yelp dataset (Yelp, 2015), which contains millions of authentic reviews. We refine this dataset to include only restaurant reviews, extracting the first 20 000 reviews of restaurants located in California to serve as our retrieval corpus. As a secondary dataset, we use a concatenation of the SEMEVAL Res15 train and test datasets and the Res16 test dataset (excluding the Res16 train dataset, as it duplicates the Res15 train and test reviews). This dataset is considerably smaller than the Yelp dataset, containing 2 280 reviews. On average, each review spans approximately 14.49 words and 1.75 opinion units. In contrast, the average Yelp review contains 92.7 words and 5.5 opinion units, with the 95th percentile extending to 257 words and 10.0 opinion units. The 50 retrieval tasks are designed to ask for

¹<https://github.com/emilhagl/Opinion-Units>

increasingly specific topics. When evaluating on the SEMEVAL dataset, we omit the 20 most specific tasks (i.e., Task 31–50) because the scope of the dataset is so limited that these fine-grained tasks do not contribute to the evaluation in a meaningful way. For similar reasons we only report Precision @5 and @10 as our evaluation metrics.

To ascertain the robustness of retrieval results we perform the evaluation using two different embedding models from the sentence-transformers framework: `all-mpnet-base-v2` and `all-MiniLM-L6-v2` (Transformers, 2024). Both embedding models are optimized for general tasks, including sentiment analysis, however `all-mpnet-base-v2` is a considerably larger model (80MB vs. 420MB). For our dense retrieval implementation, we used the Faiss package and its function `similarity_search` (Langchain, 2024).

5 Results and Discussion

5.1 Opinion Unit Evaluation

We evaluate the opinion units generated for the SEMEVAL and YELP subsets with respect to the methodology described in Section 4.1. Our analysis reveals that GPT-4-turbo achieves the best performance across datasets (YELP: Precision = 85.3, Recall = 87.4; SEMEVAL: Precision = 89.3, Recall = 92.7). GPT-3.5-turbo shows slightly lower performance (YELP: Precision = 87.0, Recall = 82.2; SEMEVAL: Precision = 87.5, Recall = 89.6), while Llama2 exhibits a more pronounced drop in performance (see Table 1). Notably, recall values are lower for the YELP dataset, where longer reviews result in a greater number of overlooked aspects. Overall, the strong performance of the GPT-models is promising for downstream tasks.

Furthermore, we categorize the errors according to the classification described in Section 4.1, to understand the types of problems the LLMs encounter when generating opinion units. The frequency of these errors is presented in Figure 4. The most common errors are missing context or categorizing non-opinion statements like “we went to sit at the bar” as opinions (see Figure 4). For the Yelp dataset with long text reviews, missing aspects were a frequent error. Although issues like missing context, injectivity, or atomicity are less than ideal in terms of error severity, an opinion unit could still function reasonably well as a retrieval unit. In contrast, missing aspects and the characterizing non-opinions as opinions have a more certain

	Yelp			SEMEVAL		
	P	R	F1	P	R	F1
GPT-4-turbo	85.3	87.4	86.3	89.3	92.7	91.1
GPT-3.5-turbo	87.0	82.2	84.6	87.5	89.6	88.5
Llama2-70B	76.9	74.5	75.7	75.6	88.8	81.6

Table 1: Precision (P), Recall (R) & F1-scores for evaluation on Yelp and SEMEVAL subsets.

negative impact on downstream tasks.

A few hallucinations were identified, primarily produced by Llama2, where the LLM invented an excerpt not present in the review. These mostly occurred when the LLM added an “overall experience” label with an invented excerpt, an artefact of the prompt template’s instructions for “overall experience.”, (see Figure 3).

5.2 Case Study: Opinion Retrieval

In our case study we compare the performance of alternative chunking strategies on 50 different retrieval tasks, each of which consists in retrieving reviews which include some specific opinions (see Section 4.3). The retrieval results, presented in Table 2, delineate the performance across datasets (Yelp and SEMEVAL-Rest) and the two different word embedding models. The larger embedding model, `all-mpnet-base-v2`, leads to better results than the smaller `all-MiniLM-L6-v2`.

Consistently, across all experimental conditions, opinion units outperform passage- and sentence chunking, with sentence chunking being most competitive. This implies that opinions in reviews are often expressed within a single sentence. The results show the benefit of the opinion units ability to provide a concise and structured representation in opinion retrieval. The increased retrieval precision stems from the ability to address challenges highlighted in Section 3 such as passages with intertwined opinions and opinion spanning multiple sentences detailed.

It is worth noting the large performance gap between standard opinion units and opinion units with sentiment filtering (opinion unit + sf). In our evaluation tasks, the objective is to retrieve reviews with certain combinations of aspects and sentiments. Filtering by the LLM-generated sentiment labels thus contributes towards an important subgoal. The resulting gains in precision also highlights the limitations of word embeddings in sentiment comprehension (Yu et al., 2017), where words with similar vector representations can exhibit contrasting senti-

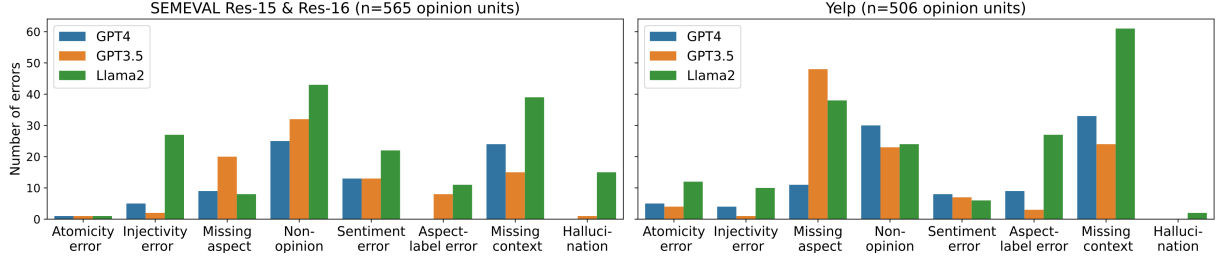


Figure 4: Error type frequency in generated opinion units for SEMEVAL and Yelp subsets.

(a) Yelp Restaurant, all-mpnet-base-v2					(b) Yelp Restaurant, all-MiniLM-L6-v2				
Tasks	Chunking strategy	Precision @5	Precision @10	Precision @20	Tasks	Chunking strategy	Precision @5	Precision @10	Precision @20
All (Task 1-50)	Passage	61.6	54.4	56.0	All (Task 1-50)	Passage	54.4	53.6	49.3
	Sentence	76.4	70.6	63.3	General (Task 1-10)	Sentence	65.6	62.8	54.6
	Opinion unit	81.6	74.4	69.5		Opinion unit	70.8	65.0	61.1
	Opinion unit + sf	88.0	82.2	77.9		Opinion unit + sf	82.0	80.4	76.1
General (Task 1-10)	Passage	78.0	76.0	70.5	Detailed (Task 11-50)	Passage	68.0	68.0	63.5
	Sentence	90.0	86.0	81.5		Sentence	78.0	74.0	70.0
	Opinion unit	94.0	90.0	86.0		Opinion unit	78.0	78.0	76.5
	Opinion unit + sf	96.0	92.0	89.5		Opinion unit + sf	84.0	89.0	88.5
Detailed (Task 11-50)	Passage	57.7	54.0	52.4	(Task 11-50)	Passage	51.0	50.0	45.8
	Sentence	73.0	66.8	58.8		Sentence	62.5	60.0	50.8
	Opinion unit	78.5	70.5	65.4		Opinion unit	69.0	61.7	57.2
	Opinion unit + sf	86.0	79.8	75.0		Opinion unit + sf	81.5	78.2	73.0

(c) SEMEVAL Res15+Res16, all-mpnet-base-v2					(d) SEMEVAL Res15+Res16, all-MiniLM-L6-v2				
Tasks	Chunking strategy	Precision @5	Precision @10		Tasks	Chunking strategy	Precision @5	Precision @10	
All (Task 1-30)	Passage	53.3	41.7		All (Task 1-30)	Passage	46.0	42.3	
	Sentence	53.3	42.0		General (Task 1-10)	Sentence	46.0	42.3	
	Opinion unit	67.3	56.7			Opinion unit	54.7	46.7	
	Opinion unit + sf	74.0	60.3			Opinion unit + sf	72.0	62.3	
General (Task 1-10)	Passage	78.0	63.0		Detailed (Task 11-30)	Passage	58.0	55.0	
	Sentence	78.0	64.0			Sentence	60.0	54.0	
	Opinion unit	80.0	81.0			Opinion unit	68.0	64.0	
	Opinion unit + sf	84.0	85.0			Opinion unit + sf	78.0	77.0	
Detailed (Task 11-30)	Passage	41.0	31.0		(Task 11-30)	Passage	40.0	36.0	
	Sentence	41.0	31.8			Sentence	39.0	36.5	
	Opinion unit	61.0	44.5			Opinion unit	48.0	38.0	
	Opinion unit + sf	69.0	48.0			Opinion unit + sf	69.0	55.0	

Table 2: Precision results for different combinations of dataset and embedding model

ment polarities, e.g., “friendly” and “unfriendly”. Refining word embeddings to better reflect both semantics and sentiment is therefore an important avenue for future work (Yu et al., 2017).

6 Summary and Conclusion

We have presented opinion units as a structured representation for subjective viewpoints, enhancing traditional aspect-sentiment pairs by incorporating substantiating excerpts that retain detailed information. Opinion units can function as an independent chunking strategy for applications that require detailed information or be utilized as a preprocessing step that allows for further abstractions such as category classification or keyword extraction. Our

findings demonstrate the ability of LLMs to accurately extract opinion units from restaurant review datasets. The most frequent errors were insufficient excerpt context and misclassifying non-opinion statements as opinions. Furthermore, a case study showcased the effectiveness of opinion units in opinion retrieval using dense embeddings, outperforming traditional segmentation methods.

The few-shot approach allows the LLM to identify aspects without annotated data or predefined aspect categories. Each opinion unit represents a single opinion, consisting of an aspect label, a text excerpt that provides context, and a sentiment label that conveys the expressed sentiment. These units facilitate downstream applications, e.g., clus-

tering and retrieval. The excerpt generation handles difficulties such as intertwined opinions, where discussions interleave opinions with other topics, and multi-sentence opinions. Furthermore, the sentiment label allows for filtering at inference time, mitigating the issue with word embeddings where words with contrasting sentiment polarities have similar vector representations (Yu et al., 2017).

7 Limitations and Future Work

In this study, we did not fine-tune the LLMs for the opinion unit generation task. While demonstrating that LLMs can perform well on this task without the requiring additional training data is a strength in itself, fine-tuning has the potential to improve accuracy and enable the use of smaller, more efficient models. Exploring the potential improvements in performance through fine-tuning, particularly with regard to specific error, is an intriguing avenue for future research.

Our study implemented a baseline dense retrieval system to isolate the impact of opinion units on retrieval performance. However, we do not demonstrate the effectiveness of opinion units in refined downstream applications. A more refined implementation could integrate various techniques. For instance, sentiment refined word embeddings (Yu et al., 2017), supervised retrievers (Chen et al., 2024), data augmentation (Wang et al., 2022), hybrid sparse-dense retrieval (Luan et al., 2021) or mixed strategy retrieval (Ma et al., 2023). These methods should be synergistic with opinion units, where the segmentation of the retrieval corpus into structured opinion is a separate pre-processing step. Additionally, it would be interesting to cluster opinions based on the corresponding opinion units, to learn how groups of aspects and sentiments correspond to overall ratings or buying decisions, and how the principles of atomicity and contextuality (see Section 3) affect the results.

The next group of limitations stem from the need for a larger labelled ABSA dataset. The current SEMEVAL datasets are restricted not only by the number of reviews, but primarily by the brevity and inauthenticity of these reviews, as they consist of individual sentences rather than complete review texts. A larger annotated dataset would facilitate the evaluation of opinion units with reduced reliance on custom annotation and assessment. Such a dataset should ideally include a significant amount of non-opinionated texts and of

opinions that require multi-hop reasoning to understand, challenges that LLMs are known to struggle with (Chen et al., 2024). Such datasets could serve as a direct benchmark or foundational basis for evaluation.

Another dataset-related limitation is the absence of annotated retrieval datasets specifically for opinion mining. To address this, we designed 50 custom retrieval tasks to simulate opinion retrieval and evaluated the top-ranked reviews returned by these tasks. Annotated datasets, akin to those used in the QA domain (Chen et al., 2024) or TREC challenges (Grossman et al., 2016), contain pre-annotated relevant documents for each task and would facilitate a more comprehensive assessment using recall and F1 metrics. Such datasets would provide a more holistic understanding of retrieval performance, complementing the precision@k-based evaluation we currently employ.

Finally, our evaluation of opinion units as a structure for opinions focused on customer reviews. Other opinionated texts, such as longer political writings, could present additional challenges. These texts may make it more difficult to extract excerpts that contextualize an opinion, and they may require a greater degree of abstractive summarization to accurately capture the context.

References

- G. L. Anand Babu and Srinivasu Badugu. 2023. A survey on automatic text summarisation. In *Proceedings of the Third International Conference on Advances in Computer Engineering and Communication Systems: ICACECS 2022*, pages 679–689. Springer.
- Roy Bar-Haim, Lilach Eden, Roni Friedman, Yoav Kantor, Dan Lahav, and Noam Slonim. 2020a. From arguments to key points: Towards automatic argument summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4029–4039. Association for Computational Linguistics.
- Roy Bar-Haim, Lilach Eden, Yoav Kantor, Roni Friedman, and Noam Slonim. 2021. Every bite is an experience: Key point analysis of business reviews. 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 3376–3386. Association for Computational Linguistics.
- Roy Bar-Haim, Yoav Kantor, Lilach Eden, Roni Friedman, Dan Lahav, and Noam Slonim. 2020b. Quantitative argument summarization and beyond: Cross-domain key point analysis. In *Proceedings of the*

- 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 39–49. Association for Computational Linguistics.
- Dhivya Chandrasekaran and Vijay Mago. 2021. [Evolution of semantic similarity—a survey](#). *ACM Computing Surveys*, 54(2).
- Tong Chen, Hongwei Wang, Sihao Chen, Wenhao Yu, Kaixin Ma, Xinran Zhao, Hongming Zhang, and Dong Yu. 2024. [Dense X retrieval: What retrieval granularity should we use?](#) In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15159–15177, Miami, Florida, USA. Association for Computational Linguistics.
- Lei Gao, Yulong Wang, Tongcun Liu, Jingyu Wang, Lei Zhang, and Jianxin Liao. 2021. Question-driven span labeling model for aspect–opinion pair extraction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 12875–12883.
- Maura R. Grossman, Gordon V. Cormack, and Adam Roegiest. 2016. [TREC 2016 total recall track overview](#). In *Proceedings of the 25th Text REtrieval Conference, TREC 2016, Gaithersburg, Maryland, USA*, volume 500-321. National Institute of Standards and Technology (NIST).
- Langchain. 2024. Faiss. <https://python.langchain.com/v0.2/docs/integrations/vectorstores/faiss/>. Accessed: 2024-04-20.
- Xin Li and Wai Lam. 2017. Deep multi-task learning for aspect term extraction with memory interaction. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2886–2892.
- Pengfei Liu, Shafiq Joty, and Helen Meng. 2015. Fine-grained opinion mining with recurrent neural networks and word embeddings. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1433–1443.
- Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2021. Sparse, dense, and attentional representations for text retrieval. *Transactions of the Association for Computational Linguistics*, 9:329–345.
- Ling Luo, Xiang Ao, Yan Song, Jinyao Li, Xiaopeng Yang, Qing He, and Dong Yu. 2019. Unsupervised neural aspect extraction with sememes. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 5123–5129.
- Dehong Ma, Sujian Li, Fangzhao Wu, Xing Xie, and Houfeng Wang. 2019. Exploring sequence-to-sequence learning in aspect term extraction. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 3538–3547.
- Kaixin Ma, Hao Cheng, Yu Zhang, Xiaodong Liu, Eric Nyberg, and Jianfeng Gao. 2023. [Chain-of-skills: A configurable model for open-domain question answering](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1599–1618, Toronto, Canada. Association for Computational Linguistics.
- Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, and Yinfei Yang. 2022. [Large dual encoders are generalizable retrievers](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9844–9855, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Haiyun Peng, Lu Xu, Lidong Bing, Fei Huang, Wei Lu, and Luo Si. 2020. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8600–8607.
- Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammed AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *ProWorkshop on Semantic Evaluation (SemEval-2016)*, pages 19–30. Association for Computational Linguistics.
- V. Priya and K. Umamaheswari. 2020. [Aspect-based summarisation using distributed clustering and single-objective optimisation](#). *Journal of Information Science*, 46(2):176–190.
- Paul Röttger, Bertie Vidgen, Dirk Hovy, and Janet B Pierrehumbert. 2021. Two contrasting data annotation paradigms for subjective nlp tasks. *arXiv preprint arXiv:2112.07475*.
- Sifatullah Siddiqi and Aditi Sharan. 2015. Keyword and keyphrase extraction techniques: a literature review. *International Journal of Computer Applications*, 109(2).
- An Tang, Xiuzhen Zhang, and Minh Dinh. 2024. [Aspect-based key point analysis for quantitative summarization of reviews](#). In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1419–1433, St. Julian’s, Malta. Association for Computational Linguistics.
- Sentence Transformers. 2024. Pretrained models. https://www.sbert.net/docs/sentence_transformer/pretrained_models.html. Accessed: 2024-04-20.
- Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. 2022. [GPL: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*,

pages 2345–2360, Seattle, United States. Association for Computational Linguistics.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-Hsuan Sung, et al. 2019. Multilingual universal sentence encoder for semantic retrieval. *arXiv preprint arXiv:1907.04307*.

Yelp. 2015. [Yelp open dataset](#). Dataset.

Liang-Chih Yu, Jin Wang, K. Robert Lai, and Xuejie Zhang. 2017. [Refining word embeddings for sentiment analysis](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 534–539, Copenhagen, Denmark. Association for Computational Linguistics.

Wenhao Yu, Hongming Zhang, Xiaoman Pan, Kaixin Ma, Hongwei Wang, and Dong Yu. 2023. Chain-of-note: Enhancing robustness in retrieval-augmented language models. *arXiv preprint arXiv:2311.09210*.

Wenxuan Zhang, Yang Deng, Xin Li, Yifei Yuan, Lidong Bing, and Wai Lam. 2021. [Aspect sentiment quad prediction as paraphrase generation](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9209–9219, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Jialin Pan, and Lidong Bing. 2023. [Sentiment analysis in the era of large language models: A reality check](#).

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2022. A survey on aspect-based sentiment analysis: Tasks, methods, and challenges. *IEEE Transactions on Knowledge and Data Engineering*.

Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2015. Representation learning for aspect category detection in online reviews. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Aligning Language Models for Icelandic Legal Text Summarization

Pórir Hrafn Harðarson

Department of
Computer Science
Reykjavik University
Iceland
thorirrh21@ru.is

Hrafn Loftsson

Department of
Computer Science
Reykjavik University
Iceland
hrafn@ru.is

Stefán Ólafsson

Department of
Computer Science
Reykjavik University
Iceland
stefanola@ru.is

Abstract

The integration of language models in the legal domain holds considerable promise for streamlining processes and improving efficiency in managing extensive workloads. However, the specialized terminology, nuanced language, and formal style of legal texts can present substantial challenges. This study examines whether preference-based training techniques, specifically Reinforcement Learning from Human Feedback and Direct Preference Optimization, can enhance models' performance in generating Icelandic legal summaries that align with domain-specific language standards and user preferences. We compare models fine-tuned with preference training to those using conventional supervised learning. Results indicate that preference training improves the legal accuracy of generated summaries over standard fine-tuning but does not significantly enhance the overall quality of Icelandic language usage. Discrepancies between automated metrics and human evaluations further underscore the importance of qualitative assessment in developing language models for the legal domain.

1 Introduction

The development of language models (LMs) for use in specialized, professional domains has the potential to create time-saving, value-adding processes. This may benefit various fields such as law, healthcare, and engineering, where much of the work involves analyzing and writing domain-specific texts and documents.

This is particularly relevant in the legal domain. An analysis of the legal systems in the USA and

Germany from 1998 to 2019 reported a monolithic growth in these systems (Coupette et al., 2021). Massive volumes of text data are a byproduct of most modern legal systems (Katz et al., 2020), leading to an environment with an ever-increasing amount of source material. Consequently, lawyers and attorneys must devote more time to analyzing and reviewing legal documents while preparing their casework, resulting in a growing workload in an already overburdened profession (Jónsdóttir, 2023; Nickum and Desrumaux, 2023).

A comprehensive awareness and understanding of relevant laws and precedents is paramount to success in legal arguments. Therefore, the ability to quickly summarize legal sources may significantly reduce the time spent reviewing pertinent material (Jain et al., 2021). Summaries can also serve as references for justifying claims and building cases. This is an area where generative LMs can be particularly useful, by processing and analyzing the bulk of the text needed.

In Iceland, there are substantial requirements within the legal domain that the quality of text meets the linguistic standards of the domain, both in terms of domain-specific terminology and general Icelandic language proficiency. Consequently, LMs must adhere to the professional standards of the domain in which they are applied. The legal domain is also characterized by a specialized vocabulary, particularly formal syntax, and semantics based on extensive domain-specific knowledge (Tiersma, 1999). This makes the task of aligning LMs to the specific language of the legal domain a non-trivial issue.

The most common method to enhance the capabilities of a pre-trained generative LM is instruction fine-tuning, where the model receives an instruction as input and the correct response as the target label (Radford and Narasimhan, 2018; Liu et al., 2019). Under this paradigm, the model is rewarded for correctly following the instruc-

tions; however, this does not necessarily entail that it captures the linguistic nuances within the target texts. *Reinforcement Learning from Human Feedback* (RLHF) is one such method that uses algorithms and reward-based methods from reinforcement learning (RL) to directly optimize a LM based on data collected from human feedback (Stiennon et al., 2020), aiming to help the model align better with both subjective and complex texts. Another more recent approach, based on the same principle, is *Direct Preference Optimization* (DPO) (Rafailov et al., 2024), which optimizes the model by transforming the RL reward maximization problem into a more simple classification problem. Though the complexity of the DPO method is less than that of RLHF, it is unclear which method is best suited to align LMs for summarizing Icelandic legal text.

This paper addresses the following research question:

RQ: Can preference training methods, such as DPO and RLHF, enhance the ability of LMs to generate domain-specific Icelandic texts that users prefer, compared to LMs fine-tuned solely with supervised learning?

We compared the quality of text summaries generated for the Icelandic legal domain by models fine-tuned with preference training to those fine-tuned solely through supervised learning. Our findings indicate that applying either RLHF or DPO on top of domain-specific pre-training and instruction fine-tuning can improve the legal accuracy of the generated summaries. However, no similar improvements were observed in the general quality of Icelandic language usage. Additionally, there were discrepancies between automated numerical evaluations and qualitative human assessments.

2 Background and Related Work

Transformer-based language models (LMs) have become central to text generation and NLP tasks, largely due to their adaptability when fine-tuned on specific tasks (Vaswani et al., 2017; Wolf et al., 2020). These models, typically containing billions of parameters (Touvron et al., 2023), excel at few-shot or zero-shot tasks that previously required supervised fine-tuning (Brown et al., 2020). However, languages with smaller speaker populations, such as Icelandic, face challenges due to

limited representation in training data. Efforts to address this include IceBERT, a masked LM for Icelandic (Snæbjarnarson et al., 2022), and GPT-SW3, a multilingual model covering most Nordic languages (Ekgren et al., 2024). These initiatives align with ongoing government initiatives in Iceland to preserve the Icelandic language amidst the rapid advancements in language technology (Nikulásdóttir et al., 2020) and with the government’s partnership with OpenAI.¹

Recent LM advancements emphasize RLHF to improve performance. Initial work by OpenAI explored human feedback to refine RL reward functions for complex tasks (Christiano et al., 2017). Stiennon et al. (2020) applied RLHF in NLP, training models for improved text summaries. Ouyang et al. (2022) extended this approach with InstructGPT, producing outputs that were preferred over those from larger models like GPT-3. RLHF-trained models have shown advantages in common sense reasoning and world knowledge (Glaese et al., 2022). A more streamlined approach, Direct Preference Optimization (DPO), optimizes the model directly via preference-based comparisons, showing similar performance to RLHF with faster results (Rafailov et al., 2024; Tunstall et al., 2024).

Given the powerful text processing capabilities of modern LMs, numerous studies have explored their applications in the legal domain, including judgment prediction (Trautmann et al., 2022), statutory interpretation (Blair-Stanek et al., 2023), legal reasoning (Yu et al., 2022), and using large models like ChatGPT as proxy legal advisors (Oltz, 2023). Research has also assessed performance on legal exams to gauge legal reasoning capabilities (Choi et al., 2022).

For domain-specific improvements, LEGAL-BERT (Chalkidis et al., 2020) demonstrates the advantages of pre-training a LM specifically for legal tasks, finding that additional domain-specific pre-training on legal corpora improved performance compared to using general-purpose BERT. Building on this work, Licari and Comandè (2024) developed Italian LEGAL-BERT, which they used in experiments for legal text summarization (Licari et al., 2023). Another Italian research, The PRODIGIT Project (Pisano et al., 2024), is a large-scale initiative aiming to support tax lawyers by

¹<https://openai.com/index/government-of-iceland/>

utilizing LMs for summarization. In a similar line of work, Schraagen et al. (2022) applied a BART-based LM for summarization of Dutch case verdicts. The LM-generated summaries were considered useful in human evaluations, although they still fall short of the quality of human-generated summaries.

3 Methods

We selected two open-source models for experimentation in generated Icelandic legal summaries. The first model, a 1.3B parameter version of GPT-SW3, has been pre-trained on Nordic languages using the Nordic Pile, a large corpus of approximately 1.2 TB, containing data in Swedish, English, Norwegian, Danish, and Icelandic (Ekgren et al., 2024; Öhman et al., 2023). The second model was a 7B parameter version of Llama2 (Touvron et al., 2023), mostly pre-trained on English texts.² With this setup, we compared the effectiveness of language-specific pre-training (GPT-SW3) to the general learning capacity of a larger model (Llama2).

To better understand the effect of pre-training on Icelandic texts, we created a sub-corpus of the Icelandic Gigaword Corpus (IGC) (Barkarson et al., 2022; Steingrímsson et al., 2018) that contained 10% of its data sampled at random. We then created a version of Llama2 (called Ice-Llama2) that was pre-trained on this sub-corpus.

All models were trained in three phases. In the first phase, the models were further pre-trained on domain-specific Icelandic legal text (see Section 3.1) and in the second phase, the models were fine-tuned to perform the supervised court case summarization task. After this training phase, the model able to produce the highest ROUGE score (Lin, 2004) – a commonly used metric for summarization tasks – was used to create summaries for a pairwise comparison dataset. Finally, in the third phase, this data was then used to perform preference training with DPO and RLHF.

3.1 Datasets

The datasets used for the training process are based on case rulings from the Icelandic supreme court, publicly available on the court’s website³. One row of data consists of a court ruling and a

summary made by a lawyer or attorney.

Many of the court rulings are too long to fit the context window of the chosen models. We therefore split the data into two parts: 1) long court rulings only (R); 2) court rulings that fit the window and their summaries (RS). The R dataset was used for the first phase of training, namely for further pre-training the models on domain-specific Icelandic legal text. The RS dataset was thus used for the second phase, fine-tuning the models to perform the summarization task. After splitting the data in this manner, the R dataset contained 5677 rows, split into 5077 rows (90%) of training data, 300 rows (5%) of test data and 300 rows (5%) of validation data. This left the RS dataset with 2,613 rows of data, further split into 2013 rows (78%) of training data, 300 rows (11%) of test data and 300 rows (11%) of validation data⁴.

3.2 Domain Specific Further Pre-Training

To investigate the importance of further pre-training on domain-specific text, the models were trained on the R dataset of court rulings only. As auto-regressive models, they were trained using self-supervised learning by shifting the input sequence forward by one token, creating target labels for predicting the next token in the sequence. The legal text in the dataset was processed by packing chunks of text together and dividing them into fixed-size blocks of 512 tokens. To measure the improvement in domain-specific text generation, the perplexity of both models on Icelandic legal text was estimated before and after fine-tuning, and the results were compared.

3.3 Instruction Fine-Tuning

Following the domain-specific training step, the models were fine-tuned using supervised learning to generate summaries. This was done using the RS dataset, where the models were given an input consisting of an instruction to create a summary, followed by the ruling text, and a token to mark the start of the summary. The corresponding label was the human-generated summary of the ruling from the court’s website. Due to the sequence length, the data was fed to the models in mini-batches of single rulings, but to ensure more stable training, the models processed eight sequences before calculating the gradient and updating the weights.

²Llama2 was the most powerful available open source models at the time of experimentation for this research.

³<https://www.haestirettur.is/domar/>

⁴https://huggingface.co/datasets/thorirhraf/domar_data

To evaluate the models’ ability to generate summaries and the impact of supervised fine-tuning, the ROUGE score was computed both before and after training.

3.4 Preference Training

The third phase of training was to apply preference training on top of the instruction fine-tuning to determine if it would improve performance. DPO requires a specialized dataset where the model is presented with two responses: one marked as preferred and the other as rejected. Then, it uses a loss function to compare these responses, directly penalizing the model for generating outputs that resemble the rejected data, increasing the likelihood of the model producing outputs that align with the preferred responses.

Implementing RLHF first involves training a reward model that serves as a reward function during training by classifying generated summaries and assigning scalar values based on its evaluation. This reward model was fine-tuned using a binary classification task on the dataset of preferred and rejected responses, which was also used for the DPO training. Training was carried out using Proximal Policy Optimization (PPO), a policy gradient algorithm that directly optimizes the policy guiding the model’s behavior. The goal is to maximize the probability of actions (i.e., generating summaries) that yield high rewards from the environment, given the current state. PPO limits the policy changes allowed at each training step, thereby ensuring greater stability and improving convergence to an optimal solution. Care must be taken that the values produced by the reward model need to be scaled appropriately. If the reward model’s interpretation of preferences is inconsistent or inaccurate, it can produce unstable reward signals, leading to conflicting feedback which can cause divergence during training (Stienon et al., 2020).

As before, performance was assessed by calculating the ROUGE score both before and after the RLHF training.

3.5 Human Evaluation

The final evaluation involved having legal experts rank the generated summaries. Using the test split of the RS dataset, the trained models were used to generate summaries which were then presented to human experts for ranking. One primary legal expert, an attorney with over five years experi-

ence and that has proceeded dozens of court cases, ranked summaries generated from 25 court rulings, selected to represent a wide variety of cases. To assess agreement, two additional legal experts ranked summaries for five of these cases. The primary expert also evaluated each generated summary by assigning two separate scores, each on a scale of 1 to 5: one score for the quality of the summary as a legal text, and another for the quality of the Icelandic used in the generated text. Here, the scores represent the quality expected within the legal domain, with a score of 5 meaning complete legal accuracy, and a near perfect use of the Icelandic language. A score of 1 would indicate a total misunderstanding of the legal argument and a totally unacceptable quality of Icelandic.

4 Results

To optimize training efficiency and make the best use of available resources, all models were trained using Low-Rank Adapters (LoRA) (Hu et al., 2022). The first parameter to be tuned and analyzed during pre-training was the adapter rank value. During the first phase of further pre-training on the R dataset (Icelandic court rulings only), increasing the rank consistently led to a lower loss. This suggests that increasing the number of trainable parameters helps the model to learn better from the training data.

Based on these findings, a relatively large adapter with a rank of 1024 was used for training Llama2-7B on the IGC sub-corpus data and a rank of 256 for phase one (further pre-training on the R data, court rulings only), and a rank of 128 for phase two (training for the summarization task on the RS data, court rulings and summaries).

To assess the impact of phase one, all models were evaluated by calculating their perplexity scores on the test split of the dataset.

Model	Perplexity
GPT-SW3-1.3B	5.281
Llama2-7B	9.283
Ice-Llama2-7B	5.048

Table 1: Perplexity evaluation on legal text in Icelandic before using further pre-training on legal data.

As shown in Table 1, the base Llama2 model initially scored significantly higher in perplexity compared to both GPT-SW3 and Ice-Llama2,

which had been pre-trained on the ICG sub-corpus data. After the phase one training process, however, both Llama2 variants achieved lower perplexity scores than GPT-SW3.

Model	Perplexity
GPT-SW3-1.3B	4.844
Llama2-7B	2.981
Ice-Llama2-7B	2.900

Table 2: Perplexity evaluation on legal text in Icelandic after further pre-training on Icelandic court rulings data.

4.1 Instruction Fine-Tuning

The second phase of training was supervised instruction fine-tuning using the RS dataset (see Section 3.1), along with the corresponding instruction text and summaries. To determine the optimal number of training epochs, the GPT-SW3 1.3B model was trained for 1, 3, and 5 epochs, with performance evaluated using 10 summaries from the validation set. The model trained for 5 epochs achieved the highest ROUGE scores, so all models were subsequently trained for 5 epochs on the training split of the dataset.

After fine-tuning, the models were evaluated by generating summaries for all 300 entries in the test set. The generated summaries were compared to human-generated baselines using ROUGE scores. As shown in Table 3, both Llama2-7B variants achieved higher scores than GPT-SW3-1.3B:

Model	Rouge1	Rouge2	RougeL
GPT-SW3-1.3B	0.2829	0.1136	0.1796
Llama2-7B	0.3055	0.1112	0.1872
Ice-Llama2-7B	0.3005	0.1121	0.1861

Table 3: ROUGE-score evaluation using all 300 summaries in the test dataset after further pre-training on legal data and instruction fine-tuning.

4.2 Preference Training

4.2.1 Direct Preference Optimization

In the third phase of training, investigating the impact of additional preference training, we first looked at using the DPO method. Since both Llama2-7B variants achieved nearly identical ROUGE scores after instruction fine-tuning, further training was only applied to the base Llama2-7B model and GPT-SW3-1.3B to highlight the dif-

ferences between larger models and those with extensive language-specific pre-training.

Following the method used by Tunstall et al. (2024) for training the Zephyr 7B model with DPO, all models were initially fine-tuned for 1, 3, and 5 epochs and then further trained with DPO for an additional 1, 2, and 3 epochs. The resulting ROUGE scores were evaluated to assess the effect of preference training on top of varying levels of supervised fine-tuning. GPT-SW3-1.3B achieved its best improvements after 2 epochs of DPO training, following 5 epochs of fine-tuning. However, performance plateaued after 3 epochs, and in some cases began to degrade, likely due to overfitting.

For the Llama2-7B model, the best improvements were observed after 2 epochs of DPO training but with only 1 epoch of prior fine-tuning. Given the larger parameter count and higher learning capacity of Llama2-7B, the risk of overfitting was more pronounced. To mitigate this, training was conducted with a low starting learning rate of $7e-07$, as even slight increases led to overfitting.

After completing this training process, the best-performing versions of GPT-SW3-1.3B and Llama2-7B were evaluated on the entire test set of summaries. This resulted in significant improvements for GPT-SW3-1.3B and modest gains for Llama2-7B, as shown in Table 4:

Model	Rouge1	Rouge2	RougeL
GPT-SW3-1.3B	0.3381	0.1637	0.2263
Llama2-7B	0.3143	0.1226	0.1963

Table 4: ROUGE-score evaluation on generating 300 summaries for court rulings in Icelandic after further pre-training on legal data, supervised instruction fine-tuning, and DPO.

4.2.2 Reinforcement Learning from Human Feedback

The second preference training method evaluated was RLHF. As outlined in Section 3, RLHF involves first training a reward model to classify the output of the policy model and return a scalar reward based on the likelihood that the generated output aligns with human preferences. The same pairwise dataset used during the DPO training phase was utilized to train this reward model. Initial attempts revealed a high susceptibility to overfitting, necessitating training for just a single epoch with a relatively low learning rate of $2e^{-06}$.

Early efforts to use the reward model to train a policy using PPO resulted in highly unstable training. The policy quickly learned to exploit the reward model by generating sequences of empty lines, random characters, or incomplete word endings, leading to a spike in KL divergence and inflated rewards.

To stabilize the training, the output of the reward model was normalized such that the rewards had a mean of $\mu = 0$ and a standard deviation of $\sigma = 1$ at the start of training. This normalization led to a much more stable training process. However, the PPO algorithm’s conservative policy updates resulted in slow learning progression, as shown in Figure 1:

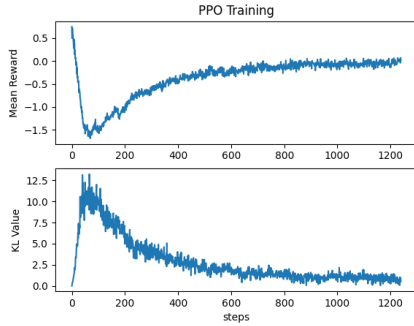


Figure 1: Mean reward and KL-divergence for GPT-SW3 1.3B after 20 epochs of training using the PPO reinforcement learning algorithm.

Due to the substantial GPU resources required for RLHF, this training method was only applied to the smaller GPT-SW3 model. As with previous evaluations, the model’s performance was assessed by calculating the ROUGE score. However, in contrast to its DPO-trained counterpart, the RLHF model did not exhibit performance improvements, as can be seen in Table 5:

Model	Rouge1	Rouge2	RougeL
GPT-SW3-1.3B	0.2690	0.1058	0.1769

Table 5: ROUGE-score evaluation on generating 300 summaries for court rulings in Icelandic after further pre-training on legal data, instruction fine-tuning, and reinforcement learning.

4.3 Human Evaluation

As a final evaluation step, the results generated from 25 court rulings by five model variations were ranked by a human expert in the legal do-

main, tasked with ranking the summaries from 1st place to 5th:

Model	Order	Average
GPT-SW3-RLHF	1	2.20
GPT-SW3-SFT	2	2.36
Llama2-DPO	3	3.24
Llama2-SFT	4	3.52
GPT-SW3-DPO	5	3.68

Table 6: Average rank for five model variations after being ranked on summary generation for 25 court rulings by a legal expert. The model names have an ending that marks if they were additionally fine-tuned using either DPO or RLHF, or if they were only instruction fine-tuned using supervised learning (SFT).

As can be seen in Table 6, the version of GPT-SW3-1.3B that had only been instruction fine-tuned and the version that had also been additionally trained with RLHF were most often chosen as the preferred models, despite having achieved the lowest scores during evaluation. Two other experts also ranked the first five of the 25 chosen rulings to get an assessment on the agreement between human legal experts, the results of which can be seen in Table 7:

Model	Primary	Comparison
GPT-SW3-SFT	1.6	2.3
GPT-SW3-RLHF	2.2	2.6
Llama2-DPO	3.0	3.0
Llama2-SFT	4.0	4.0
GPT-SW3-DPO	4.2	3.2

Table 7: Average rank for five model variations after being ranked on summary generation for the first 5 court rulings from the list of 25. Rank scores from the primary expert compared to the average from two other legal experts to assess agreement.

To further assess the models’ capabilities, the primary evaluator assigned each model a score on a scale of 1 to 5, where 1 represented the lowest performance and 5 the highest. The models were evaluated based on two criteria: the quality of the Icelandic language used in the generated text and the legal accuracy in relation to the court ruling being summarized.

Looking at the results in Table 8, both variations of GPT-SW3-1.3B that were ranked in the top two positions also achieved the highest scores for Ice-

Model	Icelandic	Legal Accuracy
Baseline	4.96	4.8
GPT-SW3-SFT	4.04	2.68
GPT-SW3-RLHF	3.96	2.56
Llama2-DPO	2.88	2.52
Llama2-SFT	2.92	2.04
GPT-SW3-DPO	3.24	1.96

Table 8: Average scores for five model variations on the quality of the Icelandic used and the legal accuracy after being assessed by a legal expert on summaries generated for 25 court rulings.

landic language quality and legal accuracy. The two Llama2-7B variations exhibited similar scores for language quality, but the DPO version scored higher in legal accuracy. In contrast, the GPT-SW3-1.3B DPO variant received notably lower scores for Icelandic language quality compared to the other GPT-SW3 versions and had the lowest score for legal accuracy, despite achieving the highest ROUGE score overall. When compared to human-generated summaries, all models scored significantly lower, particularly in terms of legal accuracy.

5 Discussion

5.1 Language Specific Pre-training

After the self-supervised training on the legal text in the R dataset, the Ice-Llama2 model, which had also previously been trained on Icelandic texts from the IGC, was expected to achieve the best scores. However, the results (see Table 3) indicate otherwise, showing only a marginal difference between the two Llama2-7B models. This suggests that when fine-tuning a model intended for further domain-specific training, it might be more beneficial to utilize more curated high-quality domain-specific datasets, even if this means training on less data. Such an approach allows the model to more effectively capture the relevant words and phrases it will encounter while performing downstream tasks within the specific domain, increasing the likelihood of accurately predicting the necessary tokens. Further evidence can be observed in the ROUGE score results after the summary generation training using instruction fine-tuning, where no significant difference was found between the two Llama2-7B models. The ROUGE scores for the model-generated summaries are generally modest. However, caution is needed when inter-

preting these results, as summaries of court rulings are often concise descriptions of the outcomes, which may not include much of the ruling’s text and can be phrased differently. Consequently, assessing the quality of the generated text, based solely on N-gram overlap, can sometimes be challenging.

5.2 Model Evaluation

While numerical evaluations are valuable for assessing the training process, they may overlook important nuances and subjective qualities in language. This discrepancy is evident in Table 9, which compares the models’ standings based on perplexity scores with the subjective assessments of domain experts regarding the quality of the Icelandic text generated by the models.

Rank	Perplexity Score	Qualitative Analysis
1	Llama2-DPO	GPT-SW3-SFT
2	Llama2-SFT	GPT-SW3-RLHF
3	GPT-SW3-SFT	GPT-SW3-DPO
4	GPT-SW3-RLHF	Llama2-SFT
5	GPT-SW3-DPO	Llama2-DPO

Table 9: Ranking of evaluated models comparing perplexity scores with results from qualitative analysis on the use of Icelandic by a domain expert.

The same limitations can also be observed in in Table 10, comparing the rankings of these models based on their ROUGE scores against the subjective analysis of domain experts on the legal accuracy of generated summaries:

Rank	ROUGE Score	Qualitative Analysis
1	GPT-SW3-DPO	GPT-SW3-SFT
2	Llama2-DPO	GPT-SW3-RLHF
3	Llama2-SFT	Llama2-DPO
4	GPT-SW3-SFT	Llama2-SFT
5	GPT-SW3-RLHF	GPT-SW3-DPO

Table 10: Ranking of evaluated models comparing ROUGE-scores with results from qualitative analysis on the legal accuracy in generated summaries by a domain expert.

The ROUGE scores and analyses by domain experts for both variations of the Llama2-7B model suggest that improvements can be achieved through preference training, such as DPO. In contrast, the results for the GPT-SW3-1.3B-DPO

model present a different narrative. While this model demonstrates a significant improvement in ROUGE scores compared to other GPT-SW3-1.3B variants, it is frequently rated as the least preferred option by domain experts.

A detailed analysis of the legal accuracy scores reveals that the GPT-SW3-1.3B-DPO model is the only one of the model variations evaluated to receive full marks for legal accuracy in its summaries. However, it also frequently garnered low scores of 1 or 2. These contradictory results suggest that the model might have over-fitted, enabling it to sometimes produce relatively high-quality summaries while most often failing to generalize effectively.

Despite the shortcomings of DPO regarding over-fitting, its user-friendliness compared to RLHF makes it a preferable starting point for exploring whether preference training can enhance performance, as achieving stable RLHF training without divergence can present a significant challenge. Additionally, RLHF demands more computational resources than DPO. However, it cannot be overlooked, as evidenced by the results showing that the RLHF model outperformed both DPO variations in evaluations by human experts. Furthermore, RLHF offers greater flexibility in developing the reward model, as it is not limited to pairwise comparisons, which could be advantageous for specific applications.

Overall, none of the models matched the capabilities of human experts in the evaluation, especially with regard to legal accuracy. Furthermore, the discrepancy in the quality of Icelandic text between Llama2-7B and GPT-SW3-1.3B highlights the importance of language-specific pre-training, as GPT-SW3-1.3B consistently produced higher-quality text in Icelandic.

5.3 Qualitative Analysis

The main domain expert reviewed the output of the models and found that they performed reasonably well overall in generating sentences that reflect the expected language and phrasing found in court rulings and summaries. However, the contextual flow between individual sentences was inconsistent, with some examples displaying a lack of cohesion and contradictory statements within the same summary, such as “the Supreme Court dismissed the case” and then “the Supreme Court denied the request for dismissal of the case”. The

models also struggled to adapt their summaries to the predetermined text length; with some being noticeably incomplete, while others including unnecessary sentences added to an otherwise complete summary.

While the models successfully identified essential components, such as the case subject and the court’s decision, the expert found that they frequently overlooked key arguments and relevant statutes that influenced the outcome. The factual accuracy was mediocre, with several instances of contradictory statements, e.g., one correctly stating the outcome while another contradicting it. Additionally, the model occasionally confused the roles of the parties involved in a case, leading to inaccuracies about which party appealed the case or made specific claims or arguments. This sometimes carried over in the use of pronouns, creating circular sentences, such as ‘the claimant requested that his [own] claim be dismissed’.

This review highlights that preference training can produce legal summaries in Icelandic that are useful to some extent, but more work needs to be done before such software can be used in practice.

6 Conclusions

We evaluated the effect of language-specific and preference training to enhance the ability of LMs in generating Icelandic legal text summaries, compared to LMs fine-tuned solely with supervised learning. An analysis of the evaluation results reveals that models further trained using either DPO or RLHF can exhibit improved performance in domain-specific language generation compared to those solely fine-tuned through supervised instruction; however, not consistently. Notably, this additional preference training did not lead to a general improvement in the quality of Icelandic used in the generated text. This underscores the critical role of language-specific pre-training in establishing a robust foundation for language generation.

A notable finding was the gap between ROUGE scores and expert preferences, suggesting earlier integration of human feedback could be beneficial. The dataset for pairwise comparison was based on responses with top ROUGE scores post fine-tuning. A more effective approach might involve gathering human feedback at this stage to identify which model is best suited for generating data for further training. However, this approach is constrained by the high costs associated with obtain-

ing feedback from professional experts. While the expert feedback gathered provides valuable insight into their preferences, achieving significant improvements driven by human feedback will likely require additional resources and investment.

Future work should emphasize language-specific pre-training on Icelandic legal texts, focusing on laws, bills, and resolutions. This could enhance Icelandic quality while expanding legal knowledge. Leveraging newer models with extended context windows, such as the now available Llama3 family, could enable better utilization of training data by processing longer rulings. This capability would allow the inclusion of more training samples, potentially leading to improvements in performance. In addition to this, Retrieval-Augmented Generation (RAG) (Lewis et al., 2020) could be used to give models access to external knowledge when generating summaries, helping to increase factual accuracy in the responses. Moreover, a greater variety of LMs should be evaluated, as well as a larger cohort of legal experts.

7 Limitations

A limitation of this research was the dataset size, capped at 2,600 rows, while comparable studies used about 120,000 rows (Stiennon et al., 2020). Expanding with public court rulings and lower court summaries could improve outcomes, as a larger dataset of quality data is crucial for successfully training viable models. Additionally, the models selected were only a subset of the models available, and we had a limited number of legal experts participating in our experiments. These limitations may affect the generalization of our findings to other domains and languages.

References

- Starkaður Barkarson, Steinþór Steingrímsson, and Hildur Hafsteinsdóttir. 2022. Evolving Large Text Corpora: Four Versions of the Icelandic Gigaword Corpus. In *Proceedings of the Language Resources and Evaluation Conference*, page 11, Marseille, France.
- Andrew Blair-Stanek, Nils Holzenberger, and Benjamin Van Durme. 2023. Can GPT-3 Perform Statutory Reasoning? In *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*, ICAIL '23, page 22–31. Association for Computing Machinery.
- 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. Curran Associates, Inc.
- Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The Muppets straight out of Law School. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904. Association for Computational Linguistics.
- Jonathan H. Choi, Kristin E. Hickman, Amy Monahan, and Daniel Schwarcz. 2022. ChatGPT Goes to Law School. *Journal of Legal Education*, 71:387.
- Paul F Christiano, Jan Leike, Tom Brown, Miljan Maric, Shane Legg, and Dario Amodei. 2017. Deep Reinforcement Learning from Human Preferences. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.
- Corinna Coupette, Janis Beckedorf, Dirk Hartung, Michael James Bommarito, and Daniel Martin Katz. 2021. Measuring Law Over Time: A Network Analytical Framework with an Application to Statutes and Regulations in the United States and Germany. *Frontiers in Physics* 9.
- Ariel Ekgren, Amaru Cuba Gyllensten, Felix Stollenwerk, Joey Öhman, Tim Isbister, Evangelia Gogoulou, Fredrik Carlsson, Judit Casademont, and Magnus Sahlgren. 2024. GPT-SW3: An Autoregressive Language Model for the Scandinavian Languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7886–7900, Torino, Italia. ELRA and ICCL.
- Amelia Glaese, Nat McAleese, Maja Trębacz, John Aslanides, Vlad Firoiu, Timo Ewalds, Maribeth Rauh, Laura Weidinger, Martin Chadwick, Phoebe Thacker, Lucy Campbell-Gillingham, Jonathan Uesato, Po-Sen Huang, Ramona Comanescu, Fan Yang, Abigail See, Sumanth Dathathri, Rory Greig, Charlie Chen, Doug Fritz, Jaume Sanchez Elias, Richard Green, Soňa Mokrá, Nicholas Fernando, Boxi Wu, Rachel Foley, Susannah Young, Iason Gabriel, William Isaac, John Mellor, Demis Hassabis, Koray Kavukcuoglu, Lisa Anne Hendricks, and Geoffrey Irving. 2022. Improving alignment of dialogue agents via targeted human judgements. *ArXiv*.

- 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*.
- Deepali Jain, Malaya Dutta Borah, and Anupam Biswas. 2021. Summarization of legal documents: Where are we now and the way forward. *Computer Science Review*, 40:100388.
- Eva Hrönn Jónsdóttir. 2023. Helmingur hefur velt fyrir sér að skipta um starfsvettvang. *Lögmannaþaðið*, 03/23:16–18.
- Daniel Martin Katz, Corinna Coupette, Janis Beckedorf, and Dirk Hartung. 2020. Complex societies and the growth of the law. *Scientific Reports*, 10(1).
- 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 *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.
- Daniele Licari, Praveen Bushipaka, Gabriele Marino, Giovanni Comandè, and Tommaso Cucinotta. 2023. Legal holding extraction from italian case documents using italian-legal-bert text summarization. In *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law, ICAIL '23*, page 148–156, New York, NY, USA. Association for Computing Machinery.
- Daniele Licari and Giovanni Comandè. 2024. ITALIAN-LEGAL-BERT models for improving natural language processing tasks in the Italian legal domain. *Computer Law & Security Review*, 52:105908.
- 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*.
- Marion Nickum and Pascale Desrumaux. 2023. Burnout among lawyers: effects of workload, latitude and mediation via engagement and over-engagement. *Psychiatry, Psychology and Law*, 30(3):349–361.
- Anna Nikulásdóttir, Jón Guðnason, Anton Karl Ingason, Hrafn Loftsson, Eiríkur Rögnvaldsson, Einar Freyr Sigurðsson, and Steinþór Steingrímsson. 2020. Language Technology Programme for Icelandic 2019–2023. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3414–3422, Marseille, France. European Language Resources Association.
- Tammy Pettinato Oltz. 2023. Chatgpt, professor of law. *University of Illinois Journal of Law, Technology & Policy*, page 207.
- 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. In *Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22*.
- Giuseppe Pisano, Alessia Fidelangeli, Federico Galli, Andrea Loreggia, Riccardo Rovatti, Piera Santin, and Giovanni Sartor. 2024. Summarization of tax rulings in the PRODIGIT projec. *i-lex*, 17(1):1–26.
- Alec Radford and Karthik Narasimhan. 2018. Improving Language Understanding by Generative Pre-Training.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2024. Direct preference optimization: your language model is secretly a reward model. In *Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23*, Red Hook, NY, USA. Curran Associates Inc.
- Marijn Schraagen, Floris Bex, Nick Van De Luitgaarden, and Daniël Prijs. 2022. Abstractive Summarization of Dutch Court Verdicts Using Sequence-to-sequence Models. In *Proceedings of the Natural Legal Language Processing Workshop 2022*, pages 76–87, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Vésteinn Snæbjarnarson, Haukur Barri Símonarson, Pétur Orri Ragnarsson, Svanhvít Lilja Ingólfssdóttir, Haukur Jónsson, Vilhjálmur Thorsteinsson, and Hafsteinn Einarsson. 2022. A Warm Start and a Clean Crawled Corpus - A Recipe for Good Language Models. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4356–4366, Marseille, France. European Language Resources Association.
- Steinþór Steingrímsson, Sigrún Helgadóttir, Eiríkur Rögnvaldsson, Starkaður Barkarson, and Jón Guðnason. 2018. Risamálheild: A Very Large Icelandic Text Corpus. In *Proceedings of LREC 2018*, Myazaki, Japan.
- Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. 2020. Learning to summarize from human feedback. In *Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20*, Red Hook, NY, USA. Curran Associates Inc.
- Peter M. Tiersma. 1999. *Legal Language*, 1 edition. University of Chicago Press.

- Hugo Touvron, Louis Martin, Kevin R. Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Daniel M. Bikel, Lukas Blecher, Cristian Cantón Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony S. Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel M. Kloumann, A. V. 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, R. Subramanian, Xia Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zhengxu 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*, abs/2307.09288.
- Dietrich Trautmann, Alina Petrova, and Frank Schilder. 2022. Legal Prompt Engineering for Multilingual Legal Judgement Prediction. *ArXiv*.
- Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Cl  mentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. 2024. Zephyr: Direct Distillation of LM Alignment. In *Conference on Language Modeling*.
- 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*, pages 5998–6008.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pi  ric 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.
- Fangyi Yu, Lee Quartey, and Frank Schilder. 2022. Legal Prompting: Teaching a Language Model to Think Like a Lawyer. *ArXiv*.
- Joey   hman, Severine Verlinden, Ariel Ekgren, Amaru Cuba Gyllensten, Tim Isbister, Evangelia Gogoulou, Fredrik Carlsson, and Magnus Sahlgren. 2023. The Nordic Pile: A 1.2TB Nordic Dataset for Language Modeling. *ArXiv*.

Question-parsing with Abstract Meaning Representation enhanced by adding small datasets

Johannes Heinecke¹, Maria Boritchev², Frédéric Herledan¹

¹Orange Innovation, 2 avenue Pierre Marzin, 22300 Lannion, France

²Paris Télécom, 19 place Marguerite Perey, 91120 Palaiseau, France

{johannes.heinecke, frederic.herledan}@orange.com
maria.boritchev@telecom-paris.fr

Abstract

Abstract Meaning Representation (AMR) is a graph-based formalism for representing meaning in sentences. As the annotation is quite complex, few annotated corpora exist. The most well-known and widely-used corpora are LDC’s AMR 3.0 and the datasets available on the new AMR website. Models trained on the LDC corpora work fine on texts with similar genre and style: sentences extracted from news articles, Wikipedia articles. However, other types of texts, in particular questions, are less well processed by models trained on this data. We analyse how adding few sentence-type specific annotations can steer the model to improve parsing in the case of questions in English.

1 Introduction

Abstract Meaning Representation (Banarescu et al., 2013) provides a framework to model the meaning of a sentence, notably actions, events or states and their participants. AMR relies heavily on (verbal) concepts defined in PropBank (Kingsbury and Palmer, 2002; Palmer et al., 2005), e.g. *bear-02* in figure 1, PropBank’s sense -02 for the verb “to bear”. Instances are indicated by a following “/”, e.g., *p* being an instance of the concept *person*. The names of the variables do not have any other semantics than being distinct. Relations are indicated by an initial colon (e.g. *:ARG1*, *:time*). Literals (strings and numbers) lack a preceding instance and “/” (c.f. “*Elizabeth*” and *1926* in the example in figure 1). This serialised format, shown in figure 1 left, is called PENMAN (Kasper, 1989).

The largest available corpus used to train models capable of parsing sentences from natural languages into AMR graphs, called AMR 3.0,

```
(b / bear-02
 :ARG1 (p / person
        :name (n / name
              :op1 "Queen"
              :op2 "Elizabeth"))
 :time (d / date-entity
       :year 1926))
```

Figure 1: AMR graph for “Queen Elizabeth was born in 1926” in PENMAN format.

LDC2020T02¹, is provided by the Linguistic Data Consortium (LDC). This corpus is composed of nearly 59 000 sentences and corresponding AMR graphs. The data contains discussions from forums (partly technical), news reels, translations to English of Chinese news broadcasts, along with a part originating from English Wikipedia pages and Aesop’s fables (see LDC2020T02 documentation).

The problem we address in this article is the following: the gold data currently available for AMR parsing is very homogeneous in form as it is composed of declarative, informative sentences. Training models on such data yields lower-than-expected results for parsing of questions in AMR. We add a small dataset of questions to the training data to bypass this problem. Even if we were intuitively expecting this kind of result, we were able to confirm it and measure improvement.

2 Related Work

Domain type adaptation research for AMR has been attempted in several contexts and perspectives, one of the most well-known leading to the development of Bio-AMR². Bio-AMR includes texts from the biomedical domain, extracted from PubMed³. Vu et al. (2022) conducted a research

¹Knight et al. (2020), <https://catalog.ldc.upenn.edu/LDC2020T02>

²Available on the new AMR webpage: <https://github.com/flipz357/AMR-World>

³<https://pubmed.ncbi.nlm.nih.gov/>

on AMR of data outside news article excerpts, focusing on the legal documents domain, using a gold dataset. The parsing results were not very conclusive, and the authors provide a detailed discussion of this result. Among the explanations for the models not-so-good performances, two stand out: first, legal documents contain mostly sentences longer than the ones from LDC datasets; then, the models faced out-of-vocabulary (OOV) issues, as some concepts, specific to the legal domain, were not defined in PropBank. This latter issue comes from the semantic difference between the news and the legal documents domains.

Lee et al. (2022) experimented on sentence-type adaptation through both algorithmic and data-based research. They created and released the QALD-9-AMR corpus, built on top of QALD-9 data (Usbeck et al., 2018). It contains AMR annotations for natural language questions in English, originally provided for executable semantic parsing. Lee et al. (2022) further mention one unavoidable difficulty for domain adaptation which is out-of-vocabulary named entities and their types, that cannot be solved without using domain-specific corpora. The authors compare the usage of silver data with that of human annotation for QALD-9.

3 AMR Parsing of Short Questions

In this section, we present our data and our parsing methods, followed by first observations and hypotheses.

Corpora The AMR 3.0 corpus mainly contains sentences from newspapers and a small part of Wikipedia. There is almost no real question in this corpus (apart from a few rhetorical ones). Our hypothesis is that a model trained on this data will not perform well on question parsing. Thus, our research question is to see whether it is possible to improve the model’s performance on questions by adding a small corpus of short questions to its training data (i.e. AMR 3.0 train).

The data we used in this article is the following:

- a. AMR 3.0, about 55 000 sentences for training, 1 722 sentences for validation and 1 898 sentences for test.
- b. QALD-9 Lee et al. (2022)⁴, contains 400 (train) and 150 (test) questions taken from the QALD-9 project and annotated using AMR.

⁴<https://github.com/IBM/AMR-annotations>

The test set of QALD-9 contains 13 sentences which are also in the train corpus and in the QALD-7 and QALD-8 data which served as input for QUERO. We deleted them from the QALD-9 test set, and use only the 137 remaining sentences. (c.f. fig. 2).

- c. QUERO: a corpus we created, which contains 406 (training) short, quiz-like questions of the same type as the ones in QALD-9, coming amongst other sources from QALD-7, QALD-8. The 406 sentences are equally divided between questions and the corresponding answers.

About 25% of the questions and all answers in QUERO were formulated prior to the AMR annotation by human annotators from our team⁵ (cf. fig. 3 and 4). Table 1 details the size of the corpora. An answer can often be formulated in various ways: “Edinburgh is the capital of Scotland” and “the Scottish capital is Edinburgh”, yielding very similar AMR graphs.

QUERO was created by two annotators by correcting AMRlib’s output annotations and checking PropBank concepts and associated arguments. The computation of pairwise Smatch scores shows a relatively good quality of annotation with an inter-annotator agreement of 87.37%. In case of disagreement, the best annotation was chosen manually by a third annotator.

corpus	training	dev.	test
AMR 3.0	55 635	1 722	1 898
QALD-9 AMR	357	51	137
QUERO	358	48	0

Table 1: Number of sentences in the used corpora. QALD-9 only comes with a train and a test set. We split 51 sentences from the training corpus in order to have a development set as well.

Parser We use a slightly modified version of the AMRlib⁶ parser, which can use as an underlying language model models other than T5. In our case, we adapted AMRlib to use the multilingual version MT5 and FLAN-T5. The base data is the

⁵The annotators used the official AMR annotation guidelines available at <https://github.com/kevincrawfordknight/amr-guidelines/blob/master/amr.md>. The AMR annotation was undertaken using metAMoRphosED (<https://github.com/Orange-OpenSource/metamorphosed/>) (Heinecke, 2023)

⁶<https://github.com/bjacob/amrlib>

```
(e / erupt-01
  :ARG1 (v / volcano
    :mod (a / amr-unknown)
    :location (c / country
      :name (n / name
        :op1 "Japan"))
    :time (s / since
      :op1 (d / date-entity
        :year 2000)))
```

“Which volcanos in Japan erupted since 2000?”

Figure 2: Example question from QALD-9 test corpus

```
(g / game
  :name (n / name
    :op1 "Winter"
    :op2 "Olympic"
    :op3 "Games")
  :time (d / date-entity
    :year 2010)
  :location (c / city
    :mod (a / amr-unknown)))
```

“In which city did the 2010 Winter Olympic Games take place?”

Figure 3: Example question from our corpus

AMR 3.0 corpus, which we augment with datasets containing short questions. We trained the models using either T5 (Raffel et al., 2020), FLAN-T5 (Chung et al., 2022) or MT5 (Xue et al., 2021) as underlying language model (base size in all four cases). For evaluation, we use the Smatch package (Cai and Knight, 2013)⁷.

Observations We have noticed that models relying on the AMR 3.0 corpus perform less well in terms of Smatch F1 when it comes to questions and answer sentences, both in QALD-9 and our own data. Questions in QALD-9 are mostly short sentences, so generally a better performance would be expected. Table 2 shows these initial results on models trained by fine-tuning different language models.

Even though the results for QALD-9 are better than the ones for AMR3.0, we were expecting a larger difference in figures. The sentences in QALD-9 are much shorter compared to the ones from AMR3.0: 43.6 characters/sentence for QALD-9, 112.0 characters/sentence for AMR3.0.

Hypotheses This under-performance could be due to two factors: the slightly different syntax of questions with respect to declarative sentences (e.g. “to do” periphrasis in English or the “est-

⁷<https://github.com/snowblink14/smatch>

```
(g / game
  :name (n / name
    :op1 "Olympic"
    :op2 "Winter"
    :op3 "Games")
  :time (d / date-entity
    :year 2010)
  :location (c / city
    :name (n2 / name
      :op1 "Vancouver")))
```

“The 2010 Olympic Winter Games took place in Vancouver.”

Figure 4: Example answer from our corpus

LM	AMR 3.0	QALD-9
T5	81.8	87.2
FLAN T5	82.2	86.4
MT5 (en-fr)	81.6	85.7

Table 2: Results on the AMR3.0 test corpus and the QALD-9 test corpus. All models were trained on AMR3.0 train corpus only.

ce que” construction in French), or the missing coverage of vocabulary used in QALD-9 and our questions compared to the AMR 3.0 training corpus (for instance the concepts abbreviate-01, skateboard-01 or novelist). Therefore, if the parser encounters an unknown concept, a solution is to use a fake concept appending “-01” to the concept’s name. However, we do not encounter this particular problem in our setting yet.

In the remainder of this paper we describe our AMR parsing based on our version of AMRlib, the additional data and the obtained results.

4 Effect of Adding Questions to the Training Data

After our first observations, we trained models using different combinations of augmented data.

Experimental setup In a first step we trained three models using the AMR 3.0 training corpus. This gives us our baseline results (table 2), for the AMR 3.0 test corpus and the QALD-9 test corpus.

We then extended the training data with the QALD-9 training data, with our data, and finally with both. The QALD-9 AMR comes in two files, a training and a test corpus. We took 51 sentences from the training corpus to have a development corpus (see table 1). We used QALD-9 AMR’s test corpus to test our model for the sake of reproducibility of our research results.

LM	lg.	train data	test data	
			AMR 3.0	QALD-9
T5	English	baseline	81.8	87.2
		+ QUEREO	82.0 (+0.2)	86.8 (-0.4)
		+ QALD-9	81.9 (+0.1)	90.0 (+2.8)
		+ QR. + Q9	82.0 (+0.2)	89.5 (+2.3)
FLAN-T5	English	baseline	82.2	86.4
		+ QUEREO	82.4 (+0.2)	86.8 (+0.4)
		+ QALD-9	82.1 (-0.1)	89.7 (+3.3)
		+ QR. + Q9	82.1 (-0.1)	89.6 (+3.2)
MT5	En + Fr	baseline	81.6	85.7
		+ QUEREO	81.4 (-0.2)	86.6 (+0.9)
		+ QALD-9	81.8 (+0.2)	89.8 (+4.1)
		+ QR. + Q9	81.8 (+0.2)	89.6 (+3.9)

Table 3: Test results: Best figures for a test corpus with the same language model (T5, FLAN-T5, MT5) in italics, best overall score in bold. QR stands short for QUEREO, Q9 stands for QALD-9. The baseline is a corpus trained only on the AMR 3.0 training data, the difference with respect to the baseline is shown in small digits. The baseline is taken from table 2.

Results The results are shown in table 3. The baseline (already shown in table 2) is given by the models trained only on AMR 3.0 training data provided by LDC. Adding a little additional data to the AMR 3.0 training corpus we were able to improve significantly the parsing results, even for the AMR 3.0 test data. This is independent of the underlying language model.

5 Discussion

In this paper we showed that even minor additions to the standard AMR 3.0 training corpus can have big impacts on the performance of an AMR parser for a new sentence type, syntactic in the case of questions. Next, we plan on taking our studies further by annotating a domain specific corpus in the domain of artificial intelligence) or noisy data.

We are aware of the problems of Smatch-based evaluation, and we follow the other algorithms for AMR comparison that have been proposed, in particular semantic Smatch such as S2match (Opitz et al., 2020). In future work, we would like to broaden our exploration of the benefits of adding a small corpus of specialised examples through different dimension of AMR, using different types of evaluation metrics.

The exploration conducted in this paper has fo-

cused on one method of parsing, the one provided by our extended version of AMRlib. It would be interesting for us to test whether the data augmentation results presented here are coherent throughout the different parsing methods, in particular in using the most efficient parsing methods for AMR such as MBSE (see Lee et al. (2022)).

We would also like to conduct an exploration of errors similar to the one presented in Boritchev and Heinecke (2023) to be able to quantify and qualify the remaining percentages of mistakes. The goal then would be to use pre- and post-processing methods to accommodate these errors when possible.

We only worked with short questions, quiz-like, since we were not (yet) able to annotate corpora with longer questions or more complex types of questions. The questions in the additional corpora (QALD-9 and QUEREO) are given without a proper context. If we were to parse dialogues, coreference resolution and ellipsis resolution should be considered.

6 Further Work: Beyond English

The work presented in the current article only concerns English, since gold AMR data is only available for this language. Another problem is that the AMR3.0 training corpus is translated to other languages using machine translation, so errors in this translation may influence the results.

Even though AMR has explicitly not been developed to be an interlingua for multi-lingual processing, it is in fact used exactly for this. A manual translation of the AMR test corpus sentences into Chinese, German, Italian and Spanish is provided by LDC (LDC2020T07⁸). In order to annotate non-English text in AMR, two variants for multi-lingual AMR can be found in the literature: 1) annotating non-English sentences using a language specific set of concepts, i.e. instead of the (English) PropBank, concepts from language specific thesauri are used (e.g. Chinese AMR, (Li et al., 2016), Spanish (Miguelles-Abraira et al., 2018), Turkish (Oral et al., 2022) amongst others) or 2) English AMR graphs represent the meaning of non-English sentences (Damonte and Cohen, 2018; Blloshmi et al., 2020; Uhrig et al., 2021; Cai et al., 2021; Heinecke and Shimorina, 2022). We followed the latter approach by machine-translating the sentences of the AMR 3.0

⁸Damonte and Cohen (2020)

corpus into French and training baseline models using this translation. We used Google Machine Translation (Wu et al., 2016) and No Language Left Behind (NLLB, Costa-jussà et al. (2022)).

For the training of the French corpus we only finetuned MT5. In addition we created a multilingual model (based on MT5) by concatenating and shuffling the English and French training and validation corpora. In this case we have an English and a French sentence for each AMR graph. The first results look very similar to the results described in this paper as shown in table 4 for French on QALD-9. Table 5 shows the results for French (similar to table 3 for English).

Another approach we would like to explore is the transition from AMR to Uniform Meaning Representation (UMR) (Bonn et al., 2024). As UMR is designed to be “cross-linguistically plausible”, the multilanguage considerations are inherent to the UMR annotations, making them particularly interesting for our type of investigations.

LM	AMR 3.0	QALD-9
MT5 (fr)	74.8	81.4
MT5 (en-fr)	74.6	80.8

Table 4: French: Results of the AMR3.0 test corpus and the QALD-9 test corpus.

LM	lg.	train data	test data	
			AMR 3.0	QALD-9
MT5	French	baseline	74.8	81.4
		+ QUEREO	74.8 (± 0.0)	82.3 ($+0.9$)
		+ QALD-9	74.7 (-0.1)	84.4 ($+3.0$)
		+ QR. + Q9	75.0 ($+0.2$)	84.6 ($+3.2$)
MT5	En + Fr	baseline	74.6	80.8
		+ QUEREO	74.5 (-0.1)	81.6 ($+0.8$)
		+ QALD-9	74.9 ($+0.3$)	85.5 ($+4.7$)
		+ QR. + Q9	74.9 ($+0.3$)	85.5 ($+4.7$)

Table 5: French test results: Best figures for a test corpus with the same language model (MT5) in italics, best overall score in bold. QR stands short for QUEREO, Q9 stands for QALD-9.

References

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract Meaning Representation

for Sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

Rexhina Blloshmi, Rocco Tripodi, and Roberto Navigli. 2020. XL-AMR: Enabling Cross-Lingual AMR Parsing with Transfer Learning Techniques. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, page 2487–2500, Online. Association for Computational Linguistics.

Julia Bonn, Matthew J. Buchholz, Jayeol Chun, Andrew Cowell, William Croft, Lukas Denk, Sijia Ge, Jan Hajič, Kenneth Lai, James H. Martin, Skatje Myers, Alexis Palmer, Martha Palmer, Claire Benet Post, James Pustejovsky, Kristine Stenzel, Haibo Sun, Zdeňka Urešová, Rosa Vallejos, Jens E. L. Van Gysel, Meagan Vigus, Nianwen Xue, and Jin Zhao. 2024. [Building a broad infrastructure for uniform meaning representations](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2537–2547, Torino, Italia. ELRA and ICCL.

Maria Boritchev and Johannes Heinecke. 2023. Error exploration for automatic abstract meaning representation parsing. In *15th International Conference on Computational Semantics*.

Deng Cai, Xin Li, Jackie Chun-Sing Ho, Lidong Bing, and Wai Lam. 2021. [Multilingual AMR Parsing with Noisy Knowledge Distillation](#). In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2778–2789, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shu Cai and Kevin Knight. 2013. Smatch: an Evaluation Metric for Semantic Feature Structures. In *51st Annual Meeting of the Association for Computational Linguistics*, page 748–752, Sofia, Bulgaria. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Zoph Barret, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shane Shixiang Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincen 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](#). <https://arxiv.org/abs/2210.11416>.

Marta Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Mailard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barraud, Gabriel Mejia Gonzalez, Prangthip Hansanti,

- John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. [No Language Left Behind: Scaling Human-Centered Machine Translation](#).
- Marco Damonte and Shay Cohen. 2020. Abstract Meaning Representation 2.0 – Four Translations.
- Marco Damonte and Shay B. Cohen. 2018. Cross-lingual Abstract Meaning Representation Parsing. In *NAACL: Human Language Technologies*, pages 1146–1155, New Orleans, Louisiana, USA. Association for Computational Linguistics.
- Johannes Heinecke. 2023. [metAMoRphosED: a graphical editor for Abstract Meaning Representation](#). In *19th Joint ACL – ISO Workshop on Interoperable Semantic Annotation*, Nancy.
- Johannes Heinecke and Anastasia Shimorina. 2022. Multilingual Abstract Meaning Representation for Celtic Languages. In *Proceedings of the 4th Celtic Language Technology Workshop within LREC2022*, pages 1–6, Marseille. ELRA.
- Robert T. Kasper. 1989. [A flexible interface for linking applications to Penman’s sentence generator](#). In *Speech and Natural Language: Proceedings of a Workshop Held at Philadelphia, Pennsylvania, February 21-23, 1989*.
- Paul Kingsbury and Martha Palmer. 2002. From Tree-Bank to PropBank. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, pages 1989–1993, Las Palmas, Canary Islands, Spain. European Language Resources Association.
- Kevin Knight, Bianca Badarau, Laura Baranescu, Claire Bonial, Madalina Bardocz, Kira Griffitt, Ulf Hermjakob, Daniel Marcu, Martha Palmer, Tim O’Gorman, and Nathan Schneider. 2020. Abstract Meaning Representation (AMR) Annotation Release 3.0.
- Young-Suk Lee, Ramón Astudillo, Hoang Than Lam, Tahira Naseem, Florian Radu, and Salim Roukos. 2022. Maximum Bayes Smatch Ensemble Distillation for AMR Parsing. In *NAACL*, pages 5379–5392, Seattle, USA. Association for Computational Linguistics.
- Bin Li, Yoan Wen, Bu Lijun, Weiguang Qu, and Nianwen Xue. 2016. Annotating the Little Prince with Chinese AMRs. In *Proceedings of LAW X – The 10th Linguistic Annotation Workshop*, pages 7–15, Berlin, Germany. Association for Computational Linguistics.
- Noelia Migueles-Abraira, Rodrigo Agerri, and Arantza Diaz de Ilarraza. 2018. Annotating Abstract Meaning Representations for Spanish. In *The eleventh international conference on Language Resources and Evaluation*, pages 3074–3078, Marrakech, Maroc.
- Juri Opitz, Anette Frank, and Letitia Parcalabescu. 2020. [AMR similarity metrics from principles](#). *Transactions of the Association for Computational Linguistics*, 8(0):522–538.
- Elif Oral, Ali Acar, and Gülşen Eryigit. 2022. [Abstract meaning representation of Turkish](#). *Natural Language Engineering*, pages 1–30.
- Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106.
- Colin Raffel, Noam Shazeer, Adam Roberts, Lee Katherine, Sharan Narang, Matena Michael, 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*, 21:1–67.
- Sarah Uhrig, Yoalli Rezepka García, Juri Opitz, and Anette Frank. 2021. [Translate, then Parse! A strong baseline for Cross-Lingual AMR Parsing](#). In *Proceedings of the 17th International Conference on Parsing Technologies and the IWPT 2021 Shared Task on Parsing into Enhanced Universal Dependencies*, pages 58–64, Online. Association for Computational Linguistics.
- Ricardo Usbeck, Ria Hari Gusmita, Muhamad Saleem, and Axel-Cyrille Ngonga Ngomo. 2018. 9th challenge on question answering over linked data (qald-9). In *Joint proceedings of the 4th Workshop on Semantic Deep Learning (SemDeep-4) and NLIWoD4*.
- Sinh Trong Vu, Minh Le Nguyen, and Ken Satoh. 2022. Abstract meaning representation for legal documents: an empirical research on a human-annotated dataset. *Artificial Intelligence and Law*, pages 1–23.
- Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. [Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation](#).
- Linting Xue, Noa Constant, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. [mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer](#). In *NAACL*, pages 483–498. Association for Computational Linguistics.

FinerWeb-10BT: Refining Web Data with LLM-Based Line-Level Filtering

Erik Henriksson*
University of Turku
erik.henriksson@utu.fi

Otto Tarkka*
University of Turku
ohitar@utu.fi

Filip Ginter
University of Turku
figint@utu.fi

Abstract

Data quality is crucial for training Large Language Models (LLMs). Traditional heuristic filters often miss low-quality text or mistakenly remove valuable content. In this paper, we introduce an LLM-based line-level filtering method to enhance training data quality. We use GPT-4o mini to label a 20,000-document sample from FineWeb at the line level, allowing the model to create descriptive labels for low-quality lines. These labels are grouped into nine main categories, and we train a DeBERTa-v3 classifier to scale the filtering to a 10B-token subset of FineWeb. To test the impact of our filtering, we train GPT-2 models on both the original and the filtered datasets. The results show that models trained on the filtered data achieve higher accuracy on the HellaSwag benchmark and reach their performance targets faster, even with up to 25% less data. This demonstrates that LLM-based line-level filtering can significantly improve data quality and training efficiency for LLMs. We release our quality-annotated dataset, FinerWeb-10BT, and the codebase to support further work in this area.

1 Introduction

In recent years, the size of large language models (LLMs) and their training datasets has expanded tremendously, as companies and researchers strive to build increasingly capable models. In fact, if current trends continue, we may run out of human-generated text data within a decade (Villalobos et al., 2024). This has led to a growing interest in data quality over quantity: rather than only expanding datasets, researchers are exploring ways

to achieve high performance with smaller, cleaner datasets. Recent studies suggest that removing low-quality text from training data can improve model performance, even when the overall size of the dataset is reduced (Longpre et al., 2023).

Furthermore, training state-of-the-art (SOTA) language models requires significant computational resources, which are expensive and, depending on the power source, can contribute to climate change. For example, the carbon emissions from training GPT-3 have been estimated at 552 tCO₂e (Patterson et al., 2021), while Meta reports that training the 405 billion parameter Llama 3.1 emitted 8,930 tCO₂e (Meta-Llama, 2024). Smaller, but higher quality datasets will speed up training and, thus, high-quality data are necessary to train not only better models but also greener ones.

While several publicly available datasets are used for training LLMs, many recent datasets are still cleaned using simple heuristic filters, which often leave substantial amounts of low-quality text while potentially discarding clean text. Machine-learning techniques offer a promising alternative, as they enable models to identify patterns related to data quality. However, labeling data to train such models is a tedious and time-consuming process. In this paper, we address these issues by investigating the following research questions (RQs):

RQ1: How well can an LLM identify low-quality content missed by heuristic filters?

RQ2: Does LLM-based quality filtering of training datasets improve model performance?

To examine these questions, we analyze FineWeb, a dataset that claims to provide “the finest text data at scale” (Penedo et al., 2024). Using GPT-4o mini (OpenAI, 2024a), we label a 20,000-document sample from FineWeb, classifying each line as either *Clean* or belonging to one

*These authors contributed equally.

of several low-quality categories, such as *copyright notice*, *programming code*, or *formatting elements*. Instead of defining a label taxonomy ourselves, we allow the model to generate its own labels as needed, resulting in 547 unique low-quality labels. After refining these labels, we group them into nine broader categories for easier classification. Next, we train a DeBERTa-v3 (He et al., 2021) classifier using the labeled data to scale the filtering process. This classifier allows us to automatically detect low-quality content in a larger 10B-token sample of FineWeb. Finally, we evaluate the impact of LLM-based filtering by training GPT-2 models (Radford et al., 2019) on both the filtered and unfiltered datasets.

We release our quality-annotated dataset, *FinerWeb-10BT*, available at <https://huggingface.co/datasets/TurkuNLP/finerweb-10bt>. The code to replicate our experiments is also provided at <https://github.com/TurkuNLP/finerweb-10bt>.

2 Background

A recent survey by Albalak et al. (2024) discusses the many steps involved in selecting data for training LLMs, including language filtering, deduplication, removal of toxic or explicit content, and heuristic-based data quality filtering. Our focus here is on the latter two—data filtering and heuristic approaches—using an LLM-driven approach to refine data quality more precisely. As Albalak et al. (2024) note, there is no universal standard for “high-quality” data. In this work, we define it as human-written, continuous English text from the main content of a website, reflecting natural language use across diverse contexts and domains. Examples include core text from interviews, forum posts, news articles, blogs, and recipes. In contrast, low-quality content includes recurring elements like navigational menus, copyright notices, programming code, and metadata.

Given that LLMs require vast amounts of text data for training, the Internet has become a primary source for these data. Since 2008, CommonCrawl has collected a corpus of approximately 10 petabytes of web content (Baack, 2024). Despite its size, CommonCrawl is neither a complete nor fully representative sample of the Internet, but it serves as a foundational source for building refined datasets used in LLM training. Here, we focus on three major datasets sourced

from CommonCrawl: C4 (Raffel et al., 2023), RefinedWeb (Penedo et al., 2023), and FineWeb. These datasets use different preprocessing techniques to filter out unwanted material, each with its strengths and weaknesses. We discuss these datasets because their preprocessing methods are well-documented, which allows us to make meaningful comparisons.

All three datasets extract plaintext from HTML documents. C4 uses the WET files provided by CommonCrawl, which come with pre-extracted plaintext, whereas RefinedWeb and FineWeb use *trafilatura*¹ to extract text directly from HTML. Although *trafilatura* and similar tools remove much of the unwanted noise, further preprocessing is often required. For instance, Penedo et al. (2023) note that “many documents remain interlaced with undesirable lines” despite using *trafilatura*. Deduplication and language filtering are also important aspects of document cleaning but we do not focus on them in this paper, as they are specialized techniques not directly related to line-level text quality.

Existing filtering methods can be grouped into three levels based on their precision: document level, line level, and character level. By far the most common method is document level filtering, which removes entire documents based on simple rules. Examples include filtering documents with phrases like “lorem ipsum”, documents with fewer than three sentences, or documents with excessive repetition. Line level filtering targets specific lines within documents, removing lines that contain terms like “javascript”, consist solely of numbers, or fall below a certain length threshold. Character level filtering is less common and is only applied in one of the three datasets: in C4, citation markers commonly found in Wikipedia, such as “[1]” and “[citation needed]”, are removed.

Document level heuristic filtering is efficient for quickly removing large volumes of low-quality data, but it can result in the loss of substantial high-quality text. In contrast, line and character level filtering provide more precision by targeting specific content but they require significantly more computational resources at scale. Simple heuristics, such as removing lines that contain the word “javascript” can be hit or miss, sometimes discarding valuable data along with the low-quality content. Given the vast size of datasets like Com-

¹<https://trafilatura.readthedocs.io/en/latest/>

monCrawl, creating a simple filtering system that only removes undesirable content without impacting valuable data is nearly impossible. The filters that are used are also often dataset and language specific. For example, FineWeb applies a heuristic that removes documents where “the fraction of lines shorter than 30 characters is ≥ 0.67 ” (Penedo et al., 2024, p. 7), but this threshold was determined through extensive manual testing and is specific to that dataset.

An ideal quality filter would work across languages and datasets, avoiding trial-and-error by focusing on actual text quality rather than proxies like line length or keywords. It should also be efficient, removing only low-quality content while keeping valuable data intact. LLMs bring us closer to this goal: rather than using heuristics, they assess text quality directly, enabling granular filtering, even within mostly clean documents. Since LLMs are effective at producing fluent and readable text, they are likely well suited to identifying high-quality text across different languages and datasets. However, it should be noted that while SOTA LLMs are fluent in English and other high-resource languages, their performance in low-resource languages is consistently worse (Li et al., 2024). In this study, we only analyze English documents, and care should be taken before generalizing the results to other languages or multilingual datasets.

The use of LLMs for quality filtering is a relatively new approach, and best practices are still emerging. For instance, Dubey et al. (2024) utilize Llama 2 to assess the quality of web documents for training Llama 3, but details of their methodology are vague. The recent trend of withholding full training datasets for SOTA models has made it difficult to understand the extent to which LLMs are currently used in data preprocessing (Nguyen et al., 2024; Maini et al., 2024). Other efforts, such as those by Wettig et al. (2024), involve ranking documents based on quality using GPT-3.5, evaluating factors such as style, educational value, and factuality. Similarly, Llama 3 was used to create the FineWebEdu dataset by evaluating educational content quality, and Gunasekar et al. (2023) employ GPT-4 to annotate code datasets based on educational value.

Our approach differs from prior work by focusing on general-purpose data quality improvements rather than curating specialized datasets.

We aim to broadly enhance training data quality through LLM-driven filtering that removes low-quality lines with minimal manual intervention. This allows us to assess how automated filtering can improve training data and, ultimately, model performance in foundation model training.

3 Methods

Our data source is FineWeb (Penedo et al., 2024), a 15-trillion-token collection of English text sourced from CommonCrawl and preprocessed with standard heuristics. The preprocessing includes steps such as length thresholds, string matching, language and URL filtering, and deduplication. Despite these measures, the authors of FineWeb acknowledge that the dataset could benefit from further refinement. For more details on the preprocessing steps, see the original paper (Penedo et al., 2024). In our study, we use a 10B-token (15 million documents) sample from FineWeb, FineWeb-10BT².

Our preprocessing pipeline consists of several steps. First, we use GPT-4o mini (OpenAI, 2024a) to label a sample of 20,000 documents from FineWeb at the line level. The model is tasked with generating descriptive labels for each line, categorizing them as either high-quality (*Clean*) or into low-quality categories. This labeling process is data-driven, allowing the model to create a dynamic labeling scheme rather than relying on pre-defined categories. Previous research has shown that LLMs can be used to annotate data and create label taxonomies (Wan et al., 2024).

Next, we use OpenAI’s o1-preview model (OpenAI, 2024c) to group the numerous labels generated by GPT-4o mini into a smaller, more manageable set. This forms the basis of a classification system, which we use to train a small encoder-based classifier. This classifier scales the labeling process by assigning quality scores throughout the FineWeb-10BT dataset, enabling line-level filtering of low-quality content.

To evaluate our filtering, we train GPT-2 models (Radford et al., 2019) on both the cleaned and original versions of FineWeb-10BT. We compare model performances using the HellaSwag benchmark (Zellers et al., 2019), a widely used test for commonsense reasoning in language models. This allows us to assess whether the filtering improves

²<https://huggingface.co/datasets/HuggingFaceFW/fineweb/viewer/sample-10BT>

training data quality and model performance.

Given the complexity of Internet text data (Laippala et al., 2023), defining low-quality categories in advance is challenging. Our data-driven approach, by contrast, allows the LLM to dynamically create labels based on the content it encounters, rather than relying on fixed categories. We believe this approach enables a more flexible and detailed analysis of low-quality content in FineWeb compared to rule-based methods or pre-defined categorizations.

4 Experiments and results

4.1 Labeling FineWeb using GPT-4o mini

We begin by labeling a 20,000-document sample from FineWeb-10BT using the GPT-4o mini model. The model is prompted to classify each line as either *Clean* (high quality and suitable for training large language models) or assign a descriptive label if the line contains low-quality content, such as HTML tags or random symbols. Initially, the model generates its own descriptive labels, which are then added to a list for subsequent classification. As the model processes more documents, it selects labels from the existing list or creates new ones if necessary. To avoid bias from label order, the list is shuffled after each iteration.

We split the documents into batches of up to 15 consecutive lines. The model receives a prompt, a list of labels, and a batch of lines. Since the lines are consecutive, each one is evaluated in context, providing the model with more information for accurate labeling. For documents containing a single line longer than 200 characters, the line is split into segments of no more than 200 characters, using sentence-ending punctuation as the split point. This prevents output errors, which we observed when processing excessively long lines during preliminary tests. Segmenting these lines also enables more precise analysis.

This process results in quality labels for 328,472 lines. Of these, 274,343 lines (83%) are labeled as *Clean*. For low-quality lines, the model generates 547 unique descriptive labels. However, we find that many of these labels are assigned to one line only; in fact, 142 labels appear only once. Upon inspection, we notice many of the lines could be considered high-quality and, thus, to streamline the label set, we map all these infrequent labels to *Clean*. For the remaining labels, we take a sample of lines and manually verify that

they represent genuinely low-quality content. If the majority of lines for a particular label are of high quality, we remap that label to *Clean*. After this refinement, the number of descriptive labels is reduced to 382, with 45,205 lines (14%) classified as low-quality. Conversely, 86% of the dataset is now labeled *Clean*.

To visualize the distribution of these classes, we generate a 2D UMAP projection (McInnes et al., 2018) of the 50 most frequent label embeddings, created using the Stella-en-400M-v5 model (StellaEncoder, 2024) (see also Section 4.3 below). The UMAP projection reduces the original 1024-dimensional embeddings to 2D, as shown in Figure 1, with each dot scaled to represent the relative frequency of each class.

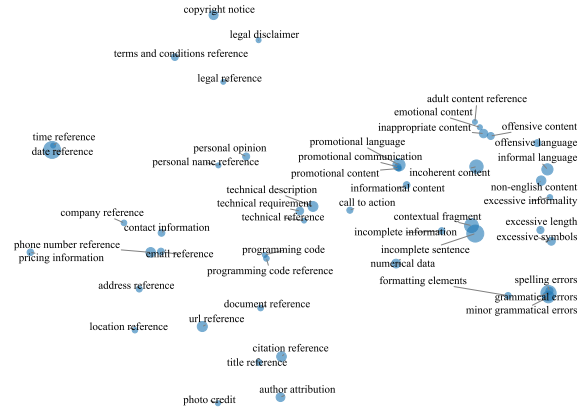


Figure 1: UMAP plot of embeddings of the 50 most frequent LLM-generated label names, created using the Stella-en-400M-v5 model.

Inspecting the plot, we observe that certain types of low-quality content tend to occupy distinct regions in the space. For instance, legal texts appear in the top-left, adult and toxic content in the top center-right, and bibliographic references near the bottom. Contact information, such as times, dates, and phone numbers, is loosely grouped on the left, while technical content, like programming code, appears in the center. These patterns suggest that the LLM-generated labels capture meaningful line quality distinctions and form a useful basis for our final class set.

4.2 Grouping the labels

The next step in our pipeline is to group the 382 detailed labels into a more concise set of broader, more manageable categories, which simplifies training the encoder classifier. We use Ope-

nAI’s o1-preview, a newly released “reasoning” model (OpenAI, 2024b), to organize the labels. We instruct the model to create clear, distinct categories that assign each label to only one group. The goal is to produce a set of classes that the classifier can learn and differentiate easily.

Category	Lines	%
<i>Clean</i>	283,267	86.24
<i>Formatting, Style & Errors</i>	13,150	4.00
<i>Bibliographical & Citation References</i>	8,768	2.67
<i>Promotional & Spam Content</i>	7,339	2.23
<i>Contact & Identification Information</i>	3,898	1.19
<i>Navigation & Interface Elements</i>	3,327	1.01
<i>Technical Specifications & Metadata</i>	3,298	1.00
<i>Legal & Administrative Content</i>	2,992	0.91
<i>Offensive or Inappropriate Content</i>	2,433	0.74
Total	328,472	100

Table 1: Label categories and the number of lines in each category.

After manually inspecting the output, we find that the groupings are mostly accurate, though some manual corrections are necessary. For example, the model occasionally fails to assign all labels or places some labels into multiple categories. After fixing these issues, we finalize a classification scheme with 9 broader categories, as shown in Table 1.

To verify that the labels match human intuition, we conduct a manual inter-annotator agreement (IAA) evaluation on a random sample of 50 documents (726 lines). Two human annotators, familiar with the 9-label class set, assess whether they agree or disagree with the LLM-generated labels. In cases of disagreement, they provide corrected labels. We compute Cohen’s Kappa scores comparing human ratings with the LLM’s for both the full label set and a simplified binary classification (*Clean* vs. *Non-clean*).

	A1	A2	Avg.
All labels	0.79	0.60	0.70
Clean vs. Non-clean	0.78	0.67	0.73

Table 2: Cohen’s Kappa scores for human annotators (A1 and A2) vs. the GPT-4o mini generated labels (LLM).

As shown in Table 2, Cohen’s Kappa for the full

label set is 0.788 for Annotator 1 (A1) and 0.604 for Annotator 2 (A2), with an average of 0.70, indicating moderate to substantial agreement. For the binary classification, Kappa scores improve slightly, with A1 at 0.78 and A2 at 0.67, averaging 0.73. This suggests that while agreement varies, the LLM-based classification generally produces acceptable labels for the FineWeb texts.

These results address RQ1, which examines how well an LLM can identify low-quality content that heuristic filters miss. The LLM’s classifications align well with those of human annotators, showing that it succeeds to detect low-quality lines overlooked by earlier heuristic methods applied to FineWeb data. While there is some variability in the IAA scores, the overall performance supports our LLM-driven approach.

4.3 Training a classifier

To scale our labeling process for the FineWeb-10BT dataset, we use encoder-based models, which are faster, more cost-effective, and often better suited to classification than large generative LLMs. We experiment with four models: DeBERTa-v3 (base and large variants) (He et al., 2021), Stella-en-400M-v5 (currently the top model of its size for English text clustering on the MTEB leaderboard (Muennighoff et al., 2023)³), and XLM-RoBERTa-base (Conneau et al., 2019). The first three models are English-only, while XLM-RoBERTa is multilingual.

For line-by-line classification, we first extract individual lines from the documents, treating each as a separate example. The data is then shuffled and split into training (70%), development (10%), and test (20%) sets using stratification. We add a classification head to each model to generate probabilities across the 9 classes for each line and fine-tune both the classification head and base model. Preliminary tests showed that this approach yielded better results than training only the classification head with a frozen base model.

For training, we use bfloat16 precision, a learning rate of 1e-5, and a batch size of 16. Early stopping is applied with a patience of 5 based on evaluation loss, with a maximum of 5 epochs; however, models typically converge after the first epoch. We also apply label smoothing (0.1) to the cross-entropy loss to improve generalization. Training is done on a single A100 GPU.

³<https://huggingface.co/spaces/mteb/leaderboard>

	μ F1	M F1	Clean		
			P	R	F1
DeBERTa-v3-base	0.81	0.66	0.88	0.91	0.90
DeBERTa-v3-large	0.81	0.65	0.87	0.92	0.89
Stella-en-400M-v5	0.81	0.67	0.87	0.92	0.89
XLM-RoBERTa-base	0.80	0.63	0.86	0.92	0.89

Table 3: Comparison of Classifiers on Multiclass Classification using the held-out test set. μ F1: Micro F1, M F1: Macro F1, P: Precision, R: Recall, F1: F1 score for the *Clean* class.

Table 3 presents the evaluation results of the models on the test set. We report micro and macro F1 scores for all classes, along with precision, recall, and F1 for the *Clean* class. The results show that the models perform similarly, with micro F1 scores ranging between 0.80 and 0.81, and macro F1 scores between 0.63 and 0.67. For the *Clean* class, precision ranges from 0.86 to 0.88, recall from 0.91 to 0.92, and F1 between 0.89 and 0.90. These metrics indicate strong performance in distinguishing between high- and low-quality content, though the lower macro F1 score suggests some classes are less easily distinguishable. Additionally, newer or larger models do not significantly improve performance. Thus, for subsequent analyses, we select the DeBERTa-v3-base model.

Confusion Matrix with Percentages									
True Labels	Bibliographical and Citation References	Contact and Identification Information	Formatting, Style, and Errors	Legal and Administrative Content	Navigation and Interface Elements	Offensive or Inappropriate Content	Promotional and Spam Content	Technical Specifications and Metadata	Clean
	78	1	4	0	1	0	1	1	14
	7	65	3	1	2	0	3	1	18
	2	1	58	0	1	0	7	2	28
	5	1	3	66	0	0	1	0	23
	6	3	11	1	48	0	6	0	26
	0	0	2	0	0	54	5	0	39
	1	1	10	0	1	2	55	3	25
	3	1	6	0	1	0	6	58	25
	2	1	2	0	0	0	2	1	91
Predicted Labels									
Bibliographical and Citation References Contact and Identification Information Formatting, Style, and Errors Legal and Administrative Content Navigation and Interface Elements Offensive or Inappropriate Content Promotional and Spam Content Technical Specifications and Metadata Clean									

Figure 2: Confusion matrix of predictions from our line quality classifier on the test set.

To further examine the performance of the classifier and spot common misclassifications, we evaluate its predictions on the held-out test set using DeBERTa-v3-base and display the results in a confusion matrix (Figure 2). Most misclassifications fall into the *Clean* class, indicating strong separation between the other classes. The least

distinct class is *Offensive or Inappropriate Content*, likely due to the inherent difficulty in defining clear boundaries for offensive material in LLM training datasets. In contrast, *Bibliographical and Citation References* stands out as the most distinct class, likely due to its easily recognizable formatting and content.

We note that it is preferable for the classifier to err on the side of labeling low-quality lines *Clean* (as shown in the confusion matrix and evaluation scores) rather than mistakenly tagging high-quality lines as low-quality. This bias helps reduce the risk of discarding valuable data from the dataset.

4.4 Cleaning FineWeb

Given our classifier’s promising evaluation results, we now label the 10B-token subset of FineWeb using our DeBERTa-v3-base classifier. For this task, we simplify to binary classification by focusing only on the probability of the *Clean* class versus all other classes combined, where probabilities closer to 1 indicate high-quality content.

Although the classifier performs well, the *Clean* class makes up 86% of the data, which may cause the model to produce overconfident predictions for this class. To correct for this imbalance, we apply Platt scaling (Platt et al., 1999) to adjust the predicted probabilities, aiming for a more accurate reflection of the true probability distribution and more reliable thresholding. Specifically, we train a Platt logistic regression model on the held-out test set and apply it on top of the classifier when predicting quality scores for the FineWeb-10BT dataset.

We predict the quality labels for the FineWeb-10BT dataset in shards of 100,000 documents. Within each shard, we process batches of 128 lines, grouping lines by length to speed up processing. We then add a “quality_score” key to each document, with each item scored from 0 to 1 to four decimal places.

Figure 3 shows a histogram of the quality scores for a 1-million-line sample from FineWeb-10BT, with calibrated probabilities binned in 10% intervals on a logarithmic scale. The distribution is bimodal, with most lines receiving high-quality scores. About 75% of lines score above 0.90, while 8% score below 0.50. Most of the data is concentrated in the highest quality bin (90–100%), with a smaller cluster confidently assigned very

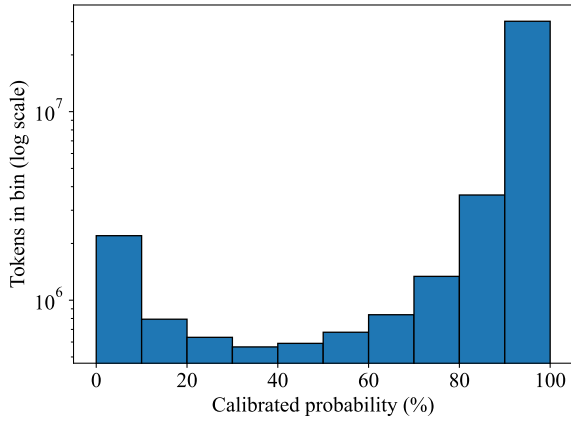


Figure 3: Quality probabilities for a 1M-line sample from FineWeb-10BT, binned in 10% intervals (log scale). A total of 8% of lines fall below the 0.50 quality threshold, and 25% fall below the 0.90 threshold.

low scores, indicating that the classifier effectively separates high-quality from low-quality lines.

Table 4 shows examples of lines with the highest and lowest quality scores according to our classifier. The highest-scoring lines are coherent, context-rich sentences, while the lowest-scoring lines contain metadata, copyright symbols, tags, and formatting artifacts, demonstrating that the method performs as intended.

4.5 Evaluation with GPT-2 and HellaSwag

Finally, we evaluate our data cleaning process by pre-training small GPT-2 models (124M parameters) on three versions of the dataset: (1) the original 10B-token sample from FineWeb, (2) a filtered version with a 0.50 quality score threshold, reducing the dataset by 8%, and (3) a version with a 0.90 quality score threshold, reducing data by 25%. The training code is adapted from Khajavi (2024), with modifications specific to our experimental setup.

The models are trained for 18,994 steps (a single epoch on the full FineWeb-10BT dataset) using four A100 GPUs. Every 200 steps, we evaluate model performance on the HellaSwag benchmark (Zellers et al., 2019), which is widely used to assess the ability of language models to complete sentences in commonsense reasoning contexts. To account for inherent randomness, we repeat the training on all datasets five times each, with each run lasting approximately 5 hours and 30 minutes.

Figure 4 shows the evaluation results, which

Line	Score
Lines with highest quality scores	
She hopes taking part in the 5K will encourage others to become or stay active.	0.9674
I'd love it if you'd visit and give me your impressions and/or suggestions.	0.9659
We aim to make the ceremony an enjoyable celebration.	0.9657
prayerfully seek peace for our partners in Nigeria.	0.9655
I loved the way this shirt looked and thought it would be cool to wear it.	0.9655
Lines with lowest quality scores	
Also published as US20040168193	0.0057
Tags: Anglesey, Beach, General, Landscape, Landscape / travel, Lighthouse, Llanddwyn, Sea, Sunrise, Wales, Water	0.0056
FOR IMMEDIATE RELEASE PRESS RELEASE #MR12-003881	0.0055
©Sunwest Bank Equal Housing Lender Member FDIC	0.0051
- ©- copyright & copy; or & #169; or & #xA9;	0.0050

Table 4: Examples of highest and lowest quality lines from a 1M-line FineWeb-10BT sample, with their probabilities of being *Clean*.

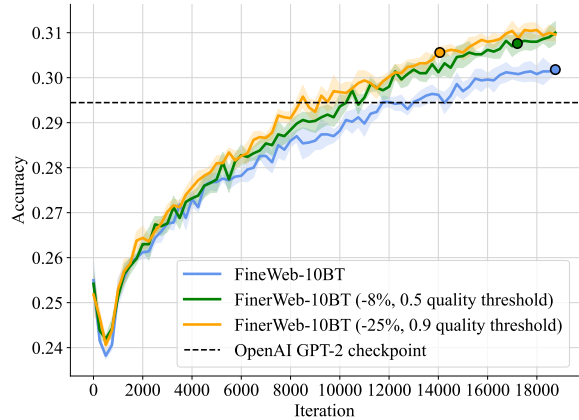


Figure 4: Average HellaSwag accuracy over 5 runs for three models: the original FineWeb-10BT and two cleaned versions with quality thresholds of 0.50 (8% data reduction) and 0.90 (25% data reduction). Dot markers indicate epoch ends for each dataset run. GPT-2 (124M) checkpoint accuracy is shown for reference.

indicate a clear positive impact from our data cleaning process. Models trained on the cleaner *FinerWeb-10BT* datasets—both the 8% and 25% reduced versions—consistently outperform those

trained on the original FineWeb-10BT data. By the end of 18,994 training steps, both cleaned versions show an average HellaSwag evaluation score that is 0.1 points higher than that of the original dataset. This improvement is robust, as shown by the shaded areas around the lines, representing standard deviations that suggest the effect is unlikely due to random variation across runs.

Additionally, both cleaned models achieve slightly higher HellaSwag accuracy than the original FineWeb-10BT model at their respective epoch ends, as indicated by the colored dots in the plot. Remarkably, both models reach the original dataset’s highest score approximately 6k steps earlier, a 32% reduction in training time. This means a reduction of roughly 1 hour and 45 minutes, based on our 5 hour 30 minute run time per training round. Interestingly, the 25% reduced dataset shows a slight edge over the 8% cleaned data, although the difference is minimal; both clean models ultimately reach an average HellaSwag score of 0.31 within the same number of steps. This suggests that a more aggressive data cleaning strategy could be worth exploring in future work. In summary, our data cleaning process produces models that (1) reach target accuracy faster and (2) achieve higher accuracy within the same training time, addressing our RQ2.

5 Discussion

The labels generated by GPT-4o mini reveal both the quantity and types of low-quality lines that remain in FineWeb. The largest categories include lines with grammatical errors, poor formatting, and incomplete sentences, along with recurring items like time stamps, legal jargon, and promotional content. While these elements do not necessarily reduce dataset quality (a good language model should recognize items like copyright notices or phone numbers), our evaluation shows that reducing their prevalence improves both accuracy and training efficiency. These findings suggest that more precise control over the types and proportions of low-quality data included could further benefit model performance. Even when simplified to binary classification, our LLM-driven approach clearly outperforms heuristic methods in enhancing dataset quality.

Specifically, our evaluation on GPT-2 using HellaSwag shows that with less but cleaner data, the model achieves comparable or even slightly

better accuracy. While GPT-2 is small relative to SOTA models, our results provide strong evidence that LLM-based data filtering can reduce training time and save energy. Although we tested our method on a small, English-only dataset, this data-driven approach to quality filtering is easily adaptable to other datasets and languages, although low-resource language may suffer from worse LLM performance.

Using an LLM as a judge of text quality introduces some bias, as the model’s training data and design choices influence the resulting labels. For example, mature SOTA LLMs have strong in-built safety features that prevent them from generating harmful or offensive content. In our case, we observe that GPT-4o mini sometimes labels mild expletives, such as “shut up”, as toxic, reflecting an overly sensitive filter for offensive language. As described in Sections 4.1 and 4.2 we made some manual adjustments to the LLM labeling to account for such biases. Also, the line between low-quality and high-quality is naturally vague, which introduces noise into the data. In future work, we plan to experiment with different models and adjust our prompts to further improve this filtering approach.

6 Conclusion

In this paper, we propose a novel approach to improving the quality of large-scale language model training datasets through fine-grained, line-level filtering with large language models (LLMs). We first used GPT-4o mini to label a sample from the FineWeb dataset, generating detailed labels that captured low-quality content often overlooked by heuristic filters, addressing our first research question (RQ1). These labels were grouped into broader categories using OpenAI’s o1-preview model, followed by training a DeBERTa-v3 classifier to scale the filtering across FineWeb-10BT. Our experiments demonstrate that this LLM-driven filtering pipeline improves model performance (addressing RQ2), as GPT-2 models trained on the filtered dataset achieved higher HellaSwag accuracy with up to 25% less data than those trained on the original FineWeb-10BT dataset.

These findings suggest that traditional heuristic filters may not be sufficient and that more sophisticated data preprocessing methods are necessary, especially as we face challenges like data scarcity and environmental concerns. Our approach con-

tributes to the emerging field of LLM-based data preprocessing, offering a promising avenue for improving training efficiency and model performance.

In future work, we plan to refine our pipeline by broadening the labeling scheme to provide a more comprehensive description of document contents. We will also experiment with more nuanced filtering approaches, moving beyond simple score-based thresholds, and compare against baselines such as random data reduction to further validate our filtering method. We also plan to test Llama-style models and other architectures to see how our findings scale to newer LLMs. Further evaluations and statistical testing will help strengthen the reliability of our results. Finally, we plan to extend our method to other datasets and languages.

Acknowledgments

This project has received funding from the European Union’s Horizon Europe research and innovation programme under Grant agreement No 101070350 and from UK Research and Innovation (UKRI) under the UK government’s Horizon Europe funding guarantee [grant number 10052546]. The contents of this publication are the sole responsibility of its authors and do not necessarily reflect the opinion of the European Union.

This work was supported by the Research Council of Finland.

Computational resources for this study were provided by CSC — IT Center for Science.

References

- Alon Albalak, Yanai Elazar, Sang Michael Xie, Shayne Longpre, Nathan Lambert, Xinyi Wang, Niklas Muennighoff, Bairu Hou, Liangming Pan, Hae-won Jeong, Colin Raffel, Shiyu Chang, Tatsunori Hashimoto, and William Yang Wang. 2024. A survey on data selection for language models.
- Stefan Baack. 2024. A critical analysis of the largest source for generative ai training data: Common crawl. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency (FAccT’24)*.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yearly, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kam-badur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Celebi, Patrick Al-rassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthi, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collob, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet,

- Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuwei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkan Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damla, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermsoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vitor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojuan Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. 2024. The Llama 3 Herd of Models.
- Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. 2023. Textbooks are all you need.
- Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTaV3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing.
- Matin Khajavi. 2024. <https://github.com/matinkhajavi/gpt-from-scratch>. <https://github.com/MatinKhajavi/GPT-from-scratch>.
- Veronika Laippala, Samuel Rönqvist, Miika Oinonen, Aki-Juhani Kyröläinen, Anna Salmela, Douglas

- Biber, Jesse Egbert, and Sampo Pyysalo. 2023. Register identification from the unrestricted open web using the corpus of online registers of english. *Language Resources and Evaluation*, 57.
- Zihao Li, Yucheng Shi, Zirui Liu, Fan Yang, Ali Payani, Ninghao Liu, and Mengnan Du. 2024. Language ranker: A metric for quantifying llm performance across high and low-resource languages.
- Shayne Longpre, Gregory Yauney, Emily Reif, Katherine Lee, Adam Roberts, Barret Zoph, Denny Zhou, Jason Wei, Kevin Robinson, David Mimno, and Daphne Ippolito. 2023. A pretrainer’s guide to training data: Measuring the effects of data age, domain coverage, quality, & toxicity.
- Pratyush Maini, Skyler Seto, Richard Bai, David Grangier, Yizhe Zhang, and Navdeep Jaitly. 2024. Rephrasing the web: A recipe for compute and data-efficient language modeling. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14044–14072, Bangkok, Thailand. Association for Computational Linguistics.
- Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. 2018. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861.
- Meta-Llama. 2024. Model card. Website.
- Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2023. Mteb: Massive text embedding benchmark.
- Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2024. CulturaX: A cleaned, enormous, and multilingual dataset for large language models in 167 languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4226–4237, Torino, Italia. ELRA and ICCL.
- OpenAI. 2024a. Gpt-4o-mini.
- OpenAI. 2024b. Learning to reason with llms.
- OpenAI. 2024c. Openai o1-preview.
- David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluís-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. 2021. Carbon emissions and large neural network training.
- Guilherme Penedo, Hynek Kydlíček, Loubna Ben al-lal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. The fineweb datasets: Decanting the web for the finest text data at scale.
- Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon llm: Outperforming curated corpora with web data, and web data only.
- John Platt et al. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.
- 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. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer.
- StellaEncoder. 2024. stella_en_400m_v5.
- Pablo Villalobos, Anson Ho, Jaime Sevilla, Tamay Besiroglu, Lennart Heim, and Marius Hobbhahn. 2024. Will we run out of data? limits of llm scaling based on human-generated data.
- Mengting Wan, Tara Safavi, Sujay Kumar Jauhar, Yujin Kim, Scott Counts, Jennifer Neville, Siddharth Suri, Chirag Shah, Ryen W. White, Longqi Yang, Reid Andersen, Georg Buscher, Dhruv Joshi, and Nagu Rangan. 2024. Tnt-llm: Text mining at scale with large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD ’24*, page 5836–5847, New York, NY, USA. Association for Computing Machinery.
- Alexander Wettig, Aatmik Gupta, Saumya Malik, and Danqi Chen. 2024. QuRating: Selecting high-quality data for training language models.
- Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence?

Margins in Contrastive Learning: Evaluating Multi-task Retrieval for Sentence Embeddings

Tollef Emil Jørgensen

Department of Computer Science
Norwegian University of Science and Technology
Trondheim, Norway
tollefe.jorgensen@ntnu.no

Jens Breitung

Department of Computer Science
RWTH Aachen University
Aachen, Germany
jens.breitung@rwth-aachen.de

Abstract

This paper explores retrieval with sentence embeddings by fine-tuning sentence-transformer models for classification while preserving their ability to capture semantic similarity. To evaluate this balance, we introduce two opposing metrics – polarity score and semantic similarity score – that measure the model’s capacity to separate classes and retain semantic relationships between sentences. We propose a system that augments supervised datasets with contrastive pairs and triplets, training models under various configurations and evaluating their performance on top- k sentence retrieval. Experiments on two binary classification tasks demonstrate that reducing the margin parameter of loss functions greatly mitigates the trade-off between the metrics. These findings suggest that a single fine-tuned model can effectively handle joint classification and retrieval tasks, particularly in low-resource settings, without relying on multiple specialized models.

1 Introduction

Tasks like text classification and semantic textual similarity (STS) are helpful for various applications, including retrieval through clustering, zero-shot categorization (Yin et al., 2019), and efficient few-shot classification with limited data (Tunstall et al., 2022). Traditionally, models addressing these tasks ranged from rule-based systems to deep learning architectures (Tai et al., 2015; Minaee et al., 2021; Li et al., 2022), with recent transformer-based models dominating the field (Joulin et al., 2017; Howard and Ruder, 2018; Devlin et al., 2019; Raffel et al., 2020). However, optimizing sentence embeddings for multiple objectives remains a challenge. In this work, we investigate the hypothesis

that training sentence-transformer models with two opposing objectives – semantic similarity and polarity – enables models that can be fine-tuned for downstream tasks while preserving their ability to capture semantic similarity. We argue that this approach is beneficial for obtaining more nuanced embeddings, e.g., for domain-specific classification and clustering, especially supporting low-resource settings with a single model capable of both. To evaluate the performance of our models on these dual objectives, we introduce two metrics:

Polarity Score (\mathcal{P}) measures the model’s classification performance by assessing how well it predicts sentence polarity (e.g., positive vs. negative sentiment). The higher the score, the more accurately the model distinguishes between classes.

Semantic Similarity Score (\mathcal{S}) quantifies how well the model retains semantic relationships between sentences by comparing the cosine similarity of sentence embeddings generated by our fine-tuned model to a reference model.

Both metrics are described in detail in Section 3.1. Experiments are conducted on (1) SST-2, Stanford Sentiment Treebank (Socher et al., 2013), a binary sentiment dataset, and (2) A dataset with sarcastic news headlines (Misra and Arora, 2023). We opted for binary datasets to efficiently verify the importance of the *margin* in contrastive learning. The remainder of this paper is structured as follows: Section 2 discusses related work. Section 3 introduces the datasets, data generation, metrics, models, and training details. Section 4 presents experimental results and Section 5 discussions. Finally, conclusions and plans for future work are in Section 6.

Code for the system is available on GitHub.¹

¹<https://github.com/tollefj/margins-contrastive>

2 Related Work

Related research is based mainly on developments within word and sentence embeddings. Commonly used embedding techniques include word2vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014), and ELMo (Peters et al., 2018). In the realm of sentence embeddings, early methods involved concatenation and aggregation of word embeddings to produce a sentence representation (Le and Mikolov, 2014; Joulin et al., 2017). However, more recent research has focused on developing specialized models to encode sentence representations, as exemplified by systems like InferSent (Conneau et al., 2017), universal sentence encoder (Yang et al., 2020), sentence-transformers (SBERT) (Reimers and Gurevych, 2019) and SimCSE (Gao et al., 2022). SBERT is trained using a pre-trained BERT model to learn the representations of a given sentence. While techniques and setups vary, an example of a training procedure is by providing triplets forming (*anchor sentence*, *positive*, *negative*), where the model attempts to maximize the distance between the anchor and the *negative* (dissimilar sentence), while minimizing the distance between the anchor and the *positive* (similar) sentence. This methodology provided efficient models for STS (Agirre et al., 2013; Reimers and Gurevych, 2019; Gao et al., 2022; Tunstall et al., 2022; Li et al., 2023; Wang et al., 2024). Several datasets and benchmarks have been published for STS since the SemEval shared task (Agirre et al., 2013), including the *STS Benchmark* (Cer et al., 2017), *SICK* (Marelli et al., 2014), and *BIOSSES* (Soğancıoğlu et al., 2017), all of which are now found in the Massive Text Embedding Benchmark (MTEB) (Muennighoff et al., 2022). Transformer models have excelled at the task, as is shown in the tables on HuggingFace’s leaderboard for the evaluation.² At the time of experiments, the *GTE* (Li et al., 2023) and *E5* (Wang et al., 2024) series of models were of particular interest given their strong performance to size ratio.

3 Methods and Data

This section includes information on datasets, evaluation metrics, baseline models, loss functions, example generation, and the fine-tuning pipeline. We mainly use two data sources for evaluation, although the provided system is generalizable to

any data source for binary classification. Figure 1 shows an overview of system components.

SST-2 The Stanford Sentiment Treebank (Socher et al., 2013) is widely used for binary classification tasks and is implemented in the GLUE benchmark (Wang et al., 2019). It consists of a train/test/validation split with 67,349/1821/872 samples respectively. However, the labels for the test split are hidden and can only be evaluated by submissions to GLUE.³ We use the validation split for presented results.

Sarcastic Headlines The “News Headlines Dataset for Sarcasm Detection” (Misra and Arora, 2023) contains 28,619 news headlines from *HuffPost* (non-sarcastic) and *The Onion* (sarcastic). Misra and Arora claims this to guarantee high-quality labels. The data is split in a 90:10 train/test ratio with a deterministic seed (0).

Additionally, results for the best-performing fine-tuning configuration are presented using the SentEval toolkit (Conneau and Kiela, 2018) on movie reviews, product reviews, subjectivity status, opinion-polarity, question-type classification, and paraphrase detection in Section 4.1.

3.1 Evaluation

For a given sentence s , the model M retrieves the k most similar sentences, denoted as s_1^M, \dots, s_k^M , based on the cosine similarity from a query sentence. The retrieved sentences are evaluated on two criteria: polarity and semantic similarity.

Polarity Score (\mathcal{P})

To evaluate if the model predicts sentences with the same polarity as the input, we compute a weighted average polarity score over the k predictions based on the polarity of s , $\mathcal{P}(s)$. Formally, the polarity score is defined as:

$$\mathcal{P}_M(s) := \sum_{i=1}^k w_i \cdot \text{pol}(s_i^M) \quad \text{where} \quad (1)$$

$$\text{pol}(s_i^M) := \begin{cases} 1 & \text{if } \text{pol}_s = \text{pol}_{s_i^M}, \\ 0 & \text{otherwise.} \end{cases}$$

To account for ranking in the top- k , we choose a linear discounting strategy, scaling the i -th weight:

$$w_i := \frac{2(k+1-i)}{k(k+1)}.$$

²<https://huggingface.co/spaces/mteb/leaderboard>

³<https://gluebenchmark.com/leaderboard>

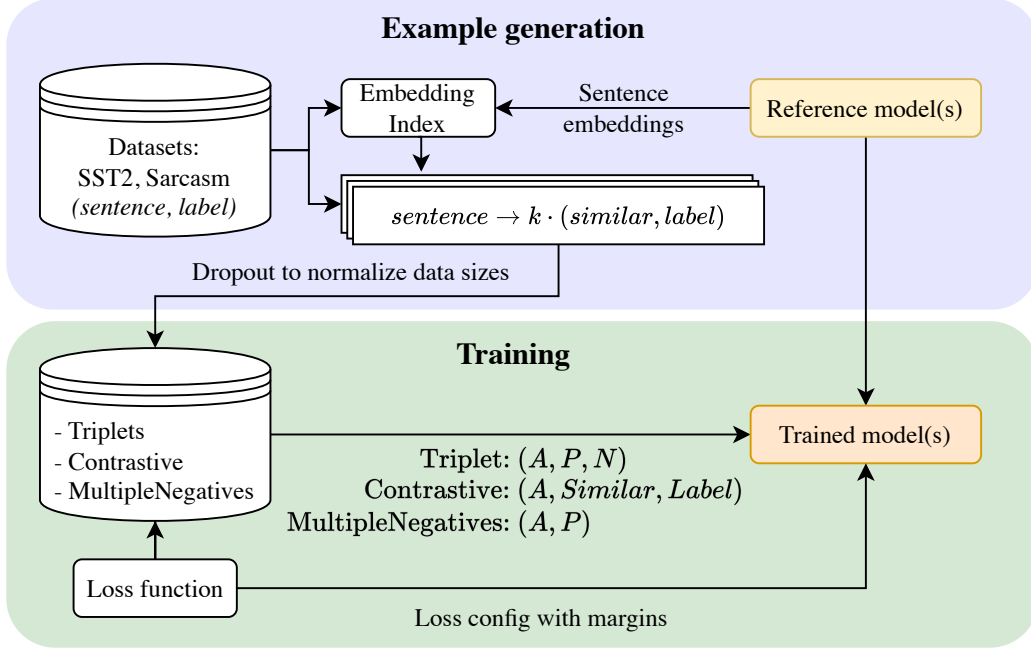


Figure 1: High-level system components of example generation and training. Sentences in the datasets are embedded and stored in an index, where k are retrieved to generate similarity-based examples corresponding to the loss functions. A dropout is added as generation varies between, e.g., triplets and contrastive pairs. Finally, a model is trained for each loss function and margin configuration.

A score near one indicates that most predictions share the input’s polarity.

Semantic Similarity Score (S)

The Semantic Similarity Score measures the cosine similarity between the predicted sentences from model M and the baseline model R . Given x_i as the embedding for sentence s_i , the cosine similarity is defined as:

$$\cos_sim(s_1, s_2) := \frac{x_1 \cdot x_2}{||x_1|| \cdot ||x_2||}$$

The semantic similarity score $S_M(s)$ for model M is then:

$$S_M(s) := \sum_{i=1}^k w_i \cdot \cos_sim_R(s, s_i^M) \quad (2)$$

The weights w_i are reused from the polarity score. A similarity score close to the reference model’s score $S_R(s)$ indicates that the predictions remain semantically aligned with the input sentence.

3.2 Baseline Models

The models in Table 1 are selected based on popularity and performance versus size. Data is sourced from the MTEB leaderboard (Muennighoff et al.,

2022). We select the commonly used sentence-transformer model, *all-MiniLM-L6-v2* (Reimers and Gurevych, 2019) – referred to as *MiniLM-6*, along with the better performing models *GTE-base/small* (Li et al., 2023) and the *E5-small-v2* (Wang et al., 2024). Retrieval performance is evaluated by constructing two embedding sets: *target embeddings* derived from the test set and *source embeddings* sampled from the training set. The source embeddings are chosen to be five times the size of the test set, providing an adequate evaluation pool while limiting the number of comparisons. For example, if the test set contains 1,000 sentences, the source set will contain 5,000 sentences randomly sampled from the training data.

From the sampled data, we compare retrieval performance to the top- k retrieved sentences by adjusting k , as shown in Figure 2. Increasing k slightly decreases performance, as larger retrieval sets are more likely to include less relevant sentences. However, we wish to keep a relatively high amount of retrieved sentences to identify model improvements (e.g., a higher fraction of returned sentences should be relevant). Based on these observations, we select $k = 16$ as a practical value.

Model	Size	Embedding MB dimension	STSBenchmark reported avg	SST-2		Sarcastic	
				\mathcal{P}	\mathcal{S}	\mathcal{P}	\mathcal{S}
E5-small-v2	130	768	85.95	81.5 _{23.7}	85.5 _{1.7}	71.4 _{21.2}	83.4 _{1.5}
GTE-base	220	768	85.73	80.4 _{22.6}	83.7 _{1.4}	67.4 _{20.7}	81.4 _{1.6}
GTE-small	70	384	85.57	77.8 _{22.2}	84.8 _{1.4}	66.8 _{20.6}	82.5 _{1.6}
MiniLM-6	90	384	82.03	63.0 _{21.9}	46.6 _{7.4}	63.8 _{20.2}	42.3 _{5.6}

Table 1: Sentence-transformer baseline model selection and performance ($k = 16$) for polarity (\mathcal{P}) and semantic similarity (\mathcal{S}) on SST-2 and sarcastic headlines. Standard deviation subscripted.

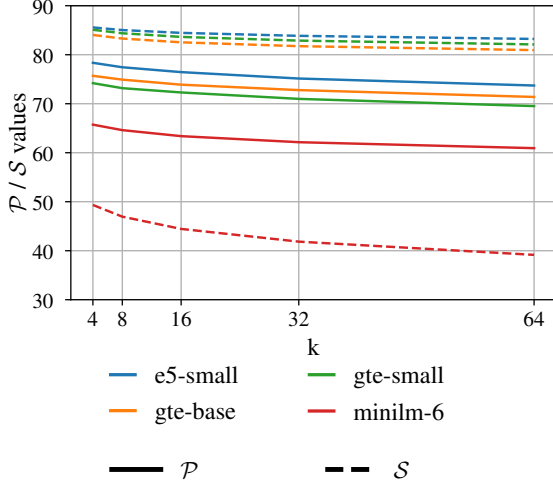


Figure 2: Baseline models with average performance across both datasets when retrieving the k nearest matches. Solid lines: \mathcal{P} , dotted lines: \mathcal{S} .

3.3 Contrastive Loss Functions

To assess the embedding quality, models are trained with different loss function configurations implemented in the Sentence-Transformers library (Reimers and Gurevych, 2019). However, not all losses can support our constraints of multiple objectives, and we constrain this study to Triplet-Loss (Schroff et al., 2015), MultipleNegativesRankingLoss (Henderson et al., 2017), OnlineContrastiveLoss and ContrastiveLoss (Hadsell et al., 2006). These require different inputs related to how the model assesses the similarity between input sentences.

TripletLoss consists of triplets of sentences (A, P, N) where A is the *anchor*, P is similar to the anchor, and N is dissimilar. We set the P to the corresponding positive example (1) in binary classification and N to the negative example (0). The loss becomes, with E_x denoting the embedding: $\max(|E_A - E_P| - |E_A - E_N| + \lambda, 0)$, where λ

is the margin, specifying the minimum separation between A and N .

MultipleNegativesRankingLoss consists of sentence pairs, assuming (a_i, p_i) pairs as positive and (a_i, p_j) pairs for $i \neq j$ as negatives. It calculates the loss by minimizing the negative log-likelihood for softmax-normalized scores, encouraging positive pairs to have higher similarity scores than negative pairs.

(Online)ContrastiveLoss consists of $\{0, 1\}$ -labelled tuples (Anchor, Sentence) where the label indicates whether $|E_A - E_S|$ is to be maximized, indicating dissimilarity (0) or minimized, indicating similarity (1). In the online variant, the loss is only calculated for strictly positive or negative pairs, reported to perform better (Tunstall et al., 2022). The margin parameter λ controls how far dissimilar pairs must be separated. To study the models' behavior, we select a range of margin values for each compatible loss function (Table 2).

Loss function	λ margin	λ default
Triplet	{0.01, 0.1, 1.0, 5.0 , 7.5, 10}	5.0
Multiple Neg.	—	—
Contrastive	{0.1, 0.25, 0.5 , 0.75, 1.0}	0.5
Online Con.	{0.1, 0.25, 0.5 , 0.75, 1.0}	0.5

Table 2: Loss functions with margin selections. Default values are highlighted.

3.4 Example generation

As the classification datasets are not labeled for similarity, we use a reference model to generate contrastive samples of varying formats, corresponding to each input type: (1) Triplet, (2) Contrastive, and (3) MultipleNegatives, referred to as *example generation*. For each (sentence, label) pair in the data, the k nearest neighbors of each polarity are computed, requiring a minimum cosine similarity

threshold of ≥ 0.5 . These examples are then combined according to the selection of loss functions, e.g., with a TripletLoss requiring (anchor, similar, dissimilar). As the different data types will generate varying numbers of sentence pairs and triplets, the generation pipeline includes a dropout to normalize data samples. Table 3 shows an example of generated data.

Loss type	Contrastive Example
Triplet	Anchor: <i>Totally unexpected directions</i>
	Similar+Same polarity: <i>Dramatically moving</i>
	Similar+Opposite polarity: <i>Utterly misplaced</i>
Multiple Negatives	Anchor: <i>Good vibes</i>
	Similar+Same polarity: <i>Awesome energy</i>
Contrastive	Anchor: <i>A movie that deserves recommendation</i>
	Similar: <i>Effort to watch this movie</i>
	Label: 0 (increase distance \rightarrow less similar)
	Anchor: <i>Bad jokes, most at women’s expense</i>
	Similar: <i>Dumb gags, anatomical humor</i>
	Label: 1 (reduce distance \rightarrow more similar)

Table 3: Examples of contrastive and polarized samples for different loss types.

4 Experiments and Results

The results are based on fine-tuning and continuous evaluation of the baseline models in different setups for loss functions and corresponding parameters. Based on similar research on fine-tuning embeddings (Gao et al., 2022), models are trained for five epochs.

Suitable sample sizes The first experiment studies the impact of training samples, limited to the range [50, 100000]. Despite the reported effectiveness of few-shot learning for sentence-transformers (Tunstall et al., 2022), we observe improvements in polarity when increasing the sample size far beyond the scope of few-shot learning. Table 4 illustrates this behavior, aggregated across all models and loss configurations. Observe the increasing gap between the *min* and *max* scores for \mathcal{S} , while the mean is reduced. This is what we aim to reduce through joint fine-tuning.

Training details Based on findings from Table 4, the sample size is set to 50,000 to reduce compute time due to the limited improvements from 50,000 to 100,000. Experiments on the loss functions with their λ margins are then performed on both datasets. Models are trained for 5 epochs with a batch size of 64 and a learning rate of 3×10^{-5} , set to retrieve $k = 16$ sentences for evaluation.

N	\mathcal{P}			\mathcal{S}		
	Mean _{std}	Min	Max	Mean _{std}	Min	Max
50	75.7 _{7.5}	63.0	81.5	75.1 _{16.6}	46.6	85.5
500	75.7 _{7.5}	63.0	81.5	75.1 _{16.6}	46.6	85.5
2,000	75.7 _{7.5}	62.9	81.7	75.1 _{16.6}	46.6	85.5
5,000	76.3 _{7.7}	63.1	83.1	75.1 _{16.6}	46.5	85.5
10,000	78.0 _{8.3}	63.2	87.3	74.9 _{16.8}	45.7	85.4
20,000	81.5 _{8.7}	61.8	89.2	73.0 _{18.3}	36.4	84.9
50,000	86.2 _{6.4}	68.0	92.5	70.2 _{21.3}	29.6	84.7
100,000	88.9 _{4.0}	72.2	93.4	69.3 _{22.3}	29.0	84.6

Table 4: Aggregated scores across all configurations for different sample sizes after 5 epochs on the SST-2 dataset.

4.1 Results

Tables 5 and 6 show the polarity and semantic similarity scores obtained after the continued training with $N = 50,000$ samples. The “Reference” refers to each respective model before training. The tables showcase the impact of the different loss functions and their λ margins.

SetFit (Tunstall et al., 2022) is included, using the default Cosine Similarity loss. Figure 3 shows the best loss configuration for the strongest model *e5-small*. We observe an improvement in polarity at a minor cost of semantic similarity for several configurations. The TripletLoss, with smaller margins, shows consistently high performance for both metrics.

Additionally, we provide an evaluation using the established SentEval toolkit (Conneau and Kiela, 2018) on out-of-domain data. Table 7 shows the results with TripletLoss using a margin of $\lambda = 0.10$ and the results using SetFit (Tunstall et al., 2022), trained with 50,000 generated contrastive samples. Note how the fine-tuning approach yields higher scores, especially for the MR (Movie Reviews), CR (product reviews), and SST-2. The joint training also transfers well to tasks like SUBJ (subjective/objective classification), while somewhat lower scores are found on TREC (question-answering). The score increase aligns well with results in Tables 5 and 6, comparing SetFit to the highlighted TripletLoss $\lambda = 0.10$.

5 Discussion

Most model configurations adjusted the embeddings towards correct polarity upon fine-tuning. However, the *minilm-6* falls short of its semantic similarity capabilities, while the remaining models seem to learn both tasks, with only minor differences between the configurations.

Loss	λ	e5-small		gte-base		gte-small		minilm-6	
		sarcastic	sst2	sarcastic	sst2	sarcastic	sst2	sarcastic	sst2
Reference	-	71.4 _{21.2}	81.5 _{23.7}	67.4 _{20.7}	80.4 _{22.6}	66.8 _{20.6}	77.8 _{22.2}	63.7 _{20.2}	63.0 _{21.9}
SetFit (Cosine)	-	85.2 _{25.4}	86.2 _{24.2}	82.1 _{26.8}	85.6 _{25.5}	82.8 _{25.4}	84.2 _{25.9}	79.5 _{27.0}	77.9 _{29.0}
Contrastive	0.10	88.8 _{24.3}	89.5 _{23.2}	86.9 _{25.6}	89.2 _{24.1}	81.9 _{27.0}	88.0 _{25.6}	75.9 _{25.6}	68.0 _{24.9}
Contrastive	0.25	89.3 _{25.1}	90.7 _{23.2}	88.2 _{26.1}	90.0 _{25.0}	84.3 _{26.9}	88.8 _{26.1}	76.8 _{26.4}	72.4 _{26.9}
Contrastive	0.50	89.8 _{25.6}	91.2 _{23.8}	88.8 _{26.5}	90.3 _{25.3}	86.8 _{27.5}	89.1 _{27.1}	77.8 _{27.2}	75.1 _{27.8}
Contrastive	0.75	89.9 _{25.1}	91.6 _{23.6}	88.9 _{26.6}	90.6 _{25.1}	87.7 _{27.3}	89.5 _{26.9}	79.0 _{27.7}	77.3 _{28.6}
Contrastive	1.00	89.8 _{25.5}	91.2 _{24.3}	88.7 _{26.7}	90.7 _{25.1}	87.8 _{27.0}	89.6 _{26.8}	80.3 _{28.1}	78.4 _{28.9}
MultipleNeg	-	73.6 _{22.2}	80.8 _{22.4}	73.1 _{22.4}	81.8 _{23.5}	72.0 _{22.6}	80.6 _{23.4}	69.0 _{22.0}	69.4 _{23.1}
OnlineContr	0.10	89.6 _{24.7}	90.4 _{23.7}	87.4 _{25.8}	89.5 _{24.2}	82.6 _{27.0}	88.2 _{25.8}	78.9 _{26.0}	70.8 _{26.5}
OnlineContr	0.25	90.0 _{25.2}	91.5 _{23.8}	88.2 _{26.4}	90.2 _{25.4}	84.4 _{27.3}	88.9 _{26.7}	78.9 _{26.4}	74.6 _{27.8}
OnlineContr	0.50	89.7 _{25.9}	91.6 _{24.4}	88.2 _{27.3}	90.6 _{26.0}	86.0 _{27.6}	89.3 _{27.2}	79.0 _{26.9}	76.5 _{27.9}
OnlineContr	0.75	89.5 _{26.5}	91.7 _{24.5}	88.6 _{27.4}	90.8 _{25.6}	87.2 _{27.9}	89.2 _{27.6}	80.0 _{27.4}	77.5 _{28.2}
OnlineContr	1.00	89.6 _{26.6}	91.7 _{25.0}	88.3 _{27.3}	90.7 _{26.0}	87.5 _{27.7}	89.6 _{27.5}	80.5 _{27.8}	78.4 _{28.7}
Triplet	0.01	90.2 _{25.6}	91.5 _{25.1}	82.5 _{25.7}	90.3 _{24.9}	84.0 _{25.5}	89.1 _{26.2}	78.5 _{24.5}	76.9 _{26.9}
Triplet	0.10	90.6 _{26.3}	91.9 _{25.0}	89.7 _{27.1}	91.2 _{25.6}	88.4 _{27.2}	89.9 _{27.0}	83.5 _{26.9}	80.6 _{28.6}
Triplet	1.00	90.1 _{25.7}	90.9 _{23.5}	88.4 _{26.6}	90.6 _{24.9}	87.4 _{27.0}	88.6 _{25.7}	84.1 _{28.6}	83.2 _{31.1}
Triplet	5.00	88.2 _{25.1}	89.3 _{23.4}	86.5 _{26.8}	90.1 _{25.1}	84.9 _{26.5}	88.2 _{26.1}	81.5 _{27.7}	81.3 _{30.1}
Triplet	7.50	88.2 _{25.4}	89.6 _{23.1}	86.6 _{27.0}	90.1 _{25.0}	84.8 _{26.4}	88.2 _{25.9}	81.4 _{27.8}	81.5 _{30.1}
Triplet	10.00	88.1 _{25.1}	89.6 _{22.9}	86.8 _{26.6}	90.2 _{24.9}	84.8 _{26.8}	88.1 _{26.2}	81.6 _{27.8}	81.2 _{30.4}

Table 5: Polarity scores for all loss configurations after 5 epochs with $N = 50,000$ samples, retrieving $k = 16$ sentences. MultipleNegatives remain close to the reference model, while larger impacts are seen from Triplet- and Contrastive losses. The highest scoring data/model pairs are boldfaced. The most suitable loss configuration, Triplet $\lambda = 0.10$ is marked in green. Reference models are marked blue.

Loss	λ	e5-small		gte-base		gte-small		minilm-6	
		sarcastic	sst2	sarcastic	sst2	sarcastic	sst2	sarcastic	sst2
Reference	-	83.4 _{1.5}	85.5 _{1.7}	81.4 _{1.6}	83.7 _{1.4}	82.5 _{1.6}	84.8 _{1.4}	42.3 _{5.6}	46.6 _{7.4}
SetFit (Cosine)	-	78.5 _{2.1}	81.6 _{2.1}	75.6 _{3.0}	79.9 _{1.8}	75.6 _{2.5}	80.7 _{1.8}	17.8 _{5.6}	27.1 _{6.9}
Contrastive	0.10	79.4 _{2.0}	83.3 _{2.0}	75.0 _{2.4}	81.0 _{1.8}	78.7 _{2.1}	82.1 _{1.8}	25.6 _{6.6}	34.8 _{7.2}
Contrastive	0.25	79.7 _{2.0}	83.7 _{1.9}	76.2 _{2.4}	81.4 _{1.8}	79.0 _{2.1}	82.6 _{1.7}	26.6 _{6.6}	34.5 _{6.8}
Contrastive	0.50	79.7 _{2.0}	83.8 _{1.9}	76.9 _{2.4}	81.6 _{1.7}	79.1 _{2.1}	82.8 _{1.6}	27.1 _{6.6}	34.2 _{6.7}
Contrastive	0.75	79.8 _{2.0}	83.8 _{1.9}	76.5 _{2.6}	81.5 _{1.7}	78.7 _{2.3}	82.7 _{1.6}	27.1 _{6.5}	34.1 _{6.6}
Contrastive	1.00	79.8 _{2.0}	83.7 _{1.9}	76.5 _{2.7}	81.3 _{1.7}	78.1 _{2.5}	82.4 _{1.6}	27.8 _{6.5}	33.9 _{6.6}
MultipleNeg	-	82.5 _{1.6}	84.7 _{1.8}	80.4 _{1.8}	82.5 _{1.6}	81.6 _{1.8}	83.9 _{1.6}	39.9 _{6.1}	43.5 _{7.8}
OnlineContr	0.10	80.1 _{1.9}	83.8 _{1.9}	75.6 _{2.3}	81.2 _{1.8}	79.2 _{2.0}	82.5 _{1.7}	25.4 _{6.7}	33.2 _{7.1}
OnlineContr	0.25	80.5 _{1.9}	84.1 _{1.9}	77.1 _{2.3}	81.7 _{1.8}	79.7 _{1.9}	82.9 _{1.7}	27.1 _{6.6}	33.0 _{6.9}
OnlineContr	0.50	80.6 _{1.9}	84.1 _{1.9}	77.8 _{2.3}	82.0 _{1.6}	79.9 _{2.0}	83.0 _{1.6}	28.3 _{6.5}	33.9 _{7.0}
OnlineContr	0.75	80.6 _{1.9}	84.0 _{1.9}	77.5 _{2.5}	81.9 _{1.6}	79.4 _{2.2}	82.9 _{1.6}	28.5 _{6.5}	34.5 _{7.0}
OnlineContr	1.00	80.6 _{1.9}	84.0 _{1.9}	77.4 _{2.6}	81.7 _{1.6}	78.9 _{2.3}	82.7 _{1.6}	29.2 _{6.4}	35.0 _{7.0}
Triplet	0.01	81.2 _{1.8}	83.8 _{2.0}	78.0 _{2.4}	81.9 _{1.7}	79.9 _{2.0}	83.0 _{1.7}	25.8 _{6.2}	33.9 _{7.3}
Triplet	0.10	81.3 _{1.7}	83.7 _{1.9}	78.1 _{2.3}	81.9 _{1.7}	79.9 _{2.1}	83.0 _{1.6}	30.5 _{6.1}	35.2 _{7.3}
Triplet	1.00	79.2 _{2.1}	82.8 _{2.1}	76.3 _{2.9}	80.3 _{1.8}	77.2 _{2.6}	81.3 _{1.6}	23.7 _{6.0}	30.5 _{7.0}
Triplet	5.00	78.3 _{2.1}	81.8 _{2.1}	74.6 _{2.7}	79.9 _{1.8}	75.8 _{2.7}	80.6 _{1.7}	20.6 _{5.9}	29.6 _{7.0}
Triplet	7.50	78.4 _{2.1}	81.8 _{2.1}	74.7 _{2.7}	79.9 _{1.8}	75.7 _{2.7}	80.6 _{1.7}	20.4 _{5.9}	29.5 _{7.0}
Triplet	10.00	78.3 _{2.1}	81.8 _{2.1}	74.7 _{2.7}	80.0 _{1.8}	75.7 _{2.7}	80.7 _{1.7}	20.5 _{5.9}	29.6 _{7.0}

Table 6: Semantic similarity scores for all loss configurations after 5 epochs with $N = 50,000$ samples, retrieving $k = 16$ sentences. MultipleNegative ranking loss, although seemingly performing strongly on the task, does so due to minimal adaptation to the new training samples and is on par with the reference model. This can be confirmed by inspecting the results for \mathcal{P} in Table 5. As such, the two highest scores for each data/model pair are boldfaced. The most suitable loss configuration, Triplet $\lambda = 0.10$ is marked in green. Reference models are marked blue.

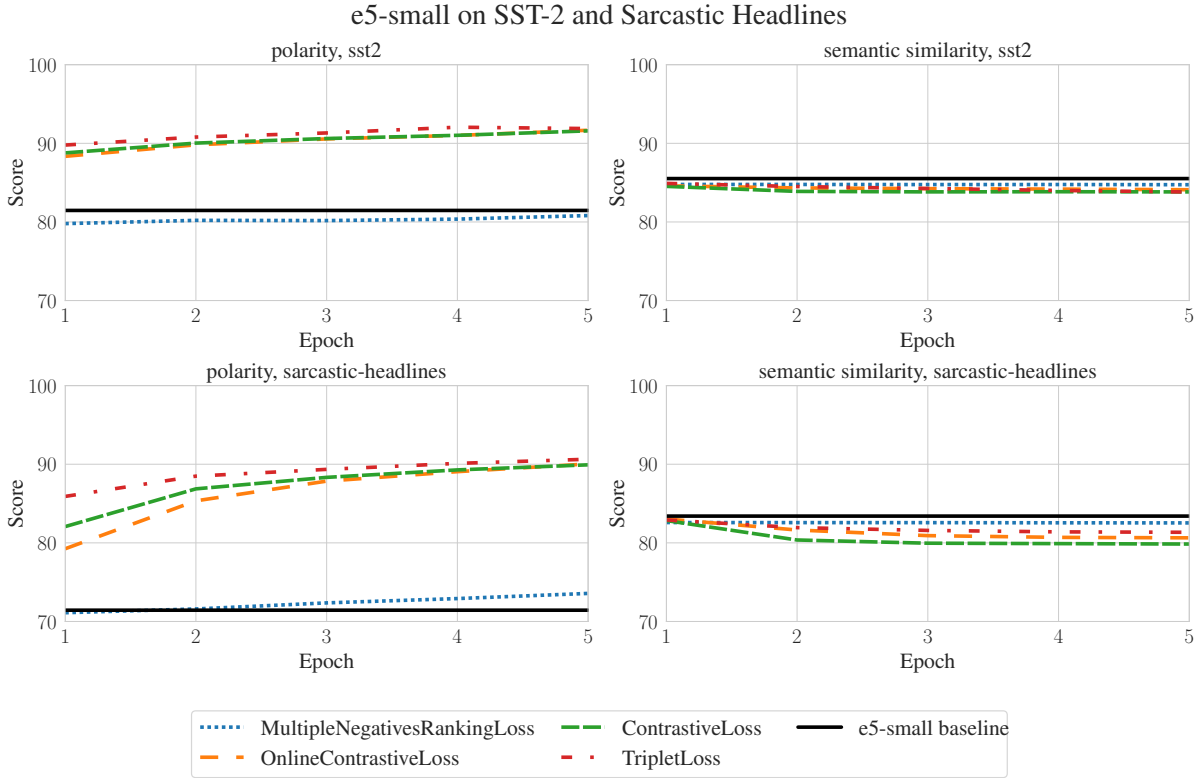


Figure 3: Best configurations per loss for the E5 Small model. Left: polarity, right: semantic similarity. TripletLoss outperforms the other alternatives. MultipleNegativesRankingLoss is insufficient due to its inability to be adjusted towards polarity.

Type	Model	Data	MR	CR	SUBJ	MPQA	SST2	TREC	MRPC	avg
Triplet λ 0.10	gte-base	sst2	89.31	89.27	92.91	85.95	93.19	80.80	73.33	85.50
Triplet λ 0.10	gte-base	sarcastic	84.33	88.82	92.82	88.04	90.83	88.40	68.52	85.01
Triplet λ 0.10	e5-small	sst2	88.95	88.98	91.06	86.28	93.41	79.80	74.55	84.97
Triplet λ 0.10	gte-small	sst2	87.72	89.59	90.85	86.86	91.38	79.00	73.39	84.83
SetFit	gte-base	sst2	84.30	88.85	90.91	86.08	89.18	86.00	72.52	84.27
SetFit	e5-small	sst2	85.43	85.16	86.58	83.93	91.05	88.00	69.39	82.18
SetFit	gte-base	sarcastic	81.61	86.52	90.01	87.50	88.69	86.00	66.55	81.92
Triplet λ 0.10	gte-small	sarcastic	80.51	83.52	90.17	86.11	87.59	84.60	66.49	81.84
SetFit	e5-small	sarcastic	82.69	83.97	90.65	86.80	88.80	90.20	66.49	81.62
Triplet λ 0.10	minilm-6	sst2	81.21	84.53	87.43	84.76	86.49	81.20	70.78	81.53
Triplet λ 0.10	e5-small	sarcastic	82.40	76.27	90.47	85.75	89.95	71.40	66.49	78.81
Triplet λ 0.10	minilm-6	sarcastic	71.20	66.44	86.57	79.63	80.94	74.40	66.49	74.61

Table 7: Performance on the SentEval benchmark, comparing TripletLoss with a margin of $\lambda = 0.10$ to SetFit with the same base models fine-tuned on sarcastic news headlines and sst-2. Sorted by average score. The highest scores for each metric are boldfaced.

Loss function analysis *TripletLoss* stands out as the best-performing loss function, especially when using smaller margin values ($\lambda \in \{0.01, 0.10\}$), strongly outperforming the default value of 5.0. For the *ContrastiveLoss* configurations, the default λ value of 0.5 seems well suited for the tasks, with minimal changes for different margins. *Multi-*

pleNegativesRankingLoss is an outlier in both results, perhaps due to poor example generation for this particular loss function. This loss treats sentences from distinct sentence pairs as dissimilar. As there are multiple generated pairs with the same anchor, this could result in contradictory examples. This problem does not arise for any of the other

loss functions.

Relations between distinct training examples (regarding polarity and semantic similarity) severely restrict example generation, and this process can be tweaked by studying the threshold for counting something as *similar* in more detail. The remaining loss functions have separate example generation implementations with control over the λ parameter that defines the margin between similar and dissimilar sentences. Interestingly, independent of the loss function, this value does not necessarily correlate with good model performance. For distinguishing polarity, higher λ values result in only slightly improved scores for ContrastiveLoss. For TripletLoss, the opposite is true, contradicting the intuition that the margin between two embeddings in vector space should be separated *more* rather than less.

Issues on Comparisons Comparing models of different loss functions is challenging due to the different data formats, as we cannot guarantee fair comparison when the inputs are unequal – e.g., comparing a triplet to a pair – for the different loss functions. Unlike typical research on loss functions, we did not consider the loss values obtained during training or evaluation, as we find these uninformative in this context, i.e., balancing two possibly opposing objectives. However, we argue that our suggested metrics in Section 3.1 are reasonable and intuitive and can likely be used for further studies on sentence embeddings.

6 Conclusion and Future Work

This paper has explored the potential of encoding polarity into sentence embeddings while retaining semantic similarity, done by fine-tuning models on data generated to suit the objectives of various sentence-transformers loss functions. We introduced two metrics to evaluate our results: the Polarity Score \mathcal{P} and Semantic Similarity Score \mathcal{S} . We found that the *e5-small* and *gte* models perform well on all evaluations. In Tables 5 and 6, the fine-tuned configurations greatly improve polarity scores while maintaining the semantic representation when evaluated on the generated datasets. For *e5-small*, performance on the *sarcastic* dataset shows great improvement in \mathcal{P} , increasing by 26.9% (from 71.4 to 90.6), while \mathcal{S} decreases by 2.5% (from 83.4 to 81.3). Similarly, on the *sst2* dataset, \mathcal{P} improves by 12.8% (from 81.5 to 91.9), and decreases by 2.1% in \mathcal{S} (from

85.5 to 83.7). Furthermore, the TripletLoss, especially for lower λ margins, e.g., $\lambda = 0.10$, strongly outperformed other configurations and has the potential to yield an efficient and high-performing model for multi-task retrieval, even outside of domains tested in this work, as the findings are mostly consistent between the evaluations.

Regarding future work, there are several paths for improvement:

- The suggested model configuration allows us to experiment with a broader range of tasks and datasets paired with our fine-tuning approach.
- The example generation process can be extended to support multiclass inputs by one-vs-rest and other methods to manage multiple classes with a system designed for contrasting two samples.
- Although our proposed metrics are a first step in assessing multiple objectives in this context, combining them better to represent the drift of the original semantic similarity remains an open question.

7 Limitations

The most prominent limitation is the number of domains implemented in the system, which is currently limited to sentiment analysis and sarcasm. Massive evaluations for multiple domains would make it difficult to present and analyze in detail. By reducing the number of loss configurations, more datasets can be evaluated and studied in detail, such as by limiting training to single margin values per loss function. The presented configuration requires 544 models to be trained *per dataset*. Another limitation is the definition and approximation of semantic similarity through the defined training pipelines. As described in the example generation procedure (Section 3.4), data points are separated on similarity by a frozen reference model. We still, however, see improvements in general semantic capabilities in the comparisons with current models, but an effort for labeling the already existing classification datasets for semantic similarity would be required for more reliable results.

8 Ethical Considerations

The datasets and pre-trained sentence-transformer models used are publicly available. However, the

system’s use for automatic retrieval may raise ethical concerns, particularly in public-facing applications. Furthermore, the Sarcastic News Headlines dataset references names of individuals and companies, requiring careful handling of personally identifiable data to prevent unintended harm.

CO₂ Emissions Experiments were conducted using a private infrastructure in Norway, which has a carbon efficiency of 0.024 kgCO₂eq/kWh according to <https://app.electricitymaps.com/>. A cumulative of 140 hours of computation was performed with an RTX 4090 averaging 270W. Total emissions are estimated to be 0.907 kgCO₂eq. Estimations were conducted using the MachineLearning Impact calculator presented in (Lacoste et al., 2019).

9 Reproducibility

All code is available on GitHub.⁴ Results and corresponding tables and figures are programmatically generated for efficient replication. Sampling operations are fully deterministic.

References

- Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. *SEM 2013 shared task: Semantic textual similarity. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 32–43, Atlanta, Georgia, USA. Association for Computational Linguistics.
- Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.
- Alexis Conneau and Douwe Kiela. 2018. Senteval: An evaluation toolkit for universal sentence representations. *arXiv preprint arXiv:1803.05449*.
- Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. 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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2022. Simcse: Simple contrastive learning of sentence embeddings.
- Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR’06)*, volume 2, pages 1735–1742. IEEE.
- Matthew Henderson, Rami Al-Rfou, Brian Strope, Yunhsuan Sung, Laszlo Lukacs, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and Ray Kurzweil. 2017. Efficient natural language response suggestion for smart reply.
- 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, Melbourne, Australia. Association for Computational Linguistics.
- Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.
- Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. 2019. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*.
- Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1188–1196, Beijing, China. PMLR.
- Qian Li, Hao Peng, Jianxin Li, Congying Xia, Renyu Yang, Lichao Sun, Philip S. Yu, and Lifang He. 2022. A survey on text classification: From traditional to deep learning. *ACM Trans. Intell. Syst. Technol.*, 13(2).
- Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning.

⁴<https://github.com/tollegj/margins-contrastive>

- Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 216–223, Reykjavik, Iceland. European Language Resources Association (ELRA).
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.
- Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narges Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2021. Deep learning–based text classification: a comprehensive review. *ACM computing surveys (CSUR)*, 54(3):1–40.
- Rishabh Misra and Prahal Arora. 2023. Sarcasm detection using news headlines dataset. *AI Open*, 4:13–18.
- Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2022. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*.
- Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.
- 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.
- 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.
- Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks.
- Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. FaceNet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.
- Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.
- Gizem Soğancıoğlu, Hakime Öztürk, and Arzucan Özgür. 2017. Biosses: a semantic sentence similarity estimation system for the biomedical domain. *Bioinformatics*, 33(14):i49–i58.
- Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1556–1566, Beijing, China. Association for Computational Linguistics.
- Lewis Tunstall, Nils Reimers, Unso Eun Seo Jo, Luke Bates, Daniel Korat, Moshe Wasserblat, and Oren Pereg. 2022. Efficient few-shot learning without prompts.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In the Proceedings of ICLR.
- Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2024. Text embeddings by weakly-supervised contrastive pre-training.
- Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2020. Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online. Association for Computational Linguistics.
- Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. *arXiv preprint arXiv:1909.00161*.

Database of Latvian Morphemes and Derivational Models: ideas and expected results

Andra Kalnača
University of Latvia
Visvalža 4a, Rīga, LV-1050
Latvia
andra.kalnaca
@lu.lv

Tatjana Pakalne
University of Latvia
Visvalža 4a, Rīga, LV-1050
Latvia
tatjana.pakalne
@lu.lv

Kristīne Levāne-Petrova
University of Latvia
Visvalža 4a, Rīga, LV-1050
Latvia
kristine.levane-
petrova@lu.lv

Abstract

In this paper, we describe “The Database of Latvian Morphemes and Derivational Models” – a large-scale manually validated database of Latvian derivational morphology currently in development at the Department of Latvian and Baltic Studies, Faculty of Humanities, University of Latvia (project funded by Latvian Council of Science, No. lzp-2022/1-0013). The database is based on lemmas extracted from the Balanced Corpus of Modern Latvian (LVK2018) and consists of two basic interlinked parts: an annotated list of morphemes and an annotated list of lemmas containing those morphemes. Morpheme-level data include morphemes with morpheme variants (allomorphs) and manually resolved morpheme homonymy/ homography, as well as information on morpheme types and hierarchical (diachronic) relations between root morphemes. Lemma-level data for each lemma include a unique lemma ID (coinciding with the original string extracted from the corpus), a manually validated base form, as well as information on morphemic segmentation, POS, grammatical features, derivational motivation (incl. compounding) and word-family membership. The focus of the database is on providing linguistically accurate comprehensive data as a reliable basis for future work in different fields, incl. computational linguistics.

1 Introduction

Latvian (Baltic group, Indo-European language family) is a language with rich inflectional and derivational morphology. Latvian inflectional

morphology is extensively documented in linguistic literature, e.g., in academic grammars (Endzelīns, 1951; Kalnača and Lokmane, 2021; Nītiņa and Grigorjevs, 2013), and, by virtue of being paradigmatic (and, as far as NLP is concerned, also synchronic), relatively readily submits to formalization, at least at the conceptual, if not at the practical, level. Over the last three decades, a number of approaches have been developed for Latvian inflectional morphology processing, resulting in solutions for wordform analysis, generation, lemmatization, POS-tagging, etc., many of them using some version of a lexicon for greater precision; for a recent proposal and an overview of previous work, see Paikens et al. (2024). Data on Latvian inflection are also available in UniMorph, which contains 136998 Latvian inflected forms corresponding to 7548 paradigms¹ (Kirov et al., 2018).

The derivational structure of words is inherently less straightforward and involves several levels of complexity (see Section 4), which need to be taken into account when developing derivational morphology processing technologies. Early computational linguistic experiments on Latvian derivational morphology have included attempts at describing possible approaches to automated morphemic segmentation of derived Latvian words and morphemic and morphological analysis, e.g., (Sarkans, 1996), but, to the best of our knowledge, no comprehensive working computational linguistic models of Latvian derivational morphology have been developed so far. It should be pointed out that up to now there has also been a lack of scientifically accurate large-scale resources (e.g., manually validated databases, lexicons) dedicated to Latvian derivational morphology that could serve as a basis for developing and testing computational linguistic, e.g., rule-based, models. The

¹<https://github.com/unimorph/lav>

most complete inventory of morphemically segmented Latvian words (base forms) to date, organized into word families based on a common root or, in some cases, on a non-segmentable stem, is Baiba Metuzāle-Kangere’s “Derivational Dictionary of Latvian” (a printed dictionary) (Metuzāle-Kangere, 1985).

Decisions about correct morphemic segmentation of complex words, derivational motivation or, e.g., allomorphy are not always straightforward for human linguists, and even less so for automated solutions unless the latter are trained or based on a large reliable body of data. In this paper, we describe a new digital resource (a database) dedicated to Latvian derivational morphology, currently in development and to be made freely available to the public in 2026. The “Database of Latvian Morphemes and Derivational Models” (DLMDM) is a corpus-based manually validated database in text format (.tsv files) with comprehensive data on the basic regularities of Latvian derivational morphology. DLMDM is designed as a general reference resource, its focus is on producing a large structured manually validated set of data accurate and consistent from the point of view of linguistic theory for the general public for all kinds of future uses, incl. as a source for NLP research.

2 Related work

Printed dictionaries of morphemes and derivational dictionaries have been around for quite some time. Particularly well represented are Slavic languages, e.g. Slovak (Sokolová et al., 1999), Czech (Slavičková, 1975; Šiška, 1998). There are also word-family dictionaries for other languages, e.g., German (Splett, 2009; Augst, 2009). Two notable dictionaries reflecting different aspects of Latvian morphemics and derivational morphology are “A Derivational Dictionary of Latvian” (Metuzāle-Kangere, 1985) and “Latīņu un grieķu cilmes vārdaļu vārdnīca” (A dictionary of Latin and Greek word parts) (Skujīņa, 1999). Metuzāle-Kangere’s dictionary is built around the concept of derivational families and is based on words extracted from two bilingual dictionaries.

The last 20 years have seen an increase in digital resources containing some sort of morphemic and/or derivational information. Such resources are often corpus-based in an effort to reflect actual

contemporary language use, but differ by focus, scope and methodology (e.g. autoconstructed vs., less frequently, manually annotated). Some of the recent examples include the Database of Lithuanian Morphemics Data (Rimkutė et al., 2013), MorphoLex, a lexical database for English words with morphological variables (Sánchez Gutiérrez et al., 2018), DeriNet (Vidra et al., 2019), a lexical network of word-formation relations in Czech, with autogenerated morphological segmentations of lemmas and identification of root morphs. Universal Derivations (UDer) is a collection of harmonized lexical networks of various languages capturing word formation, especially derivation, in a cross-linguistically consistent annotation scheme based on a rooted tree data structure as used in the DeriNet 2.0 database. MorphyNet is a large-scale, multilingual database that includes derivational and inflectional morphology data (over 13 million inflections and over 700 thousand derivations) for 15 languages extracted from Wiktionary and 90 thousand derivations in 271 languages inferred automatically from the combination of MorphyNet and the Universal Knowledge Core (Batsuren et al., 2021). UniMorph 2.0 contains some data on Latvian derivational morphology as supplementary structured data extracted from Wiktionary – 4235 complex words with a possible source word, a formally defined (POS:POS) word-formation model and means of derivation specified for each word². The quality of these data depends on the accuracy of Wiktionary and the level of detail is limited to what is available from that resource; e.g., derivation is not distinguished from compounding and formal means of derivation are not specified as morphemes of a certain type, but rather as word-initial or word-final strings of one or more morphemes merged together. Morphemic, incl. derivational, information is also included in a number of broader scope lexical resources, e.g., the lexical database of English WordNet encodes some derivational relations, the CELEX lexical databases of English, Dutch and German contain data on the derivational and compositional structure of words. Several approaches for induction of derivational families from words extracted from large corpora have been developed, e.g. DerivBase, DERivCELEX for German, DerivBase.Hr for Croatian, etc.

²<https://github.com/unimorph/lav/blob/master/lav.derivations>

3 Stages of development

DLMDM is based on a case-sensitive list of 165 090 lemmas downloaded in .xml format from The Balanced Corpus of Modern Latvian (LVK2018) via Nosketchengine (Rychly, 2007) with zero lower frequency threshold. LVK2018 contains approximately 10 million words occurring in texts of various genres (Levāne-Petrova and Dargis, 2018) and has been chosen as the primary initial source of lemmas for the database, because it provides a snapshot of real, unidealized contemporary language use and, apart from established words, also contains novel formations (hence, the zero lower frequency threshold). Adding lemmas from other sources, e.g., other corpora, dictionaries, etc., is possible by assigning a unique lemma ID in the LEMID column and providing a source ID in the SOURCE column.

Automated pre-processing:

- Data extraction.
- Consecutive automated and semi-automated removal of invalid lemmas – removing lemmas containing characters that are not part of the Latvian alphabet and then double-matching the remaining lemmas against *tēzaurs.lv* (2020 spring version³) and an open source spelling checking dictionary⁴, resulting in a list of unrecognized lemmas, which were then reviewed manually.
- Approximately 75 000 lemmas left as likely valid for further processing. The lemmas that have been filtered out include non-words, words in foreign languages, words containing spelling mistakes, erroneously generated lemmas, as well as a lot of proper names, some of which (rare or untypical for Latvian) have been left out from the final list;
- Morphological tagging⁵, using a freely available tagger for Latvian.
- Rule-based automated morphemic segmentation using custom developed scripts.

³<https://github.com/LUMII-AILab/Tezaurs.git/>

⁴<http://dict.dv.lv/download.php?prj=lv/>

⁵<https://github.com/PeterisP/LVTagger.git/>

- Grouping of lemmas into potential word families based on a shared root (or a non-segmentable stem) and a list of possible root allomorphs.

Further manual processing:

- Reviewing and correcting automatically generated lemma-level and word family data (see Section 5).
- Root homonymy/ homography resolution.
- Defining hierarchical relations between roots and non-segmentable stems.

The final stage of development will consist in defining and validating derivational relations between lemmas within word families.

In terms of workload, the most labour-intensive tasks have been morpheme homonymy/ homography resolution, as homographic morphemes have turned out to be pervasive in Latvian lexis, identifying synchronically non-evident allomorphs and also identifying hierarchical relations between roots and word-family membership of lemmas in non-straightforward cases.

4 Sources of complexity in data

As a manually validated database, DLMDM's primary focus is on providing comprehensive linguistically accurate data. This means accounting for all kinds of phenomena in derivational morphology, not just productive regular derivation. In this section, we outline some of the major sources of difficulty in derivational morphological analysis of existing words.

4.1 Morpheme homonymy and homography

Homonymy or homography is encountered much more often among Latvian roots and non-segmentable stems than among words. Derivational analysis without homonymy/ homography resolution may lead to incorrectly inferring derivational relations between words and, hence, to incorrect semantic interpretation (roots shown in round brackets):

(1) (*bur*)-*t* 'to do magic' (inherited Latvian word) – (*bur*)-*a* 'sail' (borrowing)

(2) (*las*)-*ī-t* 'to read' – (*las*)-*is* 'salmon' (both – inherited Latvian words)

(3) (*mat*)-*s* 'hair' (inherited Latvian

word) – (*mat*)-s ‘checkmate’ – (*fiz*)-(*mat*)-s ‘physico-mathematical (of students)’ (borrowing)

(4) (*log*)-s ‘window’ (inherited Latvian word) – (*virus*)-o-(*log*)-s ‘virologist’ – *ielogoties* ‘to log in’ (both – borrowings)

E.g., the string ‘lok’ or ‘loc’ corresponds to at least 7 different roots, in Latvian, occurring in hundreds of lemmas, as in (5)–(11):

(5) lok [luok], loc [luoc] – *loks* ‘circle’, *locīt* ‘to bend’, *lokāms* ‘bendable, declinable’ (inherited Latvian words)

(6) lok [lok], loc [loc] – *lokācija* ‘location’, *lokalizācija* ‘localization’, *lokātīvs* ‘locative’, *lokomotīve* ‘locomotive’, *translocēt* ‘translocate’ (all – borrowings)

(7) lok [luok], loc [luoc] – *ķiploks* ‘garlic’ (borrowing), *ķiplokains* ‘garlicky’, *ķiplociņš* ‘garlic diminutive’, *ķiploksāls* ‘garlic salt’

(8) lok [luok], loc [luoc] – *loki* ‘green onions’, *maurloki* ‘chives’, *sīpolloki* ‘spring onions’ (all – borrowings)

(9) lok [lok] – *loka* ‘hair curl’, *lokains* ‘curly’, *lokoties* ‘to curl’, *lokšķēres* ‘curling iron’ (all – borrowings)

(10) loc [luoc] – *locis* ‘ship pilot’ (borrowing)

(11) lok [lok] – *lokauts* ‘lockout’ (borrowing)

In DLMDM, homonymous/ homographic roots are listed as separate non-related morphemes each linked to their respective word family (or sub-family).

Another problem are quasi-morphemes – sequences of characters in borrowed words graphically coinciding with existing morphemes, most notably, suffixes. Quasi-morphemes may potentially lead to incorrect segmentation, e.g. in automated morphemic segmentation approaches:

(12) (*bārd*)-*ain*-is ‘a bearded man’ (inherited Latvian word) – (*sulain*)-is ‘butler’ (borrowed from Estonian *sulane*⁶)

(13) (*rūp*)-*est*-s ‘concern’ (inherited Latvian word) – (*dienest*)-s ‘service’ (borrowed from Middle Low German

*dēnest*⁷)

(14) (*vair*)-*og*-s ‘shield’ (inherited Latvian word) – (*karog*)-s ‘flag’ (borrowed from Old Russian⁸)

Other examples of quasi-morphemes include the nouns *ceriņi* ‘lilacs’, *treniņš* ‘training’, *zābaks* ‘a boot’, etc., where as a result of phonetic adaptation the segments *-iņ-* and *-ak-* have come to resemble the Latvian suffixes *-iņ-*, *-ak-*. Quasi-morphemes are less widespread than homonymous / homographic roots.

4.2 Allomorphy

The majority of Latvian roots have variants (root allomorphs) resulting from both historical and synchronic morphophonological processes (Kalnača, 2004; Kalnača and Lokmane, 2021; Nītiņa and Grigorjevs, 2013). Allomorphy is significant in inferring derivational relations between words. E.g., *ved*, *ves*, *ve*, *vez*, *vež*, *vad*, *vaz*, *važ* are all variants of the same root as in *vest* ‘to carry’, *vešana* ‘carrying’, *vedējs* ‘carrier’, *vadīt* ‘to lead’, etc.

Allomorphy also occurs in affixes, e.g., suffixes *-niek-*, *-niec-*, *-niec-*, as in (15):

(15) *saim-niek-s* ‘owner, host’ (M), *saim-niec-e* (F), *saim-niec-u* (GEN PL, F)

DLMDM encodes relations for all allomorphs occurring in the dataset, but not for all allomorphs that are, in principle, possible in Latvian.

4.3 Synchrony vs. diachrony

While most automated solutions for derivational morphology are synchronically oriented and focus on productive models, correct morphemic segmentation and word-family membership identification may sometimes require a diachronic stance, i.e. recognizing derivational models that are not synchronically productive, but are found in already established words, while retaining semantic motivation, e.g.:

(16) (*zag*)-*t* ‘to steal’ – (*zag*)-*l*-is ‘a thief’, (*bēg*)-*t* ‘to run away, to flee’ – (*bēg*)-*l*-is ‘a fugitive’, (*ie*)-*t* ‘to walk’ – (*ie*)-*l*-a ‘a street’

(17) (*sil*)-*t* ‘to warm’ – (*sil*)-*t*-s ‘warm’,

⁶<https://mev.tezaurs.lv/sulainis>

⁷<https://mev.tezaurs.lv/dienests/>

⁸<https://mev.tezaurs.lv/karogs/>

(*sal*)-*t* ‘to be cold, to freeze’ – (*sal*)-*t-s* ‘cold’

(18) (*bes*)-*t* (<**bed*-*t*) ‘to dig’ – (*bed*)-*r-e* ‘a pit, a hole’, (*svīs*)-*t* (<**svīd*-*t*) ‘to sweat’ – (*svied*)-*r-i* ‘sweat’

On the one hand, defining a synchronically unproductive word-formation model of this sort would probably lead to overgeneration (in generation tasks) and false positives (in analysis). On the other hand, not defining such models would lead to words like *zaglis*, *bēglis*, *ielā* being segmented and marked as simplex, which would also entail loss of derivational semantic motivation and word-family membership.

In DLMDM, established complex words not corresponding to synchronically productive word-formation models are segmented from a diachronic perspective.

4.4 Non-straightforward derivational relations and semantic motivation

Defining a single directed derivational relation and a single base (i.e. a single base word for derivation or a single syntactic construction for compounds) for each derivationally complex word is not always possible. Some words, in Latvian, may be simultaneously motivated by more than one base, and the perceived motivation may even vary from speaker to speaker, e.g., *burvīgs* ‘charming, enchanting’ and *burvība* ‘charm, sorcery, magic, enchantment’ are both related to *burvis* / *burve* ‘wizard, sorcerer (M) and (F)’ and to each other, esp. when taking word senses into account. Certain kinds of words, often these are compounds, rather than having a single base tend to form clusters around concepts (or some would perhaps say, fill in paradigms of possible meanings and parts-of-speech), while also forming links to one another, e.g., *aitas kopt* ‘to farm sheep’ – *aitkopis* / *aitkopa* ‘sheep farmer (M) and (F)’, *aitkopība* ‘sheep farming’; *gara aste* ‘a long tail’, *garaste*, *garastis* ‘someone having a long tail (F) and (M)’, *garastes* ‘long-tailed’ (a compound genitive noun), *garastains* ‘long-tailed’ (an adjective), *Garastene* (a proper noun in LVK2018); *lēkt ar izpletni* ‘to parachute’ – *izpletņlēcšana* ‘parachuting’, *izpletņlēcējs* ‘someone who parachutes’, etc. Another kind of examples are pairs of compound genitive nouns and adjectives related to one and the same concept, e.g., *starpnāciju*, *starpnacionāls* ‘international’; *pārreģionu*, *pārreģionāls*

‘transregional’, *bezgaršas*, *bezgaršīgs* ‘tasteless’, where a prior existence of an adjective that can fill the slot in the right-hand part of the compound seems to be a pre-requisite.

To summarize, a rooted tree does not seem to be able to accommodate all observable kinds of derivational relations in Latvian, therefore, word families in DLMDM are not designed to fit the rooted tree data structure.

4.5 Root hierarchies

Some roots or non-segmentable stems stand in a hierarchical relationship to one another. This is important for accurate morphemic segmentation and word-family membership:

- two or more inherited roots or an inherited and a borrowed root may be siblings with one common parent:

zero-element

dar

darb

zero-element

dilb

delm

deln

zero-element

as

aksi (borrowed)

akson (borrowed)

- one inherited root may be a child of another inherited root when there is no sufficient basis for further segmentation of the former:

aug, audz, audž

augst

augš

av

aun

ait

Thus, a root (or a non-segmentable stem) in DLMDM may have allomorphs and also a parent root and siblings or a child root, which, in turn, may have allomorphs of their own. Lemmas are linked to a concrete root in a root hierarchy.

5 Types of data in DLMDM

DLMDM consists of co-indexed text files for lemma-level data, morpheme-level data and

```
# ceriņ, cerīn, cerīņ
# stratum: BORROWED

ceriņš→ceriņš→(ceriņ)-š→NOUN→ncmsn1→LVK2018
ceriņš_dsk→ceriņi→(ceriņ)-i→NOUN→NounClass=PlTantum→ncmsn1→LVK2018
ceriņots→ceriņots→(ceriņ)-ot-s→ADJ→vmpcdmsnpsnpsn→LVK2018
ceriņa→Ceriņa→(Ceriņ)-a→PROPN→ncmsn1→LVK2018
ceriņš_prop→Ceriņš→(Ceriņ)-š→PROPN→ncmsn1→LVK2018
Ceriņi→Ceriņi→(Ceriņ)-i→PROPN→PropnClass=PlTantum→ncmpn1→LVK2018
ceriņs→ceriņi→(ceriņ)-i→NOUN→NounClass=PlTantum→ncmpn1→LVK2018
ceriņkrāsa→ceriņkrāsa→(ceriņ)-(krās)-a→NOUN→ncfsn4→LVK2018
ceriņkrūms→ceriņkrūms→(ceriņ)-(krūm)-s→NOUN→ncmsn1→LVK2018
ceriņlapa→ceriņlapa→(ceriņ)-(lap)-a→NOUN→ncfsn4→LVK2018
ceriņzars→ceriņzars→(ceriņ)-(zar)-s→NOUN→ncmsn1→LVK2018
ceriņzieds→ceriņzieds→(ceriņ)-(zied)-s→NOUN→ncmsn1→LVK2018
ceriņes→ceriņes→(ceriņ)-es→NOUN→NounClass=PlTantum→ncfsn5→LVK2018
```

Figure 1: The word family # ceriņ, cerīn, cerīņ ‘lilacs’ in a simplified format

source identifiers. To improve readability, manual revision is performed in a simplified format (see Figure 1). Upon completion, the files will be converted to a format compatible with CoNLL-U Plus to facilitate harmonization with other resources.

Each line in a DLMDM file contains data for one entry – a lemma, a morpheme or a source. Column values are tab-delimited.

The format of the database is largely inspired by DeriNet (Vidra et al., 2019) and Morpholex (Sánchez Gutiérrez et al., 2018), but, in terms of contents, DLMDM is different in many respects, the primary objective being to reflect the derivational morphology of Latvian as fully as possible. The major differences, apart from manual revision, include root hierarchies and morpheme-level data, as well as a different approach to marking derivational relations.

5.1 Lemma-level data

At the current stage, lemma data include the following columns:

Column	Description
LEMID	a unique case-sensitive lemma identifier coinciding with the original string extracted from the corpus
LEMMA	a manually validated base form of a lemma
SEGMENTATION	morphemic segmentation of a lemma
POS	part-of-speech tag in the UD format
FEATS	grammatical features
VARIANTS	lemma variants
MORPHTAG	an automatically generated morphological tag
SOURCE	a source identifier

Table 1: Lemma-level data

In addition, each lemma is linked to a concrete root or a non-segmentable stem in a root hierarchy through word-family membership.

Lemmas will be subsequently annotated for means of word-formation (e.g., syntactic: compounding, morphological: prefixation, suffixation), types of a derivational relationship (e.g., single base, multiple motivation) and participants of a derivational relationship.

Since DLMDM includes proper nouns, the LEMID, LEMMA and SEGMENTATION columns are case-sensitive. Two lemmas in the database can have identical values of the LEMMA and SEGMENTATION columns, but not of the LEMID column.

The parts-of-speech represented in DLMDM are shown in Table 2:

POS label	Description
NOUN	a noun
PROPN	a proper noun
ADJ	an adjective
ADV	an adverb
VERB	a verb, incl. participles
INTJ	an interjection
PRON	a pronoun
NUM	a numeral
ADP	an adposition
PART	a particle
CCONJ	a coordinating conjunction
SCONJ	a subordinating conjunction
OTHER	indeclinable words with a verbal motivation that do not fit any of the existing classes, e.g., <i>paslepu</i> ‘secret’, <i>piespiedu</i> ‘compulsory’

Table 2: POS column values in DLMDM

Developing a unified approach to what is to be considered a valid base form of a lemma (the LEMMA column) has also required some conscious decision-making, e.g., what to do in cases when the corpus contains both a masculine and a feminine version of a derivative, e.g. *nosūtītājs* (M), *nosūtītāja* ‘sender’ (F), but the automatically generated lemma list only has one of them, as inflectional endings partly overlap; or what to do in cases when the lemma list contains a participle, but not the corresponding verb, although both exist in language.

The manually validated base forms of lemmas in DLMDM are given as follows:

POS	Base forms
NOUN, PROP	nominative singular or nominative plural for pluralia tantum
ADJ	nominative singular masculine indefinite positive, unless an adjective is only used with the definite ending, e.g., <i>galvenais</i> ‘principal’
VERB	the infinitive for verb tense forms and nominative singular masculine for declinable participles, except for the past participle active, which is given in masculine and feminine

Table 3: The base forms of lemmas for major declinable parts-of-speech in DLMDM

The FEATS column encodes several specific grammatical features that either cannot be reliably automatically inferred from base forms or are required for other reasons, e.g., because participles do not have a dedicated POS tag (see Table 4).

FEATS	POS
PlTantum – pluralia tantum	NOUN, PROP, NUM
Gen – genitive nouns or numerals	NOUN, NUM
Indecl – indeclinable words	NOUN, ADJ, NUM
Part – participles	VERB

Table 4: Values of the FEATS column

The VARIANTS column is reserved for linking together different versions or variants, e.g., orthographic, dialectal, of the same word. The MORPHTAG column, which has been automatically generated for the purposes of automated pre-processing, incl. generating POS column values, will be removed in the final version of the database.

5.2 Morpheme-level data

DLMDM contains a separate file for morpheme data co-indexed with the lemma file. Morpheme-

level data will include concrete morphemes with allomorphs and homonymy/ homography resolution through unique IDs, as well as information on morpheme types, morpheme strata (e.g., for borrowed roots or non-segmentable stems), hierarchical relationships between roots or non-segmentable stems in a root hierarchy, and, for roots, links to lemmas through word-family membership.

6 Summary

We hope that DLMDM will be useful as a reliable large-scale resource for further research on Latvian derivational morphology from various perspectives, incl. computational linguistics, corpus linguistics and linguistics. Future work might include a more in-depth analysis of the structure of borrowed words in Latvian, esp. international words, words of classical (Greek, Latin) origin, incl. neoclassical compounds.

Abbreviations

GEN – genitive
F – feminine
M – masculine
PL – plural

References

- Gerhard Augst. 2009. *Wortfamilienwörterbuch der deutschen Gegenwartssprache*. Max Niemeyer Verlag, Berlin, New York.
- Khuyagbaatar Batsuren, Gábor Bella, and Fausto Giunchiglia. 2021. MorphyNet: a large multilingual database of derivational and inflectional morphology. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, Online. Association for Computational Linguistics.
- Jānis Endzelīns. 1951. *Latviešu valodas gramatika*. Latvijas Valsts izdevniecība, Rīga.
- Andra Kalnača. 2004. *Morfēmika un morfonoloģija*. Latvijas Universitātes Akadēmiskais apgāds, Rīga.
- Andra Kalnača and Ilze Lokmane. 2021. *Latvian Grammar*. University of Latvia Press, Riga.
- Christo Kirov, Ryan Cotterell, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqi, Arya D. McCarthy, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2018. Unimorph 2.0: Universal morphology. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*

- (LREC 2018). European Language Resources Association (ELRA).
- Kristīne Levāne-Petrova and Roberts Dargis. 2018. Balanced Corpus of Modern Latvian (LVK2018). CLARIN-LV digital library at IMCS.
- Baiba Metuzāle-Kangere. 1985. *A Derivational Dictionary of Latvian*. Helmut Buske Verlag, Hamburg.
- Daina Nītiņa and Juris Grigorjevs, editors. 2013. *Latviešu valodas gramatika*. Latvijas Universitātes Latviešu valodas institūts, Rīga.
- Peteris Paikens, Lauma Pretkalniņa, and Laura Rituma. 2024. A computational model of Latvian morphology. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 221–232, Torino, Italia. ELRA and ICCL.
- Erika Rimkutė, Asta Kazlauskienė, Gailius Raškinis, and Irena Markievicz. 2013. *Lietuvių kalbos morfemikos duomenų bazė*. Vytauto Didžiojo universitetas, Kaunas.
- Pavel Rychly. 2007. Manatee/bonito – a modular corpus manager. In *1st Workshop on Recent Advances in Slavonic Natural Language Processing*, pages 65–70, Brno. Masaryk University.
- Uģis Sarkans. 1996. Morphemic and morphological analysis of the latvian language. *Proceedings of the Fourth conference on Computational Lexicography and Text Research*, 28(1):219–225.
- Valentīna Skujiņa. 1999. *Latīņu un grieķu cilmes vārdaļu vārdnīca*. Kamene, Rīga.
- Eleonora Slavičková. 1975. *Retrograde morphemic dictionary of Czech language*. Academia, Prague.
- Miloslava Sokolová, Gustav Moško, František Šimon, and Vladimír Benko. 1999. *Morfematický slovník slovenčiny*. Náuka, Prešov.
- Jochen Splett. 2009. *Deutsches Wortfamilienwörterbuch: Analyse der Wortfamilienstrukturen der deutschen Gegenwartssprache, zugleich Grundlegung einer zukünftigen Strukturgeschichte des deutschen Wortschatzes*. De Gruyter, Berlin, New York.
- Claudia Sánchez Gutiérrez, Hugo Mailhot, Hélène Deacon, and Maximiliano Wilson. 2018. Morpholex: A derivational morphological database for 70,000 english words. *Behavior Research Methods*, <http://link.springer.com/article/10.3758/s13428-017-0981-8>:1–13.
- Jonáš Vidra, Zdeněk Žabokrtský, Magda Ševčíková, and Lukáš Kyjánek. 2019. DeriNet 2.0: Towards an all-in-one word-formation resource. In *Proceedings of the Second International Workshop on Resources and Tools for Derivational Morphology*, pages 81–89, Prague, Czechia. Charles University, Faculty of Mathematics and Physics, Institute of Formal and Applied Linguistics.
- Zbyněk Šiška. 1998. *Bázový morfematický slovník češtiny*. Palacký University, Olomouc.

Localizing AI: Evaluating Open-Weight Language Models for Languages of Baltic States

Jurgita Kapočiūtė-Dzikienė^{1,2}, Toms Bergmanis^{3,4}, Mārcis Pinnis^{3,4}

¹ Tilde IT, Lithuania

² Faculty of Informatics, Vytautas Magnus University, Lithuania

³ Tilde, Latvia

⁴ Faculty of Computing, University of Latvia

{name.surname}@tilde.com

Abstract

Although large language models (LLMs) have transformed our expectations of modern language technologies, concerns over data privacy often restrict the use of commercially available LLMs hosted outside of EU jurisdictions. This limits their application in governmental, defence, and other data-sensitive sectors. In this work, we evaluate the extent to which locally deployable open-weight LLMs support lesser-spoken languages such as Lithuanian, Latvian, and Estonian. We examine various size and precision variants of the top-performing multilingual open-weight models, Llama 3, Gemma 2, Phi, and NeMo, on machine translation, multiple-choice question answering, and free-form text generation. The results indicate that while certain models like Gemma 2 perform close to the top commercially available models, many LLMs struggle with these languages. Most surprisingly, however, we find that these models, while showing close to state-of-the-art translation performance, are still prone to lexical hallucinations with errors in at least 1 in 20 words for all open-weight multilingual LLMs.

1 Introduction

Since the fall of 2022, OpenAI and other big tech companies have transformed LLMs from an obscure technology little known outside the academic circles to a major household name. Key to this was the LLMs' ability to perform tasks specified in free-form instructions, making them excel as NLP tools¹. Furthermore, these models

can learn during inference from relevant examples provided as inputs, making them adaptable to new requirements or even tasks. Moslem et al. (2023) showed that in such a setting, GPT-3, provided with relevant translation examples, outperforms machine translation systems of major companies, including Google, DeepL, and ModernMT.

However, data privacy concerns often constrain the use of commercially available LLMs hosted outside EU jurisdiction, limiting their application in governmental, defence, and data-sensitive private sectors. Fine-tuning and operational deployment of adapted models can incur prohibitive costs in the case of commercially available LLMs, emphasizing the need for sovereign AI solutions—locally deployable alternatives that ensure security, control, and compliance. Recently, many powerful alternatives to the commercially available online LLMs have emerged (Jiang et al., 2023; Dubey et al., 2024; Team et al., 2024; Mesnard et al., 2024; Abdin et al., 2024). Although many of these LLMs officially support only a handful of languages with a large speaker base, their training data often incorporate texts from many other languages. Therefore, in practice, these languages receive some degree of support. However, the extent to which these languages are supported, to the best of our knowledge, still needs to be evaluated.

In this work, we aim to answer the question of to what extent, if at all, several popular open-weight models support Lithuanian, Latvian, and Estonian. All three languages have relatively small speaker bases and thus are unlikely to be focal points of major multilingual open-weight LLMs. We examine variants of Meta's Llama 3, Google's Gemma2, Mistral's NeMo, and Microsoft's Phi3 in their performance in multiple-choice question answering (MCQA) and machine translation (MT). We also manually assess the text quality generated by these models by identifying

¹<https://artificialanalysis.ai/leaderboards/models>.

the rate of incorrect words when answering open-ended questions.

We find that while some models like Gemma 2 nearly match the performance of top commercial models, many LLMs struggle with these languages. Surprisingly, even those models that approach state-of-the-art translation capabilities are still susceptible to lexical hallucinations.

2 Experimental Setting

We evaluate LLMs on multiple-choice question answering, machine translation, and text generation quality in open-ended question answering. While our experiments focus on the model performance for three languages—Lithuanian, Latvian, and Estonian—each with a speaker base under 3 million, we also include results for Czech and English for comparison purposes. We have chosen to evaluate models that consistently appear on various leaderboards. Specifically, we assess the **8.03B** and **70.6B** parameter versions of **Llama 3** and **Llama 3.1**, as well as the **3.21B** parameter version of **Llama 3.2** (Dubey et al., 2024) from Meta; the **9.24B** and **27.2B** parameter versions of **Gemma 2** (Team et al., 2024; Mesnard et al., 2024) by Google; the **3.8B** and **14B** versions of **Phi 3** by Microsoft (Abdin et al., 2024); and the **12.2B** parameter **NeMo** by Mistral AI (Jiang et al., 2023). To provide context for our experiments, we include online models by OpenAI such as **GPT-3.5 Turbo** and **GPT-4o** (OpenAI et al., 2024) and **DeepL** machine translation systems. In experiments assessing the quality of Lithuanian text generation, we incorporate the Lithuanian language-specific fine-tuned versions of Llama 2 (Touvron et al., 2023) with **7B** and **13B** parameters, developed by Neurotechnology – **Lt-Llama 2** (Nakvosas et al., 2024).

We run LLMs on our local hardware using the default inference parameters of the Ollama platform², which offers several levels of precision for model quantization: 4bit, 8bit, and full-precision – 16bit. By default, we use **4bit precision** in all our experiments, albeit at the cost of some performance degradation. We also evaluate the performance drop due to quantization by contrasting the results of quantized models with their full-precision counterparts.

²<https://ollama.com/>

Machine Translation For MT experiments, we use the FLORES-200 benchmark dataset (Goyal et al., 2021; Costa jussà et al., 2022), which comprises parallel sentences in over 200 languages³. We use the *devtest* subset from FLORES-200, which contains 1,012 sentences. We test LLMs in a zero-shot inference scenario. We use the following English prompt to request text translations from the specified source and to the specified target language:

*“{lang_a}: {sentence_{lang_a}}
Translate the above {lang_a} text into {lang_b}
{lang_b}: ”*

The translation and evaluation are performed at *sentence-level*; the inference is conducted in a single run for each test sentence. For automatic evaluation of MT quality, we use COMET⁴ (Rei et al., 2020, 2022) as it has been shown to have a higher correlation with human judgments than BLEU (Papineni et al., 2002) and to be more suitable for unrelated system comparison (Kocmi et al., 2024).

Multiple-Choice Question Answering For MCQA experiments, we employ the Belebele dataset, a benchmark in multiple-choice machine reading comprehension (Bandarkar et al., 2024). This dataset pairs each question with a short passage from the FLORES-200 dataset. Each question includes four multiple-choice answers, with one being the correct option. The dataset consists of 900 questions involving 488 distinct passages, each linked to one or two related questions. We use LLMs in a zero-shot inference scenario. We use the following English prompt where “{context}”, “{question}” and “{answer_#}” are in a specific language (Latvian, Estonian, etc.):

“This is the context: '{context}'. This is the question: '{question}'. Here are the 4 candidate answers: '1) {answer₁}; '2) {answer₂}; '3) {answer₃}; '4) {answer₄}'. Report only the correct answer's ID (1, 2, 3, 4) using the mandatory JSON format: {answer_id : "}. ”

The prompt explicitly requests the ID (e.g. ‘1’, ‘2’, ‘3’, or ‘4’) of the correct answer formatted in JSON. Our evaluation metric is accuracy.

³<https://github.com/facebookresearch/flores/tree/main/flores200>.

⁴<https://huggingface.co/Unbabel/wmt22-comet-da>

	DeepL	GPT		Llama: 3		3.1		3.2	NeMo	Gemma 2		Phi 3	
		3.5-T	4o	8B	70B	8B	70B	3B	12B	9B	27B	3B	14B
EN-LT	0.92	0.88	0.91	0.62	0.83	0.61	0.84	0.46	0.73	0.86	0.89	0.26	0.32
EN-LV	0.92	0.88	0.91	0.59	0.82	0.58	0.83	0.44	0.72	0.83	0.88	0.25	0.27
EN-ET	0.93	0.92	0.92	0.65	0.86	0.63	0.87	0.48	0.75	0.84	0.89	0.27	0.37
EN-CS	0.93	0.91	0.92	0.81	0.90	0.82	0.90	0.66	0.84	0.89	0.91	0.25	0.51
LT-EN	0.87	0.86	0.88	0.77	0.81	0.77	0.82	0.75	0.82	0.87	0.87	0.32	0.34
LV-EN	0.89	0.87	0.89	0.77	0.83	0.78	0.82	0.76	0.84	0.87	0.88	0.33	0.34
ET-EN	0.90	0.90	0.90	0.78	0.83	0.79	0.82	0.76	0.85	0.88	0.89	0.33	0.34
CS-EN	0.89	0.89	0.89	0.86	0.88	0.86	0.87	0.86	0.87	0.89	0.89	0.32	0.33
Avg.	0.91	0.89	0.90	0.73	0.85	0.73	0.85	0.65	0.80	0.87	0.89	0.29	0.35

Table 1: Automatic MT quality evaluation results in COMET scores across models and translation directions. DeepL and OpenAI GPT 3.5-Turbo and 4o are provided for reference. Top results by open-weight models for each translation direction are marked in **bold**.

	GPT		Llama: 3		3.1		3.2	NeMo	Gemma 2		Phi 3
	3.5-T	4o	8B	70B	8B	70B	3B	12B	9B	27B	14B
LT	0.734	0.941	0.607	0.768	0.618	0.834	0.435	0.715	0.861	0.898	0.001
LV	0.756	0.950	0.571	0.710	0.581	0.783	0.410	0.689	0.869	0.914	0.002
EN	0.903	0.962	0.883	0.938	0.872	0.947	0.740	0.898	0.931	0.943	0.886
ET	0.773	0.928	0.576	0.770	0.560	0.821	0.397	0.686	0.859	0.893	0.003
CS	0.818	0.937	0.769	0.888	0.743	0.892	0.676	0.800	0.907	0.910	0.296
Avg.	0.797	0.944	0.681	0.815	0.675	0.855	0.532	0.758	0.885	0.912	0.238

Table 2: Automatic MCQA evaluation results measuring accuracies across models and languages. OpenAI GPT 3.5-Turbo and 4o are provided for reference. Top results by open-weight models for each language are marked in **bold**.

	Llama 3.1		3.2	Gemma 2	
	8B	70B	3B	9B	27B
Δ MT	0.074	0.009	0.015	0.006	0.001
Δ MCQA	0.100	0.058	0.004	0.010	0.000

Table 3: Performance drop (difference between Avg. scores across all languages) for several 4bit models compared to their respective full precision versions on the two tasks – MT (COMET points) and MCQA (accuracy).

Text Quality in Free Form Question Answering

To assess LLMs’ ability to generate answers that adhere to Lithuanian and Latvian conventions and grammatical norms, we prompt models to answer ten free-form questions such as “*When did the Soviet Union collapse, how many new countries appeared, and what are their names?*” and “*Provide a definition of artificial intelligence.*” We conduct human evaluation by two Lithuanian and Latvian native speakers and linguistics experts. We require evaluators to count text errors, mark grammatically incorrect words or incorrect inflexions, mark invented words not existing in the language, and mark words within syntactically incorrect sentence structures (see Table 4).

We also assess whether the provided answers are factually correct. However, the factual accuracy results lack statistical significance due to the small sample size and should be interpreted with caution. For instance, it happened that GPT-4 answered all ten questions correctly for the Latvian language, but this outcome reflects a preliminary observation rather than a deep investigation. The results, therefore, should be viewed as part of a pilot study and not as definitive findings.

3 Results and Discussion

MT evaluation results (see Table 1) demonstrate the Gemma 2 family as the most capable open-weight model family. Gemma 2 27B emerges as the best locally deployable model, yielding COMET scores on par with OpenAI’s GPT-3.5 Turbo and only marginally worse than GPT-4o. Although specialised proprietary MT models like DeepL achieve the highest average score (0.91), freely available Gemma 2 models are not far off, with average COMET scores of 0.89 and 0.87 for the 27B and 9B versions, respectively. In this context, the Llama family has little to offer, with the Llama 3.0 and 3.1 70B param-

		GPT 4o		Llama 3.1 8B 70B	Gemma 2 27B	Lt-Llama 2 7B 13B
LT	Words/Sentences	320/38		724/68 300/24	1273/135	1132/69 1020/58
	Error Rate (%)	3.44		12.98 7.67	4.08	1.15 0.98
	Answer acc.	0.9		0.2 0.9	0.9	0.5 0.4
LV	Words/Sentences	1249/91		362/27 619/39	1171/97	- -
	Error Rate (%)	2.48		18.51 11.31	5.98	- -
	Answer acc.	1		0.3 0.9	0.8	- -

Table 4: Human evaluation results for text generation quality in free form question answering.

ter models surpassing the much smaller Gemma 2 9B only in two out of eight translation directions. Smaller models, like Llama’s 3B and 8B versions and Mistral’s 12B NeMo, show equally meagre results given the high performance of Gemma 2 9B. Lastly, the results of Phi 3 prove that these models have very little support for multilingualism.

Quantisation impact on MT quality analysis (see Table 3) reveals that while Llama models are negatively affected to some degree, the performance of full-precision models does not justify their use either. Increased inference time and memory requirements for the 70B model are too prohibitive unless several top-of-the-shelf enterprise-grade GPUs are available⁵. However, the full precision 8B parameter Llama 3.1 still does not reach the performance of the Gemma 2 9B 4bit version (0.80 vs 0.87). Gemma 2 family models, on the other hand, show a statistically insignificant drop in translation performance when the 4bit version is used, suggesting that their architecture is very robust to quantization.

While the current MT results provide valuable insights into LLM capabilities, future work could benefit from more fine-grained error analysis using frameworks like MQM (Multidimensional Quality Metrics) and ESA (Error Span Annotation). These approaches allow detailed classification of errors-such as those related to accuracy, fluency, and terminology, and help quantify their impact on text usability. Incorporating these methods could provide deeper insights into model limitations and guide targeted improvements, particularly for smaller languages like Lithuanian, Latvian, and Estonian.

MCQA results (see Table 2) show that the Gemma 2 27B parameter model outperforms GPT-3.5 Turbo across all languages, coming second only to OpenAI’s flagship model, GPT-4o. No-

tably, Gemma 2 27B achieves the highest accuracy among the open-weight models, outperforming Llama 3.1 and NeMo models for the Lithuanian, Latvian, Estonian, and Czech. The Phi models, however, perform poorly, particularly in non-English languages, and their results often fail to meet the required JSON output format, providing detailed responses instead of just answer IDs.

Quantization impact on MCQA analysis unveils a similar picture as the analysis above for MT: Llama models are more sensitive to quantization, while Gemma 2 are more robust. As a result, Gemma 2 models show little performance degradation when much more efficient 4bit models are used. It’s worth noting, however, that the accuracy drops because quantization differs depending on the amount of data each language has. Less spoken languages like Lithuanian, Latvian, and Estonian are affected more than English and Czech, for which overall results are better. For example, the Llama 3.1 70B model loses 0.06, 0.12, and 0.08 accuracy for Lithuanian, Latvian, and Estonian, respectively, but only 0.01 and 0.02 for English and Czech.

Text Generation Quality evaluation results (see Table 4) show that most models produce more than one error per 100 words. The Lt-Llama 2 models, specifically fine-tuned for Lithuanian, are the exception, with an error rate of just 0.98% and no invented words. Among multilingual models, OpenAI’s GPT-4o achieves strong performance, with 0.94 grammatically incorrect or incorrectly inflected words per 100 for Lithuanian and 1.52 for Latvian, while generating a very small number of invented words (0.31 and 0.56 per 100 for Lithuanian and Latvian, respectively). In contrast, Llama 3.1 models show significant shortcomings, with the highest frequency of grammatical errors: 8.01 per 100 for Lithuanian and 11.88 for Latvian. Additionally, Llama 3.1 generates a substantial number of invented words: 4.28 per 100 for

⁵<https://huggingface.co/blog/llama31>

Lithuanian and 3.87 for Latvian. Gemma 2 models perform considerably better, with 2.36 grammatical errors per 100 for Lithuanian and 4.10 for Latvian, and fewer invented words: 0.39 per 100 for Lithuanian and 1.45 for Latvian. These findings highlight clear quality differences among models. While Llama 3.1's high error rates make it unsuitable for most commercial applications, Gemma 2 strikes a better balance, approaching GPT-4o's quality but still falling short. Notably, Lt-Llama 2 sets the strongest benchmark with near-perfect output, minimal grammatical errors, and no invented words. On average, users can expect at least one linguistic error in every 2-3 sentences from the best open-weight models like Gemma 2, or every sentence for models like Llama 3.1, unless further multilingual specialization becomes available.

Lesser-spoken Languages like Lithuanian, Latvian, and Estonian have less support in open-weight models compared to more populous languages such as Czech. These differences are more pronounced in smaller and lower-quality models, especially in tasks where models generate text in lesser-spoken languages (e.g., MT from English into Lithuanian). Comparatively good results for Czech suggest that these disparities are related to the amount of data each LLM has seen for each language during training, rather than factors such as the structural complexity of the language.

4 Conclusions

Our findings demonstrate that certain open-weight LLMs, such as the Gemma 2 family, achieve performance comparable to top-tier commercial products, such as general-purpose models like OpenAI's GPT-4o and specialized machine translation services like DeepL. This progress enables local, secure, and private solutions, supporting the development of sovereign AI for many language tasks in governmental, defence, and other data-sensitive private sectors. Nevertheless, unless specifically fine-tuned for languages like Lithuanian, most multilingual models are still surprisingly prone to lexical hallucinations, highlighting the need for 1) high-quality language data for languages of the Baltic states and 2) research on language-specialized LLMs.

Acknowledgments

This research has been supported by the ICT Competence Centre (www.itkc.lv) within the project *2.4 Daudzvalodīgs uzņēmuma informācijas semantiskās meklēšanas un atbilžu gatavošanas risinājums* (2.4 Multilingual Semantic Search and Question-Answering Solution for Enterprises) of EU Structural funds, ID no 5.1.1.2.i.0/1/22/A/CFLA/008.

References

- Marah Abidin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Weizhu Chen, Yen-Chun Chen, Yi-Ling Chen, Hao Cheng, Parul Chopra, Xiyang Dai, Matthew Dixon, Ronen Eldan, Victor Fragoso, Jianfeng Gao, Mei Gao, Min Gao, Amit Garg, Allie Del Giorno, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Wenxiang Hu, Jamie Huynh, Dan Iter, Sam Ade Jacobs, Mojan Javaheripi, Xin Jin, Nikos Karampatziakis, Piero Kauffmann, Mahoud Khademi, Dongwoo Kim, Young Jin Kim, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Xihui Lin, Zeqi Lin, Ce Liu, Liyuan Liu, Mengchen Liu, Weishung Liu, Xiaodong Liu, Chong Luo, Piyush Madan, Ali Mahmoudzadeh, David Majercak, Matt Mazzola, Caio César Teodoro Mendes, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Liliang Ren, Gustavo de Rosa, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacrose, Shital Shah, Ning Shang, Hiteshi Sharma, Yelong Shen, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Praneetha Vaddamanu, Chunyu Wang, Guanhua Wang, Lijuan Wang, Shuohang Wang, Xin Wang, Yu Wang, Rachel Ward, Wen Wen, Philipp Witte, Haiping Wu, Xiaoxia Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Jilong Xue, Sonali Yadav, Fan Yang, Jianwei Yang, Yifan Yang, Ziyi Yang, Donghan Yu, Lu Yuan, Chenruidong 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.
- Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2024. The belebele benchmark: a parallel reading comprehension

dataset in 122 language variants. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Marta R. Costa jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Y. Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loïc Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation. *CoRR*, abs/2207.04672.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lomakin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnston, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Old-

ham, Mathieu Rita, Maya Pavlova, Melanie Kam-badur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Al-rassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Rapparthi, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collet, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Conguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yun-ing Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkan Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Sweeney, Gil Halpern, Govind

- Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damla, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabisa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Rutu Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vitor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihalescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojuan Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. 2024. The llama 3 herd of models.
- Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc’Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2021. The flores-101 evaluation benchmark for low-resource and multilingual machine translation.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Léo Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.
- Tom Kocmi, Vilém Zouhar, Christian Federmann, and Matt Post. 2024. Navigating the metrics maze: Reconciling score magnitudes and accuracies. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1999–2014, Bangkok, Thailand. Association for Computational Linguistics.
- Gemma Team Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, L. Sifre, Morgane Riviere, Mihir Kale, J Christopher Love, Pouya Dehghani Tafti, Léonard Hussenot, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Am’elie H’elieu, 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 Reid, Maciej Mikula, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Pier Giuseppe Sessa, 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 Klimentko, Tom Hennigan, Vladimir Feinberg, Wojciech Stokowiec, Yu hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Brian Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeffrey 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. *ArXiv*, abs/2403.08295.
- Yasmin Moslem, Rejwanul Haque, John D. Kelleher, and Andy Way. 2023. Adaptive machine translation with large language models. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 227–237, Tampere, Finland. European Association for Machine Translation.
- Artūras Nakvosas, Povilas Daniušis, and Vytas Mulevičius. 2024. Open llama2 model for the lithuanian language.
- OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, 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 Curry, 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 Kopic, 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.
- 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.
- Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Pro-*

cessing (EMNLP), pages 2685–2702, Online. Association for Computational Linguistics.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhatipatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshov, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iversen, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshtir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao, Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya,

Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. 2024. Gemma 2: Improving open language models at a practical size.

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.

How Aunt-Like Are You? Exploring Gender Bias in the Genderless Estonian Language: A Case Study

Elisabeth Kaukonen¹ Ahmed Sabir² Rajesh Sharma²

¹University of Tartu, Institute of Estonian and General Linguistics, Estonia

²University of Tartu, Institute of Computer Science, Estonia

Abstract

This paper examines gender bias in Estonian, a grammatically genderless Finno-Ugric language, which doesn't have gendered noun system nor any gendered pronouns, but expresses gender through vocabulary. In this work, we focus on the male-female compound words ending with *-tädi* 'aunt' and *-onu* 'uncle', aiming to pinpoint the occupations these words signify for women and men, and to examine whether they reveal occupational differentiation and gender stereotypes. The findings indicate that these compounds go beyond occupational titles and highlight prevalent gender bias.

1 Introduction

Languages are divided into three groups based on gender expression: firstly, there are grammatical gender languages (such as Russian, French, German, *etc.*), which use a gendered noun class system. Secondly, there are natural gender languages (*e.g.* English, Swedish, *etc.*), which incorporate gender-specific pronouns. Lastly, there are genderless languages (*e.g.* Hungarian, Finnish, Turkish, *etc.*), which lack gendered nouns as well as pronouns (Stahlberg et al., 2007). Estonian, representing a Balto-Finnic language, is grammatically genderless and thus incorporates only lexical resources, *i.e.* gender-specific vocabulary for gender expression.

While a grammatical gender does not correlate with gender equality or neutrality in a certain society (Aikhenvald, 2016), gender bias and stereotyping can still be prevalent not only in societies where a genderless language is spoken, but also within those languages themselves. This work illustrates how gender stereotypes are manifested in Estonian gendered vocabulary, specifically compound words

ending with lemmas *tädi* and *onu* that refer to occupations, shedding light on which professions are more commonly associated with women or men, and thus, how a genderless language exhibits bias and stereotypes. The gender stereotypes referred to here are mainly beliefs about occupational and social roles that are assumed to be held by men or by women more dominantly (Gygax et al., 2016; Vaidya, 2021).

In Estonian, the terms *tädi* and *onu* are primarily used to denote kinship, however, they also serve other purposes. For instance, they are commonly used in children's language, when referring to unfamiliar individuals or family friends when talking to children. Additionally, *tädi* and *onu* can be used humorously and they frequently appear in compound words denoting occupations (Puna, 2006). Such words were chosen for this paper, since they represent more informal and non-standardized language use. Furthermore, as these words represent informal language, they might reflect stereotypes more directly and with less linguistic filtering, as opposed to potentially more moderated words used in formal contexts. Examining gender bias in genderless languages, such as Estonian, is crucial because this topic has received little attention in the context of low-resource languages. Such languages still provide valuable insights into gender dynamics and social beliefs, which help to identify harmful and discriminatory gender stereotypes as well as raise awareness of gender inequality and occupational segregation. The research questions this study aims to address are as follows: (1) What kinds of occupations do the compound words ending with *tädi* and *onu* express? (2) How do occupational titles ending with *tädi* (aunt) and *onu* (uncle) propagate gender bias in Large Language Models (LLMs)?

2 Gender Expression in Estonian

Gender in Estonian is only expressed through vocabulary. This can be done, for instance, by using

Male-dominated		Female-dominated	
occupation	%	occupation	%
Doctor	84	Cashier, shopkeeper	80
Construction worker	1	Cook	72
Security worker	22	Librarian	98
Bus or tram driver	10	Kindergarten teacher	99
Electrician	1	High school teacher	86
EU politician	27	Receptionist	74
IT support specialist	28	Ticket seller	91
Waste collector	0	Social worker	92
Warehouse worker	8	Cleaner	88
Mailman	40	Hairdresser	94

Table 1: The percentage of females in male- and female-dominated occupations (%) in the Estonian labor force statistics, 2021.

separate words (*e.g.* mees ‘man’, naine ‘woman’, tüdruk ‘girl’, poiss ‘boy’, ema ‘mother’, isa ‘father’). In addition, another option to express gender is through compounding. This means adding two single words together, one of which carries a gendered meaning. There are two ways to indicate gender with a compound word in Estonian. Firstly, gendered prefixes *nais-* ‘female’ and *mees-* ‘male’ that function as adjectives can be added to a role noun (*e.g.* naisarst ‘female doctor’, naisujuja ‘female swimmer’, meesmodell). Secondly and similarly, gender-specific base forms (*i.e.* suffixes) can be used (*e.g.* esimees ‘chairman’, ärinaine ‘businesswoman’, spordinaine ‘sportswoman’). In the second option, the noun indicating gender conveys the main meaning of the word.

There is also a third option - derivation, which specifically denotes female agents, including female representatives of different ethnicities, for example, lauljanna ‘female singer’, venelanna ‘female Russian’, poetess ‘female poet’, sõbratar ‘female friend’, ‘girlfriend’ *etc.* (Kasik, 2015; Haselblatt, 2015). However, derivation, compared to single words and compound words, is perhaps not that widely used in everyday language today. Derivation is the only instance where a gendered morpheme is used in Estonian vocabulary.

As for compound words with a gendered base form, generally the most common nouns used in such compounds are *-mees* ‘man’ and *-naine* ‘woman’ (especially *-mees*, since *mees*-ending compounds are used generically, like *esimees* ‘chairman’). However, other nouns, such as *-tädi* ‘aunt’ and *-onu* ‘uncle’ can also be included. (Gu, 1990; Clyne et al., 2009; Kiss, 2022).

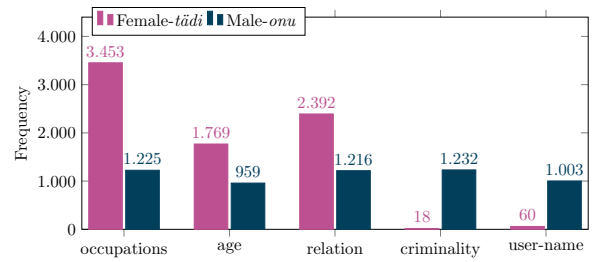


Figure 1: Frequencies of the more dominant semantic categories of gendered *tädi* and *onu* compound words that occurred in the Estonian web corpus.

3 Dataset

The dataset used in this study comes from the web subcorpus of the Estonian National Corpus of 2021, contains 724 million words (882 million tokens), with a variety of genres (*e.g.* online forums, e-commerce, online periodicals, property portals, recipe collections, *etc.*). To navigate the corpus, the SketchEngine tools (Kilgarriff et al., 2004) are used to extract compound words ending with lemmas *-tädi* and *-onu*. The extract token frequencies of compound words are 6500 (830 types) for the male compounds *-onu* and 6100 (700 types) for the female compounds *-tädi*. Compound words that occurred in the data were classified into semantic categories (see Figure 1), based on the meaning of the first part (or the prefix) of the compound. From these categories, words referring to occupations were specifically selected and chosen for analysis. The total number of occupational titles after pre-processing is 206 titles. We use the Estonian labor force statistics¹ to illustrate descriptive gender bias as shown in Table 1, the percentage distribution of females in male-dominated and female-dominated occupations.

4 Data Analysis and Result

4.1 Statistical Evaluation

As for words denoting occupations and activities, primary focus was on identifying the occupations associated with *tädi* and *onu* and whether titles denoting women and men correspond to different occupations, thereby revealing gender-based occupational stereotypes. To categorize occupational titles, words that appeared at least three times were considered. Table 2 shows an overview of the different types of occupations that emerged with *tädi*- and *onu*-compounds. The percentages show the

¹<https://palgad.stat.ee/>

Occupation	Female- <i>tädi</i> -compounds			Male- <i>onu</i> -compounds			Examples
	occ	%	type	occ	%	type	
Customer service	1502	44	57	87	7	12	<i>raamatukogutädi</i> (library aunt), <i>garderoobitädi</i> (wardrobe aunt)
Healthcare	464	13	10	150	12	1	<i>arstionu</i> (doctor uncle), <i>haigladädi</i> (hospital aunt)
social work	378	11	15	–	–	–	<i>koolitädi</i> (school aunt), <i>kasvatatädi</i> (kindergarten teacher aunt)
Construction	–	–	–	66	5	8	<i>remondionu</i> (repair uncle), <i>toruonu</i> (pipe uncle)
Entertainment	42	1	8	58	5	7	<i>kunstitädi</i> (art aunt), <i>kaameraonu</i> (camera uncle)
Law-enforc	134	4	7	247	20	10	<i>turvapädi</i> (security aunt), <i>valvurionu</i> (guard uncle)
Journalism	29	1	4	35	3	4	<i>raadioädi</i> (radio aunt), <i>leheonu</i> (newspaper uncle)
Business	–	–	–	14	1	4	<i>naftaonu</i> (petroleum uncle), <i>corp-onu</i> (corporate uncle)
Science	–	–	–	17	1	5	<i>teadlaseonu</i> (scientist uncle), <i>tehnikaonu</i> (technology uncle)
Politics	43	1	6	66	5	8	<i>riigonu</i> (government uncle), <i>europädi</i> (European parliament aunt)
Cleaning	130	4	3	7	1	1	<i>koristatädi</i> (cleaning aunt), <i>prügionu</i> (garbage uncle)
Animal	111	3	8	4	0.3	1	<i>koeratädi</i> (dog aunt), <i>farmitädi</i> (farm aunt)

Table 2: Groups of occupations emerged occupational titles ending with female compounds *tädi* (aunt) and male compounds *onu* (uncle) expressed, including occurrences, percentages from the whole group, type frequencies and example words. Type frequency denotes the number of different compounds in the corpus (*i.e.* how many different *tädi*-compounds emerged).

proportion of certain types of occupations among all occupational title ending with either *tädi* or *onu*.

Tädi in occupational titles primarily marked professions related to customer service (44% from all occupational titles), healthcare (13%), and social work (11%), while *onu* in occupational titles predominantly represented law enforcement (20%), followed by healthcare (12%) and customer service (7%). Thus, women are more often associated with occupations related to children, teaching, and (elder) care, while men are often found in the role of guards and police officers. As for *tädi*-compounds, there were no instances of words expressing occupations related to repairing and construction, business and entrepreneurship, and science and technology. Therefore, occupational titles ending with *tädi* and *onu* reflect the traditional gender associations regarding occupations, highlighting those typically attributed to women and men (Kaukonen, 2023).

4.2 LLMs Evaluation

In this section, we examine the propagation of occupational title biases in compound words ending with *tädi* (female 'aunt') and *onu* (male 'uncle') in LLMs. For this, inspired by the human-written CrowS-Pairs dataset (Nangia et al., 2020), which uses sentence pairs to highlight stereotypes across social categories, we manually created sentence pairs using the same Estonian National Corpus (see Section 3). These pairs are based on 87 occupations, with one occupation per pair of sentences (in total 174 sentences) where the occupational bias can be used with either gendered compound word (see Table 3) *e.g.* "The [cleaning aunt/uncle] carefully dusted the drawers.". The Estonian la-

bor force statistics database (2021) is also used as a reference to identify descriptive gender bias, reflecting gender-stereotyped professions.

We employ the most recent state-of-the-art LLMs models, ChatGPT (OpenAI, 2022), GPT-4 (Achiam, 2023), GPT-4-Turbo, GPT-4o (OpenAI, 2024a), GPT-o1 (OpenAI, 2024b), LLAMA-3 (Touvron et al., 2023) (8B and 70B models), and LLAMA-2-7B fine-tuned Estonian models LLAMAS (Kuulmets et al., 2024): (1) LLAMAS-base that is fine-tuned on 5B tokens (Both are fine-tuned on 75% Estonian, 25% English dataset), (2) LLAMAAS is an improved version of LLAMAS-base that is additionally instruction-tuned, and (3) LLAMAAS-MT is additionally finetuned on translation instructions on English-Estonian dataset.

For the prompt-based model, we set the temperature parameter to zero through all experiments to ensure consistent output and run the experiments three times. Majority voting is used to finalize the model's decision, except for the reasoning-based GPT-o1 model. For the LLAMA models (LLAMA-3 and LLAMAS), we extract the mean probability of all tokens in the sentences containing compound words as the occupational title gender bias score, as shown in Table 3 with LLAMA models.

Gender Bias Amplification Score. For evaluation, we measure the bias amplification as correlation measure (Zhao et al., 2017) towards the protected attribute $g \in \{uncle \text{ male or } aunt \text{ female}\}$ compounds words and the occupational title:

$$b(\text{occ}, g) = \frac{c(\text{occ}, g)}{\sum_{g' \in \{m, f\}} c(\text{occ}, g')}$$

where $c(\text{occ}, g)$ is the occurrences of the occupa-









Model	Sentence with Compound Words English Translation with Original Estonian Sentence	Bias Ratio	
		to-Uncle	to-Aunt
Eng	Baker aunt /Baker uncle made delicious pretzels		
LLAMMAS	pagari tädi d/pagari onu d valmistasid maitsvaid kringleid	0.48 	0.52 
LLAMA-3-70B	pagari tädi d / pagari onu d valmistasid maitsvaid kringleid	0.46 	0.54 
Eng	The cleaning aunt /cleaning uncle carefully dusted the drawers		
LLAMMAS	Koristaja onu /Koristaja tädi pühkis hoolega tolmu kummutilrecip	0.48 	0.51 
LLAMA-3-70B	Koristaja onu /Koristaja tädi pühkis hoolega tolmu kummutilrecip	0.50 	0.49 

Table 3: Examples of occupational title bias using the fine-tuned Estonian LLAMA (LLAMMAS) and the off-the-shelf LLAMA-3-70B models. (Top) The example demonstrates how the models measure gender bias, associating bakery tasks with women. (Bottom) In the example with the cleaning [aunt/uncle] occupational title, the standard LLAMA-70B incorrectly reflects the female-biased occupation.

Model	Occupational Title Ratio		
	M <i>onu</i>	F <i>tädi</i>	%
Labor Force Data	0.37	0.63	
ChatGPT (OpenAI, 2022)	0.43	0.57	0.64
GPT-4 (Achiam, 2023)	0.68	0.32	0.66
GPT-4-Turbo	0.63	0.37	0.71
GPT-4o (OpenAI, 2024a)	0.34	0.66	0.83
GPT-o1 (OpenAI, 2024b)	0.36	0.64	0.85
LLAMA-3-8B (Touvron et al., 2023)	0.46	0.54	0.52
LLAMA-3-70B	0.38	0.62	0.60
LLAMMAS (Kuulmets et al., 2024)	0.47	0.53	0.64
LLAMMAS-Base	0.48	0.52	0.63
LLAMMAS-MT	0.55	0.45	0.49

Table 4: Comparison result between different LLMs on occupational title using *tädi* and *onu* compound word. For the LLAMA-3 and Estonian LLAMMAS-7B, we rely on the mean probability, of the sentence with the bias occupations, for measuring the bias. The results indicate that the GPT-o1 model aligns closely with Estonian labor force statistics.

tions and the male-female compound words ending with *tädi* and *onu*. Table 4 shows a comparison results between different state-of-the-art LLMs. The best model aligned with labor force statistics is GPT-o1, especially concerning less common biased occupational titles (*e.g.* *piimatädi*, which refers to milk lady). The GPT-4o model achieved a comparable alignment level of 83%. The Estonian fine-tuned model LLAMA-2-7B (LLAMMAS) reflects the biases more accurately than the standard LLAMA-3 models with a 4-point difference in descriptive bias alignment compared to the 70B model.

Table 3 shows examples of the open-source model bias scores for the fine-tuned model LLAMMAS and the standard LLAMA-3-70B. The bottom example shows that the off-the-shelf model

incorrectly reflects a female-biased occupational title from the labor data, *cleaning aunt/uncle*.

5 Discussion

The analysis of compound words suggests women typically assume caregiving roles and are often associated with children, while men occupy professions like law enforcement. Additionally, men are more common in fields such as construction, business, entrepreneurship, and science. Conversely, the data indicates that men are rarely seen working in the educational sector. While this could indicate coincidental occupational gender differences, the results appear to reflect sectorial segregation, with women overrepresented in low-paid sectors like care, education, and customer service. Evidence from the 2021 Estonian Census supports this, showing 86% of healthcare and social welfare, 83% of education, and 82.3% of the service sector workers are women.

The analysis of LLMs revealed that these models propagate occupational biases related to compound words. Specifically, the fine-tuned Estonian LLAMMAS model reflects biases from Estonian labor force statistics more accurately than the similar-sized LLAMA-3-8B and larger LLAMA-3-70B models. This indicates that the process of fine-tuning has amplified the inherent biases within the model. For instance, the secretary as a female-biased occupation aligned correctly with all models (except for LLAMA family). However, in the fine-tuned model that incorporates additional parallel data (English-Estonian sentence pairs), the labor force data alignment bias ratio is lower compared to all other models, particularly for highly female-biased occupations (*e.g.* nanny, hairdresser, *etc.*).

6 Conclusion

This paper examined Estonian compound words ending with *tädi* ('aunt') and *onu* ('uncle') in the Estonian web corpus 2021. The findings indicate these terms reflect traditional gender roles and stereotypes in occupational contexts, which are also mirrored by LLMs, reinforcing gender biases.

Limitation

The limitations of the present study include the analysis of only informal gendered language units such as compounds ending with *tädi* and *onu*. If, for example, *-mees* 'man' and *-naine* 'woman' compounds, some of which constitute official occupational titles, were examined, then a more broad view of entrenched stereotypes could be achieved. Furthermore, several of the examined occupational titles were low in frequency as well as expressing quite novel or uncommon professions. As for the analysis of usage, the study included only specific uses of *tädi* and *onu*, and such an analysis may not translate to all other cases.

Ethics Statement

In this work, we measure gender bias patterns using descriptive modeling, which reflects observed real-world statistics. However, we also recognize the importance of normative analysis, which provides critical insights into promoting fairness and achieving equitable and unbiased outcomes. Balancing these approaches contributes to building a more just and inclusive society.

The corpus used in this study have been obtained from publicly available sources and have been anonymized. Any conflicts of interest or biases that may influence the interpretation of results are acknowledged. The authors acknowledge that this approach to gender does not encompass the entirety of gender identities, many of which are not represented by this vocabulary. Furthermore, only one bias considering gender is addressed in this paper, while the dataset may contain other demographic biases, such as race, religion and nationality. Also, this study focuses on occupational titles ending with *tädi* (aunt) and *onu* (uncle), which may propagate specific gender biases tied to cultural stereotypes regarding roles traditionally associated with women or men.

Acknowledgment

This work has received funding from the EU H2020 program under the SoBigData++ project (grant agreement No. 871042), by the CHIST-ERA grant No. CHIST-ERA-19-XAI-010, (ETAg grant No. SLTAT21096), and partially funded by HAMISON project.

References

- Josh et al. Achiam. 2023. Gpt-4 technical report. *arXiv:2303.08774*.
- Alexandra Y Aikhenvald. 2016. *How gender shapes the world*. Oxford University Press.
- Michael G Clyne, Catrin Norrby, and Jane Warren. 2009. *Language and human relations: Styles of address in contemporary language*. Cambridge University Press.
- Yueguo Gu. 1990. Politeness phenomena in modern chinese. *Journal of pragmatics*.
- Pascal M. Gygax, Alan Garnham, and Sam Doehren. 2016. [What do true gender ratios and stereotype norms really tell us?](#) *Frontiers in Psychology*, 7.
- Cornelius Hasselblatt. 2015. [The representation of gender in estonian](#). *Gender Across Languages. The linguistic representation of women and men*, 4:125–151.
- Reet Kasik. 2015. *Sõnamoodustis*. Tartu Ülikooli Kirjastus.
- Elisabeth Kaukonen. 2023. Cleaning aunts and police uncles in action. unveiling gender dynamics in estonian compound words. *Journal of Estonian and Finno-Ugric Linguistics*, 14(3):137–171.
- Adam Kilgariff, P Rychly, P Smrz, D Tugwell, G Williams, and S Vessier. 2004. The sketch engine. pages 105–115.
- Orsolya Kiss. 2022. Forms of address in the tatar language spoken in finland and estonia. Master's thesis, University of Tartu, Tartu.
- Hele-Andra Kuulmets, Taido Purason, Agnes Luhtaru, and Mark Fishel. 2024. [Teaching llama a new language through cross-lingual knowledge transfer](#).
- Nikita Nangia, Clara Vania, Rasika Bhalerao, and Samuel R. Bowman. 2020. CrowS-Pairs: A Challenge Dataset for Measuring Social Biases in Masked Language Models. In *EMNLP*.
- OpenAI. 2022. [Chatgpt: Optimizing language models for dialogue](#).
- OpenAI. 2024a. Gpt-4o system card. <https://openai.com/index/gpt-4o-system-card/>. Accessed: 2024-08-08.

- OpenAI. 2024b. [Openai o1 system card](#).
- Kerli Puna. 2006. *Soospetsiifilised isikunimetused sõnaraamatutes ja tekstides*. Ph.D. thesis, Tartu Ülikool, Tartu.
- Dagmar Stahlberg, Friederike Braun, Lisa Irmen, and Sabine Sczesny. 2007. Representation of the sexes in language.
- 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*.
- Shubhangi Vaidya. 2021. *Gender Stereotypes*, page 654–663.
- Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2017. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. *arXiv preprint arXiv:1707.09457*.

Estonian isolated-word text-to-speech synthesiser

Indrek Kiissel

ikiissel@gmail.com

Liisi Piits

Liisi.Piits@eki.ee

Heete Sahkai

Heete.Sahkai@eki.ee

Indrek Hein

Indrek.Hein@eki.ee

Liis Ermus

Liis.Ermus@eki.ee

Meelis Mihkla

Meelis.Mihkla@eki.ee

Institute of the Estonian Language, Tallinn, Estonia

Abstract

This paper presents the development and evaluation of an Estonian isolated-word text-to-speech (TTS) synthesiser. Unlike conventional TTS systems that convert continuous text into speech, this system focuses on the synthesis of isolated words, which is crucial for applications such as pronunciation training, speech therapy, and (learners') dictionaries. The system addresses two key challenges: generating natural prosody for isolated words, and context-free disambiguation of homographs.

1 Introduction

Text-to-speech synthesis (TTS) is typically used to convert texts and sentences into speech. However, there are many applications that require the speech synthesis of isolated words: pronunciation training applications, speech and language therapy applications, (learners') dictionaries, etc. Such applications additionally require a careful and correct pronunciation of the synthesised words. To achieve this, the TTS system must fulfill two additional requirements beyond the general requirements for TTS systems. First, the training data must contain a sufficient amount of short utterances in order for the system to be able to generate isolated words with a natural utterance prosody. Second, the system must allow for a context-free disambiguation of input words that have phonologically different homographs. While the first requirement is unproblematic, the second requirement is a considerable challenge for a language like Estonian. Estonian possesses a large number of homographs that are mainly due to the absence of orthographic marking for two phonological features of Estonian: palatalisation and, in certain cases, third quantity (overlong length degree). This gives rise to two main types of homograph pairs: homographs differing in

palatalisation, and homographs differing in second quantity (Q2) vs. third quantity (Q3). Palatalisation in Estonian is, on the one hand, a coarticulatory phenomenon, meaning that all alveolar consonants /t, s, n, l/ preceding /i/ or /j/ at the boundary of the primary stressed syllable and the following syllable become palatalised. On the other hand, it is also a phonological phenomenon that distinguishes meaning (Metslang et al., 2023). The distinction between second and third quantity results from a difference in the prosodic structure of long stressed syllables, which can occur either in a disyllabic (Q2) or monosyllabic (Q3) foot (Metslang et al., 2023). Both palatalisation and quantity distinctions can be challenging for learners of Estonian as a second language and thus require attention in language pedagogy applications (Malmi et al., 2022b; Meister and Meister, 2014).

The homograph pairs differing in palatalisation are always (inflectional forms of) different lemmas whereas quantity distinguishes both between homographic lemmas and inflectional forms of the same lemma. For example, the orthographic form *tulp* represents both /tulp:/ 'signpost.NOM.SG' and /tul'p:/ 'tulip.NOM.SG', and *maitse* represents both /maitse/ 'taste.NOM.SG' and /mait'se/ 'taste.GEN.SG' or 'taste.IMP.2SG'. In addition, numerous words have pronunciation variants differing only in quantity or palatalisation. The Estonian Combined Dictionary (CombiDic) (Langemets et al., 2023) contains altogether 756 homographs and pronunciation variants differing in palatalisation, and 22,618 homographs and variants differing in quantity (excluding compounds). While the incorrect pronunciation of these homographs does not necessarily hinder comprehension in context, it does so without context and is particularly problematic in pedagogical applications.

A TTS system that is able to generate isolated words with a correct pronunciation must thus include a means for disambiguating homographs. Current supervised Estonian TTS systems include morphological parsing and disambiguation as part of their pre-processing pipeline. The standard morphological parser and disambiguator currently used in the Estonian TTS systems is Vabamorf¹ (Kaalep and Vaino, 2001). In addition to part-of-speech and inflectional categories the parser annotates compound boundaries and the following pronunciation features: third quantity, irregular stress, and palatalisation. Morphological parsing is followed by disambiguation; however, disambiguation is based on the probability of tag sequences within sentences and thus cannot be applied to isolated input words. As a result, the probability that an existing supervised TTS system generates the desired member of a homograph pair is at chance level. Likewise, the disambiguation of homographic input words is infeasible in unsupervised TTS systems, which may produce palatalisation, quantity, stress and compound identification errors also in words without homographs. In order to solve this problem, we developed a dedicated Estonian TTS system for generating isolated words with a correct pronunciation. Section 2 describes the development and the features of the system (training data, pre-processing, TTS technique, and user interface), Section 3 evaluates the performance of the system in terms of the pronunciation accuracy of homographic minimal pairs differing in palatalisation or quantity, Section 4 describes the planned and potential use cases of the system, and Section 5 presents the conclusion and future steps.

2 Development and features of the Estonian isolated-word TTS system

2.1 Training Data

The training data consisted of human-recorded sound files of isolated words and the corresponding text files. The sound files had been recorded for language pedagogy purposes by a female voice talent in a sound studio in order to exemplify the pronunciation of a subset of the headwords of the CombiDic (the basic vocabulary). The dataset consisted of a total of

31,215 words (10 h 36 min) with a good coverage of Estonian sounds and sound combinations and a high phonetic quality². The materials thus provided appropriate training data for ensuring a natural production of isolated words as utterances and a good phonetic coverage and quality suitable for pedagogical and speech therapeutic applications. The text versions of the words were drawn from the database of the CombiDic along with diacritics for third quantity, irregular stress, palatalisation, and compound boundaries. The annotation principles are based on Viks (1992)³ and are standardly used in Estonian dictionaries and parsers, including Vabamorf.

2.2 Pre-processing

The pre-processing did not include the standard stage of parsing as the input words were already annotated for the relevant features normally assigned by the parser. Otherwise, the standard pre-processing steps and grapheme-to-phoneme conversion used in Estonian TTS were applied (Mihkla et al., 2000).

2.3 TTS technique

We used the Merlin TTS toolkit developed by the Centre for Speech Technology Research (CSTR) at the University of Edinburgh⁴ (Wu et al., 2016). It is designed for building deep neural network models for statistical parametric speech synthesis. Merlin TTS was considered a suitable technique as it requires a relatively small amount of training data and allows good control. The model was developed specially for isolated word synthesis⁵ (Kiissel, 2024).

2.4 User interface

The synthesiser is available online via <https://elo.eki.ee/yksiksona/> (see Figure 1). The user must enter the word to be synthesised along with the appropriate diacritics for third quantity, palatalisation, irregular lexical stress and compound boundaries to obtain the desired pronunciation. The interface provides instructions for inserting the diacritics. To help users insert the necessary diacritics the web page will additionally include a Vabamorf interface for automatically annotating input words with morphological tags, compound boundaries, and pronunciation marks.

¹ <https://github.com/Filosoft/vabamorf/tree/master>

² The corpus is available at https://koneveeb.ee/korpused/#eva_yksiksonad (eva_yksiksonad_1, eva_yksiksonad_2).

³ see also <https://eki.ee/teatmik/haaldusmargid-uhendsonastikus-us/>

⁴ <https://github.com/CSTR-Edinburgh/merlin> and <https://www.cstr.ed.ac.uk/projects/merlin/>

⁵ https://github.com/ikiissel/mrln_et_iw

Users can download the synthesised pronunciations as WAV files.

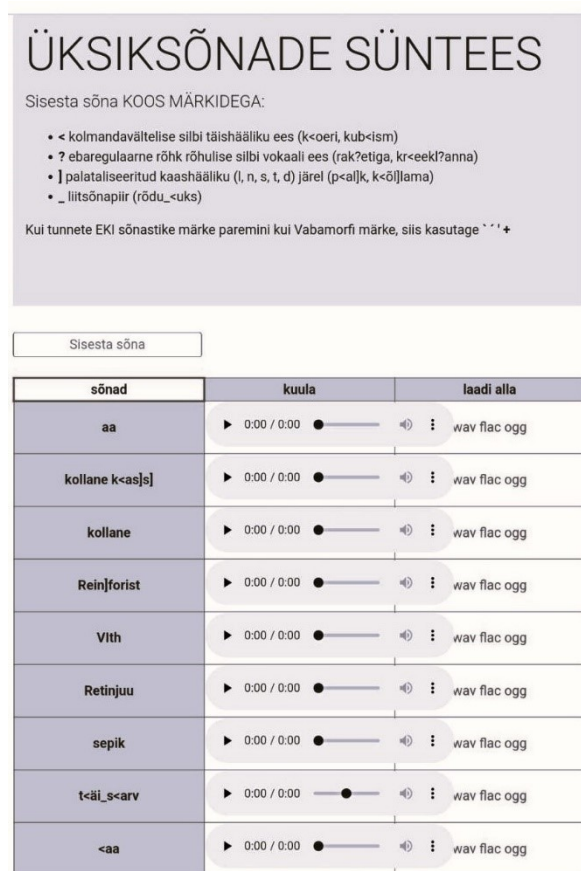


Figure 1: User interface of the synthesizer.

3 Evaluation

3.1 Materials, evaluators, procedure

We conducted a perception test to evaluate the performance of the TTS system in terms of pronunciation accuracy. We used 16 pairs of homographs that differ in palatalisation and 16 pairs of homographs that differ in quantity. Two types of monosyllabic word pairs were included for the evaluation of palatalisation: words ending with a long consonant like *kott*, *konn*, *tall*, and words with a consonant cluster like *palk*, *mulk*, *sulg*. The homographs distinguished by quantity were selected to include words with different syllable structures: (C)VCCV, e.g., *paksu*, *kommi*, *arve*; CVVCCV, e.g. *maitse*; CVVV, e.g., *saia*; CVVVCV, e.g. *heina*; CVCV, e.g., *hoone*.

All the items were synthesised using the diacritics corresponding to the two pronunciations, e.g., “p<al[k]” for /palːk:/ and

“p<alk” for /palk:/, and “kommi” for /kommi/ and “k<ommi” for /komːmi/.

The perception test was carried out online in the LimeSurvey⁶ environment. The task of the evaluators was to listen to each item and to answer one of the following questions, depending on the case: Is this word palatalised or not? Is this word in second or third quantity? There were in total 32 cases where the evaluators had to determine whether the word they heard has palatalisation or not, and 32 cases where they had to decide whether the word was in the second or third quantity⁷.

The evaluators were eight linguistics and phonetics experts who had previous experience in identifying both palatalisation and quantity.

3.2 Evaluation results

Palatalisation. Out of 32 homographs, 26 were correctly recognised by all the experts (100%). For the words /tulːp:/, /kotː:/ and /patːs/ the intended pronunciation was recognised by 88% of the experts, and for the words /jutː:/, /nutː:/, /mütːs/ by 75% of the experts. It appears that problems mainly arise with words involving /t/ and /tː/ (except for /tulːp:/). Given that all the test items were correctly recognised by a majority of the evaluators, the performance of the synthesiser can be considered very good. Occasional failures to perceive palatalisation were to be expected as palatalisation in Estonian has been found to be variable, weak, and gradient, and it has been noted that, especially in connected speech, experts’ opinions on the identification of palatalisation may not always coincide (Kalvik and Piits, 2019).

Quantity. The intended quantity of each test word was recognised by almost 100% of the evaluators. Only in the case of the word /maitse/ ‘taste.NOM.SG’ did one out of the eight experts fail to recognise that it was a Q2 form. For the remaining 31 word forms, all the experts recognised the intended quantity.

In summary, the performance of the isolated word synthesiser in terms of the phonetic accuracy of homographic words is very good, whereas the probability of obtaining a desired pronunciation variant with other Estonian TTS

⁶ <https://www.limesurvey.org/>

⁷ The materials and evaluations are available at <https://doi.org/10.6084/m9.figshare.27275964>

systems is only 50% due to the absence of disambiguation.

4 Use cases

The isolated-word TTS synthesiser allows the user to generate correctly pronounced isolated words and multi-word units by manually specifying the features of quantity, palatalisation, lexical stress and compound structure. The synthesiser generates isolated words with an appropriate utterance prosody and high phonetic quality, being thus suitable for language pedagogical and speech therapeutic purposes. Below, we describe three planned or potential use cases of the isolated-word synthesiser.

Generation of pronunciation examples for dictionaries. CombiDic currently uses TTS to generate the audio for example sentences. For headwords, the dictionary currently includes human-recorded pronunciation examples (used as the training data of the isolated-word synthesiser, see Section 2.1). However, pronunciation files are available only for the basic vocabulary, and only for three or four inflectional forms of inflecting words, depending on part-of-speech. The first application of the isolated-word synthesiser will therefore be the generation of pronunciation files for all the headwords and for all the inflectional forms in the CombiDic. In addition, pronunciation files are essential for learners' dictionaries, e.g., the Estonian Picture Dictionary⁸.

Pronunciation practice. The isolated-word synthesiser can be used to generate pronunciation examples for pronunciation training applications (for example, the pronunciation exercises created by the Institute of the Estonian Language⁹, and the Estonian pronunciation training app SayEst¹⁰ (Malmi et al., 2022a), which currently use human-recorded pronunciation examples), electronic and online teaching materials (e.g., the Estonian Language E-Course Keeleklikk¹¹), classroom practices and self-study. For instance, unlike the other Estonian TTS systems, the isolated-word synthesiser allows for a controlled synthesis of minimal pairs differing only in palatalisation, quantity, lexical stress, or the presence/absence or location of a compound boundary, which is useful

in the practice of the production and perception of these phonological features of Estonian.

Speech therapy exercises. The isolated-word synthesiser can also be used in speech therapy applications like Kõneravi.ee¹², where speech therapists can utilise existing exercises as well as create new ones. The available pronunciation and perception exercises use units at the phoneme, word, phrase, and sentence levels, words being the most frequently used perception or pronunciation units. So far, human-recorded audio examples have been used, which means that in order to create new exercises, the users must record the audio examples themselves or use examples from a limited speech database.

5 Conclusions and future work

The paper described the development, features, evaluation and use cases of the Estonian isolated-word TTS synthesiser (Kiissel, 2024 and <https://elo.eki.ee/yksiksona/>). The synthesiser allows the user to generate correctly pronounced isolated words and multi-word units by manually specifying the diacritics for third quantity, palatalisation, lexical stress and compound boundaries. The synthesiser generates isolated words with an appropriate utterance prosody and high phonetic quality, being thus suitable for language pedagogical and speech therapeutic applications.

Future steps include the improvement of the user-friendliness of the user interface. To help users insert the necessary diacritics, automatic tagging of the orthographic input text will be added, with multiple outputs for homographs among which the user can choose.

A second line of future work will be the development of a similar TTS application for longer texts, enabling the user to correct parsing and disambiguation errors that cause pronunciation errors.

Finally, we will employ more advanced TTS techniques to train additional isolated-word synthesisers.

⁸ <https://sonaveeb.ee/wordgame?uilang=en>

⁹ <https://sonaveeb.ee/pronunciation-exercises/#/>

¹⁰ Available in Google Play store <https://play.google.com/store/apps/details?id=mobi.lab.sayest&pli=1>

¹¹ https://www.keeleklikk.ee/index_en.html

¹² <https://koneravi.ee/>

Acknowledgments

This study was supported by the National Programme for Estonian Language Technology 2018–2027 and by the basic governmental financing of the Institute of the Estonian Language from the Estonian Ministry of Education and Research.

References

- Heiki-Jaan Kaalep and Tarmo Vaino. 2001. Complete Morphological Analysis in the Linguist's Toolbox. In Anu Nurk, Tõnu Seilenthal, and Triinu Palo, editors, *Congressus Nonus Internationalis Fenno-Ugristarum, 7.-13.8.2000 Tartu: Pars V*, Congressus Nonus Internationalis Fenno-Ugristarum, 7.-13.8.2000 Tartu, pages 9–16. Eesti Fennougristide Komitee.
- Mari-Liis Kalvik and Liisi Piits. 2019. Sõna esinemissagedus ja tähenduste eristamise vajadus häälduse mõjutajana. *Eesti ja soome-ugri keeleteaduse ajakiri. Journal of Estonian and Finno-Ugric Linguistics*, 10(1):71–88.
- Indrek Kiissel. 2024. Merlinil põhinev üksiksõnade kõnesüntesaator [Merlin based Estonian isolated word speech synthesizer]. https://github.com/ikiissel/mrln_et_iw.
- Margit Langemets, Indrek Hein, Madis Jürviste, Jelena Kallas, Olga Kiisla, Kristina Koppel, Külli Kuusk, Tiina Leemets, Sirje Mäearu, Tiina Paet, Peeter Päll, Maire Raadik, Lydia Risberg, Tuuli Rehema, Hanna Tammik, Mai Tiits, Katrin Tsepelina, Maria Tuulik, Udo Uibo, et al. 2023. *EKI ühend sõnastik*. 2023. [The EKI Combined Dictionary]. Eesti Keele Instituut. <https://sonaveeb.ee>.
- Anton Malmi, Katrin Leppik, and Pärtel Lippus. 2022a. SayEst - mobiilirakendus eesti keele häälduse harjutamiseks. *Oma Keel*, 2:85–88.
- Anton Malmi, Pärtel Lippus, and Einar Meister. 2022b. Articulatory properties of Estonian palatalization by Russian L1 speakers. *Eesti ja soome-ugri keeleteaduse ajakiri. Journal of Estonian and Finno-Ugric Linguistics*, 13(2).
- Einar Meister and Lya Meister. 2014. L2 production of Estonian quantity degrees. In *Speech Prosody 2014*, pages 929–933. ISCA.
- Helle Metslang, Mati Ereht, Külli Habicht, Tiit Hennoste, Reet Kasik, Pire Teras, Annika Viht, Eva Liina Asu, Liina Lindström, Pärtel Lippus, Renate Pajusalu, Helen Plado, Andriela Rääbis, and Ann Veismann. 2023. *Eesti grammatika*. Tartu Ülikooli Kirjastus.
- Meelis Mihkla, Einar Meister, and Arvo Eek. 2000. Eesti keele tekst-kõne süntees: grafeem-foneem teisendus ja prosoodia modelleerimine. In Tiit Hennoste, editor, *Arvutuslingvistikalt inimesele*, Tartu Ülikooli üldkeeleteaduse õppetooli toimetised, pages 309–320. Tartu Ülikooli kirjastus, Tartu.
- Ülle Viks. 1992. *Väike vormisõnastik: Sissejuhatus ja grammatika*. Eesti Teaduste Akadeemia, Keele ja Kirjanduse Instituut.
- Zhizheng Wu, Oliver Watts, and Simon King. 2016. Merlin: An Open Source Neural Network Speech Synthesis System. In *Proc. 9th ISCA Workshop on Speech Synthesis Workshop (SSW 9)*, pages 202–207. ISCA.

BiaSWE: An Expert Annotated Dataset for Misogyny Detection in Swedish

Kätriin Kukk^{1,2}, Danila Petrelli¹, Judit Casademont¹,
Eric J. W. Orlowski³, Michał Dzieliński⁴, Maria Jacobson⁵

¹AI Sweden

²Linköping University

³AI Singapore

⁴Stockholm University

⁵Anti-Discrimination Agency West Sweden

katriin.kukk@liu.se, danila.petrelli@ai.se,
juditcasademont@gmail.com, ericorlowski@aisingapore.org,
michal.dzielinski@sbs.su.se, maria.jacobson@adbvast.se

Abstract

In this study, we introduce the process for creating BiaSWE, an expert-annotated dataset tailored for misogyny detection in the Swedish language. To address the cultural and linguistic specificity of misogyny in Swedish, we collaborated with experts from the social sciences and humanities. Our interdisciplinary team developed a rigorous annotation process, incorporating both domain knowledge and language expertise, to capture the nuances of misogyny in a Swedish context. This methodology ensures that the dataset is not only culturally relevant but also aligned with broader efforts in bias detection for low-resource languages. The dataset, along with the annotation guidelines, is publicly available for further research.

1 Introduction

Large Language Models (LLMs) have experienced immense growth over the past years due to being capable of solving diverse tasks that previously required a separate model for each specific task (De Angelis et al., 2023). Despite their apparent benefits, it is known that the characteristics of the dataset used to train a language model play a fundamental role in determining the model’s behavior (Geburu et al., 2021). LLMs are typically trained on large amounts of data from the Internet and thus inevitably reflect the opinions and biases of its users. For example, a 2018 survey showed

that about 85% of English Wikipedia contributors identified as male (Oldach, 2022). As LLMs’ behavior “reflects the Collective Intelligence of Western society”, LLMs can perpetuate and even amplify biases and stereotypes of social minorities (Kotek et al., 2023). The widespread presence of misogyny online is illustrated by a study from 2020 where 65% of women reported knowing another woman that had been the target of online violence (The Economist Intelligence Unit, 2020).

The way to avoid harmful machine learning models is to ensure that the datasets used for training are responsibly curated, involving diverse stakeholders (Delgado et al., 2021). However, dataset creation alone is not sufficient, and additional approaches, such as alignment, play a role in guiding model outputs towards human values. In the context of bias detection, misogyny varies by language and culture (Zeinert et al., 2021). Therefore, we consider creating expert-annotated, language-specific datasets crucial for detecting biases, helping to identify areas where models may risk perpetuating harmful stereotypes or undesirable attitudes.

To address these challenges, we make two key contributions¹:

1. We present BiaSWE, a small annotated dataset for misogyny detection in Swedish, annotated for hate speech, misogyny, misogyny type categories and severity.
2. We share the creation process of the BiaSWE

¹Link to the dataset and annotation guidelines:
<https://huggingface.co/datasets/AI-Sweden-Models/BiaSWE>

dataset and our annotation guidelines. By doing this, we show how an existing experiment can be adapted to the Swedish needs and cultural context.

2 Related Work

In recent years, work has been done in the field of dataset creation for bias and hate speech in general, paying great attention to data coming from online sources, especially social media, such as Twitter, Facebook, Reddit, or blogs. This kind of work has been carried out in a multitude of languages, across several cultural contexts, and tends to cover various forms of sexism as it presents in written language. This is the case of Chiril et al. (2020), who present a corpus for detecting sexism in French tweets. Another example is the work of Zeinert et al. (2021), who sample their Bajer dataset from Twitter, Facebook and Reddit posts in Danish. However, research is also carried out with the target of more subtle, less explicit misogyny in mind; this is the case of the Biasly dataset by Sheppard et al. (2024), who gathered their data from scripts from North American movies, in English.

The most common method for data collection among the different existing datasets is using keywords (Chiril et al., 2020; Zeinert et al., 2021; Sheppard et al., 2024). The degree of detail or the number of keywords varies from words that do not necessarily imply misogyny (e.g. “she”) to ambiguous keywords, to keywords that are very highly related to misogyny and sexism (e.g. “#MeToo”).

Most of the existing misogyny detection datasets provide a taxonomy for different categories of misogyny in addition to the binary classification. Many regard the addition of a multi-label classification layer as necessary, given that “binary detection [...] disregards the diversity of sexist content, and fails to provide clear explanations for why something is sexist” (Kirk et al., 2023). There is no clear consensus regarding the types of misogyny to classify the sentences into, or even on the optimal level of detail regarding the categories.

To the best of our knowledge, this work is the first attempt to create resources for misogyny detection for the Swedish language.

3 Method

This section provides an overview of our data preparation process, introduces the team of expert

annotators and details the annotation workflow.

3.1 Data

As a data source for our dataset, we used the Swedish website Flashback, one of the largest Swedish internet forums since the 1990s. Known for its focus on freedom of speech, the forum hosts discussions on controversial subjects, and its anonymity often leads to misuse (Norlund and Stenbom, 2021).

To ensure the presence of enough misogynistic examples in the final dataset, we decided to use keyword search. Taking into account the cultural and linguistic closeness of Danish and Swedish, our initial list of keywords was based on the work of Zeinert et al. (2021) that used both keywords and hashtags in Danish. We excluded all hashtags but “#MeToo” because of their rarity on Flashback. The keywords were first translated from Danish into Swedish with the help of a Danish speaker. Thereafter, we presented the resulting Swedish keywords to our team of expert annotators, who suggested removing some of the keywords and adding others. Our list consists of 118 key terms including words and phrases in Swedish (e.g., “kvinna”) as well as some terms and names in popular English slang (e.g., “Chad”). The full list of keywords is available in the annotation guidelines (see section 1).

Based on these keywords, we gathered 450 data points, each to be annotated by two or more annotators. We did not want several annotators to have the exact same set of data points, so we used a rotation system that distributed them. Each expert was assigned 210 data points.

3.2 Annotation

The team of annotators included researchers and experts from the humanities and social sciences, as well as civil society actors. Four of our experts identify as women and the other three as men. Our experts volunteered to participate in the project amongst a bigger pool of experts in humanities, social sciences, and civil society representatives that have been introduced to the basics of LLMs and AI within a broader interdisciplinary project at AI Sweden². From this point on, we refer to them as ‘experts’, acknowledging their role in both

²Link to the project page:
<https://www.ai.se/en/project/interdisciplinary-expert-pool-nlu>

annotation and providing critical insights into the interdisciplinary process.

To facilitate the annotation process, we prepared annotation guidelines (see section 1) by taking inspiration from the work by Sheppard et al. (2024). Their guidelines included a definition of misogyny that was modified together with the team of experts to obtain the following final definition³:

Hatred of, dislike of, contempt for, ingrained prejudice, control of or oppression against women as well as going against the idea of feminism. It is a form of sexism and can be either intentional or unintentional. Misogyny can contain different types of opinions and values such as seeing women as inferior, asserting men’s sexual entitlement, objectifying women, accepting violence, celebrating traditional gender roles but also the need to ”protect” women as well as thinking that equality and feminism have gone too far. One of the ways misogyny can be expressed is through language and, in this project, we focus on misogynistic language portrayed in text. Misogyny can be perpetrated by people regardless of their gender.

We also present a taxonomy of five misogyny categories constructed by combining the twelve categories by Sheppard et al. (2024) and the six categories by Zeinert et al. (2021) and modifying these with the help of the team of experts. The guidelines also give detailed instructions on how to carry out the annotation task and provide examples. For annotation, we used an open-source platform called Label Studio. After everyone had annotated up to 50 examples, we held a workshop to discuss examples where the experts had disagreed.

We divided the annotation of each data point into four small tasks, each more fine-grained than the previous. Once a negative answer was given by the expert, the annotation process ended and the downstream tasks did not need to be completed.

Hate speech Our experts were asked to perform binary classification of whether the post they were reading contained hate speech or not. This gave the experts the chance to mark any type of hate speech or hateful behavior.

³Link to the complete guidelines, as well as the definition in Swedish, can be found on Hugging Face (see Footnote 1).

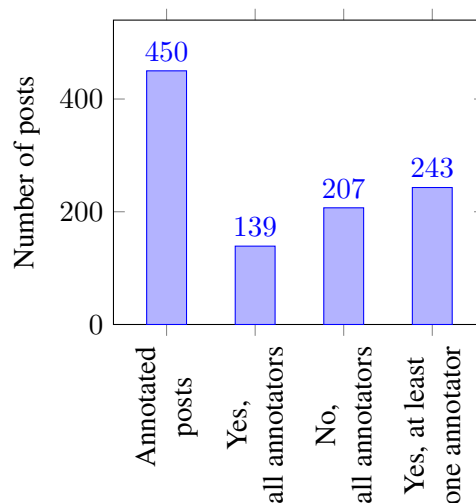


Figure 1: Is this post misogynistic?

Misogyny The second task was the binary classification of misogyny based on the instructions and the definition of misogyny provided in the annotation guidelines.

Category Once a post was classified as misogynistic, our experts were requested to choose a category label. Our taxonomy of misogyny consists of the following categories: stereotype, erasure and minimization, violence against women, sexualisation and objectification, anti-feminism and denial of sexualisation. The experts could only choose one category and could not choose a subcategory outside of the ones presented.

Severity Experts hold that misogyny exists in a spectrum and it depends on individual perception. To portray this, we asked them to give a score ranging from 1 to 10, where 1 is the least misogynistic. Although one would assume that, for example, a post portraying violence would have a high score, we did not give them any specific guidelines they had to follow to assign these scores and asked them to trust their own judgement.

4 Results

This section provides an overview of the annotation results.

Hate speech Each of the 450 posts was annotated by two to four experts and in almost two-thirds of the cases all experts agreed on whether hate speech was present. In the 334 cases where all experts agreed, slightly more posts were annotated as containing hate speech compared to not containing hate speech but the difference was marginal. However, there was a large difference

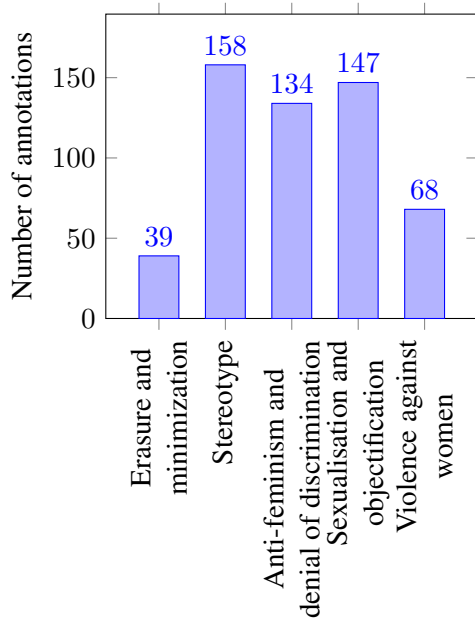


Figure 2: Which category of misogyny does this example belong to?

between the number of posts that were annotated as hate speech by all experts (172) and posts that were considered to be hate speech by at least one expert (288).

Misogyny Figure 1 shows the annotation results for the misogyny classification task. A negative label in the previous task is considered to be a negative label in the misogyny classification task as well, in which case, all 450 posts were again annotated by two to four experts. In slightly more than two-thirds of the examples, all experts agreed on the label. However, in this case, there was a larger imbalance between the two possible labels: 207 posts were considered to be non-misogynistic and 139 misogynistic by all experts. The number of posts considered to be misogynistic by at least one expert was however larger at 243 posts.

Category There was more disagreement in choosing the category of misogyny. There were between zero and four category annotations for each of the 450 posts in the dataset. Out of the 185 posts in the final dataset with more than one category annotation, in 93 cases all experts chose the same category. Figure 2 gives an overview of all 546 category annotations in the dataset, comparing the number of times each of the five possible categories was chosen.

Severity The last annotation task asked the experts to estimate the severity of the misogyny in the post. Similarly to the previous task, 185 posts

had at least two severity annotations and a closer analysis of those revealed that although the experts seldom selected the same rating, in 91% of cases the difference between the minimum and the maximum rating was not larger than 3.

5 Discussion and Conclusion

This project’s primary contribution lies in its interdisciplinary approach to misogyny detection in Swedish rather than the dataset itself, which remains small. Collaborating with experts from diverse fields, we developed an annotation process that captures the complexity of misogyny as it manifests in Swedish online discourse. This experiment provides a valuable framework for future studies focused on bias detection in under-resourced languages.

In the context of misogyny detection, defining what constitutes misogynistic language is inherently challenging. Attempting to capture a wider range of potentially harmful expressions risks being too broad, while using a stricter approach might fail to recognize subtler forms of misogyny. The challenge lies in determining who defines misogyny, as cultural, linguistic and societal factors have an influence over the definition, making it a complex decision.

The feedback from the experts highlighted the need for clearer operational definitions and stronger contextual support. Misogyny detection, particularly in a complex environment like Flashback, would benefit from additional discussion on cultural nuances and interdisciplinary perspectives. Additionally, better alignment between academic rigor and practical applicability is crucial to ensuring that interdisciplinary projects like this one fully realize their potential. We also took into account the experts’ perspective in section 6.

In conclusion, while the dataset is limited, the interdisciplinary approach and methodology offer a valuable starting point for future research. Refining the annotation process and expanding the dataset could further improve the effectiveness of misogyny detection tools, especially for lower-resourced languages like Swedish.

6 Limitations and Future Work

This project has several limitations that provide avenues for future work.

Dataset Size and Diversity The current dataset, while robustly annotated, is relatively small, lim-

iting its capacity to capture the full spectrum of misogynistic expressions within the Swedish on-line discourse. The limited number of examples might not adequately represent less overt forms of misogyny, which are increasingly prevalent and harmful. Future work should focus on expanding the dataset to include a larger variety of sources.

Keyword Selection Bias The reliance on pre-defined keywords to scrape forum posts inherently introduces selection bias, primarily focusing on explicit forms of misogyny. This method may overlook subtle or emergent forms of misogynistic language that do not necessarily conform to expected patterns. Future iterations of this project should aim to refine the keyword selection process. Additionally, incorporating machine learning techniques to identify potential posts could reduce bias introduced by keyword dependency.

Decontextualisation A key challenge was annotating decontextualised posts, which made it difficult to detect subtle misogyny. Without context, the experts had to rely on isolated phrases, often missing nuances that could clarify intent or severity. Providing more context in future datasets would enhance accuracy.

Consensus Building Disagreements among the experts highlighted the subjective nature of misogyny detection and the challenges in classifying complex human behaviors and attitudes. While we utilized workshops to align annotator perspectives, a more systematic approach to handling disagreement could enhance the consistency and reliability of annotations. Future work could include developing detailed guidelines based on the initial rounds of annotation to standardize responses and improve inter-annotator reliability. Implementing an adjudication process where the experts discuss and resolve disagreements before finalizing annotations could also be beneficial.

Acknowledgments

This work is a result of the “Interdisciplinary Expert Pool for NLU” project funded by Vinnova (Sweden’s innovation agency) under grant 2022-02870.

Experts Involved

- Annika Raapke, Researcher at Uppsala University, Department of History
- Eric Orlowski, Sociocultural Anthropologist, Research Fellow (AI Governance), AI

Singapore

- Michał Dzieliński, Assistant Professor at Stockholm Business School, International Finance
- Maria Jacobson, Anti-Discrimination Agency West Sweden
- Astrid Carsbrin, Swedish Women’s Lobby
- Cia Bohlin, The Swedish Internet Foundation
- Richard Brattlund, The Swedish Internet Foundation

Special Thanks

Special thanks to:

- Francisca Hoyer at AI Sweden for making the Interdisciplinary Expert Pool possible from the start
- Magnus Sahlgren at AI Sweden for guidance
- Allison Cohen at MILA AI for Humanity for participation and support during the experiment

BiaSWE’s multi-disciplinary engagement process was, in part, inspired by the Biasly project from Mila - Quebec AI Institute.

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

References

- Patricia Chiril, Véronique Moriceau, Farah Benamara, Alda Mari, Gloria Origgi, and Marlène Coulomb-Gully. 2020. <https://aclanthology.org/2020.lrec-1.175> An annotated corpus for sexism detection in French tweets. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1397–1403, Marseille, France. European Language Resources Association.
- Luigi De Angelis, Francesco Baglivo, Guglielmo Arzilli, Gaetano Pierpaolo Privitera, Paolo Ferragina, Alberto Eugenio Tozzi, and Caterina Rizzo. 2023. <https://doi.org/10.3389/fpubh.2023.1166120> Chatgpt and the rise of large language models: the new ai-driven infodemic threat in public health. *Frontiers in Public Health*, 11.
- Fernando Delgado, Stephen Yang, Michael Madaio, and Qian Yang. 2021. Stakeholder participation in ai: Beyond “add diverse stakeholders and stir”. In *Proceedings of the Human-Centered AI Workshop at NeurIPS 2021*.
- Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. 2021.

<https://doi.org/10.1145/3458723> Datasheets for datasets. *Commun. ACM*, 64(12):86–92.

Hannah Kirk, Wenjie Yin, Bertie Vidgen, and Paul Röttger. 2023. <https://doi.org/10.18653/v1/2023.semeval-1.305> SemEval-2023 task 10: Explainable detection of online sexism. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2193–2210, Toronto, Canada. Association for Computational Linguistics.

Hadas Kotek, Rikker Dockum, and David Sun. 2023. <https://doi.org/10.1145/3582269.3615599> Gender bias and stereotypes in large language models. In *Proceedings of The ACM Collective Intelligence Conference, CI '23*, page 12–24, New York, NY, USA. Association for Computing Machinery.

Tobias Norlund and Agnes Stenbom. 2021. <https://aclanthology.org/2021.nodalida-main.38> Building a Swedish open-domain conversational language model. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 357–366, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.

Laura Oldach. 2022. What’s with wikipedia and women? Retrieved April 19, 2024, from <https://www.asbmb.org/asbmb-today/careers/030822/what-s-with-wikipedia-and-women>.

Brooklyn Sheppard, Anna Richter, Allison Cohen, Elizabeth Smith, Tamara Kneese, Carolyne Pelletier, Ioana Baldini, and Yue Dong. 2024. <https://doi.org/10.18653/v1/2024.findings-acl.24> Biasly: An expert-annotated dataset for subtle misogyny detection and mitigation. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 427–452, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

The Economist Intelligence Unit. 2020. Measuring the prevalence of online violence against women. Infographic. Retrieved October 11, 2024 from <https://example.com/online-gender-gap-infographic>.

Philine Zeinert, Nanna Inie, and Leon Derczynski. 2021. <https://doi.org/10.18653/v1/2021.acl-long.247> Annotating online misogyny. 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 3181–3197, Online. Association for Computational Linguistics.

Predictability of Microsyntactic Units across Slavic Languages: A Translation-based Study

Maria Kunilovskaya, Iuliia Zaitova, Wei Xue, Irina Stenger, and Tania Avgustinova

University of Saarland
izaitova@lsv.uni-saarland.de

Abstract

The paper presents the results of a free translation experiment, which was set up to explore Slavic cross-language intelligibility. In the experiment, native speakers of Russian were asked to read a sentence in one of the five Slavic languages and return a Russian translation of a highlighted item. The experiment is focused on microsyntactic units because they offer an increased intercomprehension difficulty due to opaque semantics. Each language is represented by at least 50 stimuli, and each stimulus has generated at least 20 responses. The levels of intercomprehension are captured by categorising participants' responses into seven types of translation solutions (paraphrase, correct, fluent_literal, awkward_literal, fantasy, noise, and empty), generally reflecting the level of the cross-linguistic intelligibility of the stimuli. The study aims to reveal linguistic factors that favour intercomprehension across Slavic languages. We use regression and correlation analysis to identify the most important intercomprehension predictors and statistical analysis to bring up the most typical cases and outliers. We explore several feature types that reflect the properties of the translation tasks and their outcomes, including point-wise phonological and orthographic distances, cosine similarities, surprisals, translation quality scores and translation solution entropy indices.

The experimental data confirms the expected gradual increase of intelligibility from West-Slavic to East-Slavic languages for the speakers of Russian. We show that intelligibility is highly contingent on the

ability of speakers to recognise and interpret formal similarities between languages as well as on the size of these similarities. For several Slavic languages, the context sentence complexity was a significant predictor of intelligibility.

1 Introduction

Cross-linguistic intercomprehension (receptive multilingualism) is defined as a phenomenon where speakers of different but related languages can communicate without studying each other's language (Trudgill, 2003). It can be viewed as specific cognitive conditions that tap into the mechanisms of human language processing (Meulleman and Fiorentino, 2018). Previous studies have focused on various aspects of intercomprehension within different language groups (Gooskens and Swarte, 2017; Stenger et al., 2017; Jagrova et al., 2018).

Some studies (Zaitova et al., 2024b,a) have looked at cross-linguistic intelligibility of functional multiword expressions with non-compositional semantics, called microsyntactic units (MSUs) (Avgustinova and Iomdin, 2019). MSUs can be grouped with prepositions, conjunctions, particles and other such word classes based on their function in the sentence. They are an interesting object for language processing studies because they are often important as discourse structuring items, signalling relations between clauses or conveying the speaker's attitude. Their intelligibility implies at least some understanding of the underlying proposition. Besides, MSUs present an additional difficulty for comprehension, especially across languages, because their meaning cannot be inferred from the components. An example of MSU in English is *all the same* or in Russian *тем не менее* (translit.: "tem ne menee", "nevertheless").

The exact mechanisms of intercomprehension

employed to process MSUs under various cross-linguistic conditions are still under-researched. To address this gap, our study presents the analysis of a free translation experiment in which native speakers of Russian translated MSUs from five Slavic languages (Czech, Polish, Bulgarian, Belarusian, and Ukrainian, hereinafter referred to as source languages) into Russian. We only used the data if the participants reported no training or exposure to the respective Slavic language.

The study aims to assess the level of intelligibility of the five Slavic languages for Russian speakers and to reveal the factors contributing to it. To this end, the translation solutions offered by native Russian speakers when rendering foreign MSUs into Russian are analysed. We employ several computational features (phonological distance, cognitive metrics, and translation quality scores) and provide a quantitative and qualitative description of Slavic MSU intelligibility as manifested by the participants' responses in the translation experiment.

It is expected that the East-Slavic languages (Belarusian and Ukrainian) would return the highest degree of intercomprehension, i.e., they would have the lowest difficulty in translation because Russian also belongs to the East-Slavic languages, followed by the South-Slavic Bulgarian (due to the use of Cyrillic script), with the Latin script-based West-Slavic languages (Czech, Polish) demonstrating the highest difficulty for the participants. Generally, translation difficulty indicators are expected to be reliable predictors of translation quality, i.e., of the outcomes of the translation task in this study¹.

2 Free Translation Experiment

Data collection: Platform, task and participants. The free translation experiment was held online² and aims to measure the degree of intelligibility of the targeted MSUs in the Slavic languages for Russian native speakers. The targeted MSU items come from a multi-parallel set, centred on Russian, which makes them comparable across the languages involved. In total, the experiment involved 126 unique participants without prior knowledge of the Slavic language they were

translating from and 6,579 responses. The translation tasks for each Slavic language include between 50 and 60 unique sentences containing one of the target items. The study engaged from 101 to 121 native Russian participants per Slavic language who did not have any formal knowledge of that language. Table 1 provides basic descriptive statistics of the experimental data and participants. As can be seen from the table, the data is well-balanced across the languages in terms of the number of phrases and their part-of-speech category (PoS). There is approximately the same number of unique participants per language and the same number of responses per phrase.

	MSUs	ppt.	ppt./task	MSUs/PoS
CS	60	121	24.2±4.7	12.0±0.0
PL	50	116	23.1±5.3	10.0±1.3
BG	56	122	24.4±6.5	11.2±0.4
BE	57	121	24.4±5.4	11.4±0.5
UK	59	101	20.5±4.0	11.8±0.4

Table 1: Quantitative parameters of the free translation experiment. Abbreviations: ppt. (participants), CS (Czech), PL (Polish), BG (Bulgarian), BE (Belarusian), UK (Ukrainian)

Annotation of translation solutions and intelligibility scores. The participants' responses from the free translation experiment were categorised into seven groups of translation solutions reflecting the types of linguistic behaviour as well as the degree of understanding. These categories are explained below (in the order of decreasing intelligibility of the annotated response):

correct: a translation variant, which coincides with the reference ('gold') translation in cases where the available literal translation is different from the gold translation (otherwise, the response is categorised as 'fluent_literal'); it is the most expected standard solution that signals good understanding of the source phrase or even sentence,

fluent_literal: an acceptable translation variant, which coincides with both gold and literal translations; the cases where exploiting the cross-linguistic parallels yields good results,

paraphrase: a translation variant, which does not coincide with either gold or literal translation but faithfully renders the meaning of the

¹Our code and datasets are available at <https://github.com/SFB1102/b7-c4-slavic-translation-nodalida2025>.

²<https://intercomprehension.coli.uni-saarland.de/en/>

source phrase; this can be a less expected descriptive response,

awkward_literal: this is a type of literal translation which is neither *fluent_literal* nor semantically incorrect, a translation technique to fall back to perceived cross-lingual similarities,

fantasy: a translation variant, which misrepresents the content of the source in the target language signalling lack of understanding,

noise: an irrelevant input, which does not allow to infer any specific translation solution; noisy solutions sometimes include comments like ‘I have no idea’ and ‘I don’t understand’,

empty: no input provided indicating that the participant could not come up with a translation solution in the given time.

Note that this categorisation is developed for the purposes of this study and does not reflect translation quality of the participants’ responses.

As can be seen from the description, the categorisation relied on existing gold and literal translations. The gold translations for the MSUs were extracted from the parallel subcorpora of the Russian National Corpus³ and of the Czech National Corpus⁴ with Russian as a target language (for more details see Zaitova et al., 2024a). The literal translations were generated by GPT-4 (22 July 2024) for isolated MSUs, i.e., for MSU outside of their context. To obtain literal translations, we used a prompt that included the task description “Return a literal word-for-word translation for a phrase in one of the Slavic languages into Russian.”, a one-shot example in Czech and the task itself containing the name of the stimulus language and the phrase to translate. Automatic literal translations were preferred to human-generated literal translations to avoid subjective biases with regard to what was a literal translation. The sanity of the GPT-4 literal translations was controlled manually on an approximately 20% sample from each of the stimulus languages. The participants’ responses were first pre-annotated for ‘empty’, ‘correct’, ‘fluent_literal’ and ‘awkward_literal’ categories because these annotations could have been filled in automatically based on matching gold and/or literal translations (see their description

above). Two human annotators – trained linguists specialising in the Slavic languages and native speakers of Russian – contributed annotations for the remaining categories following formal and exemplified annotation guidelines. The annotators had access to gold and literal translations, as well as to the source language contexts. Conflicting annotations were resolved in a post-annotation discussion session.

To represent the overall intelligibility of the MSUs in a stimulus language for a Russian speaker, we assigned intelligibility weights to the annotated translation solutions on the following scheme: ‘correct’: 7, ‘fluent_literal’: 6, ‘paraphrase’: 5, ‘awkward_literal’: 4, ‘fantasy’: 2, ‘noise’: 0, ‘empty’: 0. The higher weights indicate greater intelligibility. The aggregate **intelligibility score** for each MSU item was calculated as a sum of weighted response probabilities across all responses for that stimulus. For example, the probabilities of responses for the Belarusian particle *ледзьве не* [hardly] had probabilities of the translation solutions distributed as follows: 0.0625, 0.0, 0.0625, 0.03125, 0.40625, 0.125, 0.3125. The sum of weighted probabilities is 1.6875.

3 Feature Extraction and Regression Analysis

Feature extraction. Generally, we explored four types of features: (a) surprisal values and (b) cosine similarities, both based on a pre-trained Transformer model, (c) Phonologically Weighted Levenshtein Distance (PWLD), and (d) automatic translation quality scores. These features were extracted for every source language items using gold and literal translations. We provide additional details on feature calculation below. Note that contextualised items were required when extracting some of the features, namely surprisals, cosine similarities, and automatic quality scores. Recall that the literal translations from GPT-4 were isolated phrases, not entire sentences. Therefore, we generated sentence-level contexts for these items by replacing them with the GPT-4 literal translations in the contexts from the parallel corpora.

(a-b) Transformer-based features: Surprisal and cosine similarity values reported in this study were generated using ruRoBERTa-large model (Zmitrovich et al., 2024)⁵, a dedicated Russian language

³<https://ruscorpora.ru/en/>

⁴<https://www.korpus.cz/>

⁵<https://huggingface.co/ai-forever/ruRoberta-large>

Transformer⁶. To get a surprisal value for an MSU, we summed up surprisals of its components. The sentence-level surprisals are averaged across all words in a sentence, with the word-level surprisal being a sum of subword token surprisals. Cosine similarities were calculated using MSU embeddings that were mean-pooled across word-level embeddings of MSU components. The word embeddings were generated from subword representations using *minicons* python library.⁷ Care was taken to minimise the number of extraction errors caused by mismatching tokenisation for isolated and contextualised MSUs, and by overmatching MSU components in a sentence. Specifically, we extracted surprisal values for the source, gold and literal MSUs themselves (surprisal_stim, surprisal_gold and surprisal_lit) and average surprisals for the sentences containing them (surprisal_stim_sent, surprisal_gold_sent and surprisal_lit_sent). The cosine similarity was calculated between (1) the stimulus source items in the five Slavic languages and their gold translations (cosine_stim_gold), and (2) the stimuli and their literal translations (cosine_stim_lit).

(c) PWLD: PWLD is a metric of weighted phonological similarity based on the Levenshtein distance between two phonemic sequences (Fontan et al., 2016). It takes into account the cost of each phoneme substitution given their phonemic features. We use an adaption of the PWLD proposed in Abdullah et al. (2021). PWLD is more suitable for cross-linguistic analysis than Levenshtein Distance because PWLD can catch more fine-grained phonological similarities. For example, in the pair of Czech and Russian cognates *ucho* /u x o/ and *yxu* /u x ɔ/, where phonemes /o/ and /ɔ/ are very similar to each other, PWLD would capture this similarity more effectively compared to Levenshtein Distance. To obtain the IPA transcriptions of all stimuli, we used *Char-siuG2P*, a transformer-based tool for grapheme-to-phoneme conversion (Zhu et al., 2022). We extracted PWLD scores between (1) the stimulus items and their literal translations (pwld_stim_lit), (2) the stimulus items and their gold translations (pwld_stim_gold), and (3) gold and literal translations (pwld_gold_lit).

tions (pwld_gold_lit).

(d) Automatic translation quality scores: We use scores from the reference-based and reference-free pre-trained COMET models⁸. The reference-based score was used to generate translation quality scores for literal translations, with the gold translation as reference. Additionally, we used reference-free quality scores (translation quality estimation scores) for the gold (qe_gold), literal (qe_lit), and participants' translations (eval_lit).

MSU translation entropy as an alternative to intelligibility score. The intelligibility score is based on annotated translation solutions, and thus takes into account **types of responses** abstracting from the individual choices. A more straightforward approach to judge about translation difficulty of an item is to calculate the its translation entropy from the distribution of valid translation variants seen in the data. We used the Shannon entropy formula:

$$H = - \sum_{i=1}^n p_i \log_2(p_i) \quad (1)$$

where p_i denotes the probability of the i -th unique response, and n denotes the total number of unique responses. The responses annotated as noise or empty were considered as having a `None` value. Shannon entropy captures the unpredictability of responses and can be interpreted as a measure of translation task difficulty: the higher the entropy, the more difficult the translation task is (Wei, 2022). It can also be views as a measure of literality: low entropy signals conditions for more automated literal translation (Carl and Schaeffer, 2017).

In sum, the analysis is based on 14 features shown in Appendix A. The Appendix reflects Pearson correlation of each feature with the entropy and intelligibility score for the source MSUs in each language, highlighting indicators that returned significant results. It can be seen that at least in terms of univariate analysis intelligibility scores are better aligned with the proposed features than entropy.

Regression analysis. The relevance of the features for intercomprehension was explored through their ability to predict the intelligibility score in a regression setup. The regression

⁶We also tried other Russian transformers such as https://huggingface.co/ai-forever/rugpt3large_based_on_gpt2, which returned similar results (omitted here for brevity).

⁷<https://pypi.org/project/minicons/>

⁸<https://huggingface.co/Unbabel/wmt22-comet-da> and <https://huggingface.co/Unbabel/wmt22-cometkiwi-da> respectively, described in Rei et al., 2022

was performed using Support Vector Machine algorithm (SVR) as implemented in *scikit-learn*.⁹ The performance of SVR is reported in terms of Pearson’s correlation coefficient (r) and Mean Absolute Error (MAE) with corresponding two-sided standard deviation across the 10 runs of the experiment (\pm). The error reported for intelligibility score as the response variable across all languages was lower than can be obtained by predicting the mean of the scores. This was not the case for entropy as an alternative response variable. Feature selection was performed using the Recursive Feature Elimination (RFE) technique, which iteratively applied a linear regressor to the feature space, eliminating the least important feature in each iteration until the desired number of features (here, N was arbitrarily set to 5) was reached.

4 Results and Discussion

4.1 Translation Solutions and Intelligibility

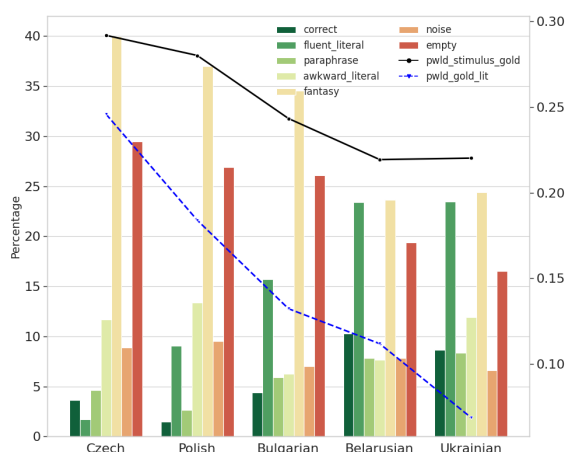


Figure 1: Bars for translation solutions and line plots for mean PWLD between original and gold MSUs (`pwld_stim_gold`), and between gold and literal variants (`pwld_gold_lit`), with PWLD values on the left y-axis. The greener end of the spectrum marks more successful translation task completion.

Figure 1 shows the distribution of translation solutions for each source language. The translation solutions are colour-coded and ordered based on the declining degree of intelligibility from the green end of the spectrum towards red. It can be seen that the percentage of correct translations (height of greener bars) increases from left to right

⁹<https://scikit-learn.org/1.5/modules/generated/sklearn.svm.SVR.html>

across the languages. The intelligibility of the Slavic languages for speakers of Russian (as one can see from the bar charts) increases from left to right, i.e., from Czech to Ukrainian.

The lines in Figure 1 represent the PWLD values (i) between the original MSUs in Slavic languages and their gold translations attested in parallel corpora (`pwld_stim_gold`; solid black line), and (ii) between gold and literal translations (`pwld_gold_lit`; dashed blue line). The lower the PWLD values, the more similar the items are. As shown in the figure, both lines have a clear left-to-right downward pattern confirming the intuitively expected relation between the cross-lingual formal similarity and intelligibility captured by the distribution of translation solutions. That is, when stimulus items have smaller distances to gold translations (and between gold and literal translations), the participants are more likely to return a higher proportion of acceptable translation solutions (correct, paraphrase or literal) and there are fewer fantasy, noise and empty responses.

The difference in slopes of the two lines can be interpreted as reflecting the properties of the automatically generated literal translations. GPT-4 generated literal translations that were closer (lower PWLD) to the gold translations for Ukrainian than for Belarusian. The analysis of distances between stimulus MSUs and literal translations for these languages shows that GPT-4 variants in Russian for Ukrainian items were more distant from the stimulus than the Russian translations for Belarusian items. This might reflect the relations between East-Slavic languages, where for the Ukrainian items it was difficult to find more literal Russian variants than gold translations.

To further explore the literal translation as an intercomprehension strategy, we used two approaches to identify stimulus MSUs that might be more suitable for literal cross-comprehension strategy: (a) items with small PWLD between stimulus and gold translation, and (b) items, where GPT-4 returned translations identical to the professional gold translations. Figure 2 shows which types of translations were offered for the Slavic MSUs extracted by each sampling method. The complementary line plots show the average intelligibility scores and the stimulus-to-gold PWLD values across each MSU sample. The sample in Figure 2a is based on the top 33% of original MSUs (the cut-off is selected arbitrary) that

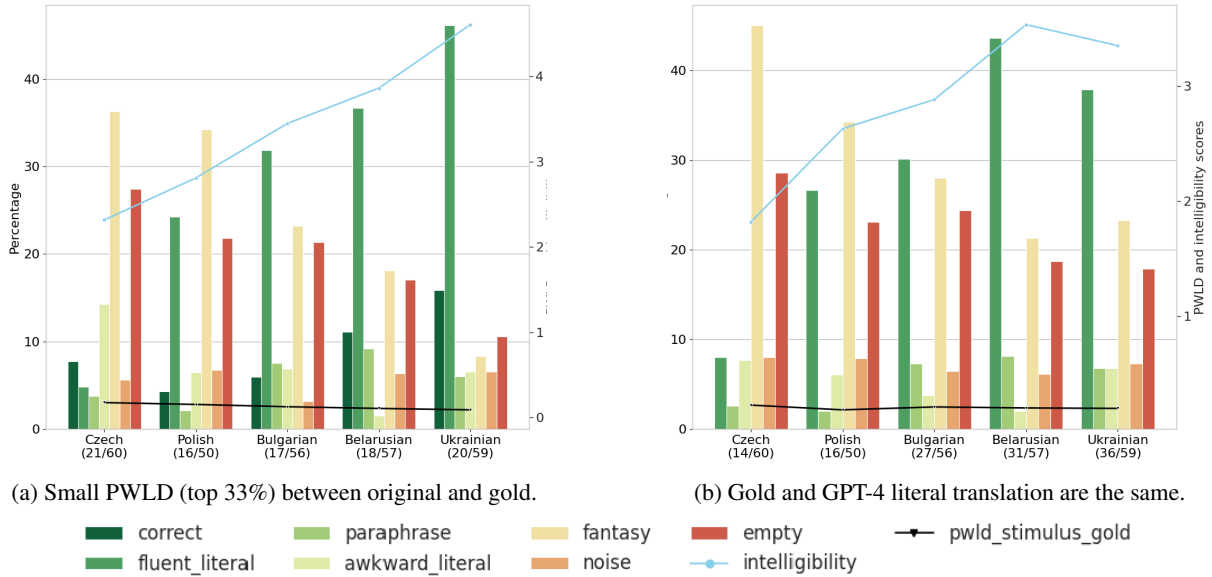


Figure 2: Two approaches to define suitable conditions for literal translation. Translation solution bars and mean intelligibility scores across the stimulus MSUs in each sample. Flat black line indicates that stimulus-to-gold PWLD is about the same level across languages on average. Brackets have the number of sampled stimuli to their total for each sampling method.

have the smallest distance to the gold translations. For these items, the intercomprehension pattern is clear: MSUs with the same cross-linguistic distance are more successfully processed in East-Slavic than in West-Slavic languages. Although the Czech data in our experiment offered as many opportunities (21 MSUs) for literal comprehension as Ukrainian (20 MSUs), the participants failed to recognise these similarities. We can hypothesise that the Latin script can introduce some of the confusion. In Figure 2b, the literal-translation-friendly sample includes MSUs, for which GPT-4 returned the same Russian variants as used in gold translations. This plot highlights the differences between Belarusian and Ukrainian as processed by GPT-4 and by the participants (compare lighter-green bars of fluent_literal translations for these languages). The participants did not see the fluent Russian correspondences for Ukrainian items picked by GPT-4 and returned fewer fluent translations and more mistranslations (light-yellow phantasy bars) than for Belarusian in this sample. For other languages, the distance between gold and literal established by GPT-4 was proportional to the participants' success in the translation task.

Figure 3 shows the distribution of intelligibility scores for each source language. The mean score across all MSUs (red diamonds) increases

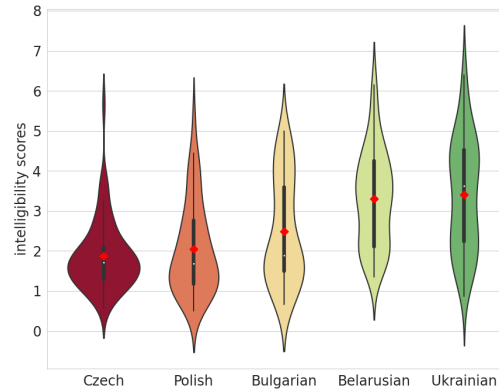


Figure 3: Distributions of the intelligibility scores. Red diamonds are means; the dark stripes with a white dot inside violins represent 25th, 50th, and 75th percentiles.

from left to right (from Czech to Ukrainian), which confirms previous findings and is intuitively expected. The scores are more homogeneous and centred around the low mean value for the less cross-intelligible West-Slavic languages, especially Czech. The distribution of intelligibility scores for the Ukrainian MSUs is more spread, with a bimodal tendency. It suggests that some Ukrainian MSUs are very intelligible, while others trigger intercomprehension difficulties and mis-

language	Pearson	MAE	nobs
Czech	0.21±0.43	0.63±0.21	60
Polish	0.23±0.50	0.90±0.30	50
Bulgarian	0.50±0.35	0.85±0.26	56
Belarusian	0.34±0.53	0.87±0.16	57
Ukrainian	0.62±0.31	0.98±0.34	59

Table 2: Regression results on intelligibility score for the top five language-specific predictors.

language	Pearson	MAE	nobs
Czech	0.23±0.36	0.40±0.10	60
Polish	0.19±0.55	0.51±0.17	50
Bulgarian	0.32±0.38	0.58±0.13	56
Belarusian	0.36±0.51	0.52±0.20	57
Ukrainian	0.65±0.37	0.52±0.21	59

Table 3: Regression results on entropy for the top five language-specific predictors.

translations.

4.2 Predicting Intelligibility via SVR

Table 2 shows correlations using the five features which returned the highest results for each language described. The intelligibility scores for the MSUs in the Cyrillic-based South- and East-Slavic languages are not only consistently higher than in the West-Slavic languages (see Figure 3) but also more predictable. Bulgarian and Ukrainian have the Pearson correlation coefficients 0.50 and 0.62, while the values of Pearson r for Polish and Czech do not exceed 0.23. For Belarusian (as well as for Ukrainian and Bulgarian) adding more features (up to a certain level) yield higher results. However, for West-Slavic languages the performance is unstable, and new features often introduce noise. The correlations on all features are considerably lower, especially for West-Slavic languages.

The regression results on MSU translation entropy as the learning target are 2% higher for Czech, Belarusian and Ukrainian but much lower for Polish and Bulgarian (see Table 3).

The variation in performance on the two variables describing the participants’ translation choices (intelligibility scores and MSU translation entropy) is due to the lack of consistency in their relations across the Slavic languages. The Pearson correlation coefficient (r) between the entropy of translation variants and intelligibility score ranges

from -0.799 (Ukrainian) to -0.325 (Czech) at $p < 0.05$. Figure 4 shows the regression lines fitted for each language separately and in combination. The entropy values are on average higher for Czech and Polish (2.70 and 2.47) than for Belarusian and Ukrainian (2.19 and 2.12) but for Czech and Polish they are less associated with intelligibility judging by the slopes and univariate r . It means that the participants’ responses were less more varied across functionally similar MSUs in the West-Slavic languages than in Ukrainian.

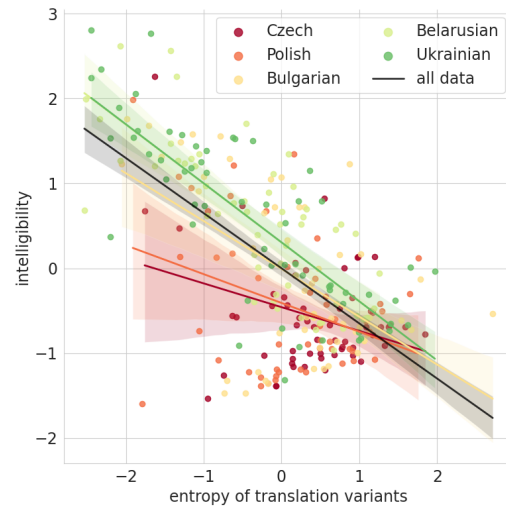


Figure 4: Relation between entropy of translation variants and intelligibility score for Slavic MSUs.

Low entropy scores characterise cases where the participants returned only a few unique responses and the probability distribution of these responses is skewed towards one type of translation solution. In other words, participants largely agreed on a Russian rendition for a given MSU.

- (1) For example, a Ukrainian conjunction *чим більше* (*the more*; *чем больше*) is formally very similar to the gold Russian variant (PWLD=0.085) and has a low entropy of 0.569 based on the three types of solutions: correct (*чем больше*), fantasy (*больше*) and *empty*. The probability of the first variant is 0.9, and the intelligibility has a maximum value of 6.4 across all MSUs.

For East-Slavic languages, this consensus often meant successful task completion, i.e., high intelligibility. The ratio of MSUs with the lower-than-average entropy and higher-than-average in-

telligibility was 36.8% and 42.4% for Belarusian and Ukrainian, respectively. For West-Slavic languages it does not exceed 24%. However, low entropy can also signal lack of comprehension for West-Slavic languages: in another 24% of cases (more for Czech) lower-than-average entropy was linked to lower-than-average intelligibility.

- (2) The Czech discourse marker *Ize rici* (*it can be said*, можно сказать) had a low entropy ($D=1.849$) and below average intelligibility of 0.941. Despite the formal distance for this MSU was below the Czech average (0.260 vs. 0.287), only one response was correct, and 65% of participants did not come up with any solution within the given time.

The next most important predictor of a different type is the formal distance between original MSUs and their gold translations (*pwld_stim_gold*). It is reasonable to expect that smaller original-to-gold PWLD would be negatively correlated with intelligibility. While this general trend is observed in our data (see row 1 in Table 4b in Appendix A), it is less expressed for West-Slavic languages. Figure 2a shows that the same level of PWLD results in lower intelligibility for them. Formal similarities between West-Slavic languages and Russian are more often false friends or prompt awkward solutions. Hence, PWLD is a less reliable predictor for intelligibility of West-Slavic MSUs.

- (3) The Czech particle *nejen ze* (*not only from*, не то что) has a low PWLD=0.161 (Czech average 0.286) and relatively low intelligibility (1.083 vs. average 1.872). For this item participants returned a variety of false literal solutions (e.g. неужели, нужен ли, не один же).

Another factor that correlates with the intelligibility of MSUs both within and across Slavic languages is the context sentence complexity. This property of the translation task is captured by the surprisal of the source or translated sentence (*surprisal_stim_sent* and *surprisal_gold_sent*). These features do not return significant correlations with intelligibility in univariate analysis for all languages but they are seen among the most informative features (except Belarusian).

Other features either are not consistently selected among the strong predictors and/or do not

demonstrate a significant correlation with intelligibility in univariate analysis.

Thus, the analysis of the combinations of strong predictors (Table 5, Appendix A) and the correlation analysis outcomes suggest the following conditions for the intelligibility of Slavic MSUs for Russian speakers. We have seen that the most important role is played by the participants' perception of the similarities, their ability to recognise and interpret them, captured by the entropy of translation variants. Then, the scale of these similarities between the languages matters. It is reflected by the point-wise PWLD distance between original MSU and its gold translation. Finally, average context sentence surprisal in either source or target language is an important intelligibility factor for all languages. Although West-Slavic and East-Slavic languages demonstrate some group similarities, each language seems to have a unique set of MSU intelligibility conditions. For Ukrainian, for example, the stimulus-to-gold PWLD is strongly positively correlated with entropy ($r = 0.555$) and with the number of participants' variants ($r = 0.636$). That is, the smaller the PWLD, the fewer variants are generated by the participants, the lower the entropy of translation variants and the higher the intelligibility of original MSU. This pattern is not seen in any other Slavic language so clearly.

5 Conclusion

This study explored the intelligibility of microsyntactic units (MSUs) in Slavic languages. We conducted a free translation experiment where Russian-speaking participants were asked to translate MSUs from Czech, Polish, Bulgarian, Belarusian, and Ukrainian into Russian. The aim of the study was to measure intercomprehension levels manifested in participants' responses and to explore the factors related to intelligibility between similar languages.

As expected, the MSUs in East-Slavic languages (Belarusian and Ukrainian) were most intelligible, followed by the South-Slavic Bulgarian. West-Slavic languages (Czech, Polish) presented a greater challenge for our participants. We demonstrated that the level of intercomprehension was related to the ability of the participants to identify and interpret the cross-lingual similarities. Generally, fewer translation variants for an original MSU indicated higher intelligibility.

Lower phonological distance between the MSUs in the source and target languages was another well-correlated and typical predictor of intercomprehension. Intra-linguistically, MSUs that were offered in easier contexts returned higher intelligibility scores.

6 Limitations

The data is limited to one direction of intercomprehension. Our approach is highly contingent on how the formal distance between original and gold items is calculated and what is accepted as a literal translation from the Slavic languages into Russian. The context sentences were not controlled for complexity or topic across stimulus languages. The phonological distance calculations rely heavily on automated grapheme-to-phoneme conversion.

Acknowledgments

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 232722074 – SFB 1102 and by Saarland University (UdS-Internationalisierungsfonds).

References

- Badr Abdullah, Marius Mosbach, Iuliia Zaitova, Bernd Möbius, and Dietrich Klakow. 2021. Do Acoustic Word Embeddings Capture Phonological Similarity? An Empirical Study. In *Proceedings of Interspeech 2021*, pages 4194–4198.
- Tania Avgustinova and Leonid Iomdin. 2019. Towards a typology of microsyntactic constructions. In *Computational and Corpus-Based Phraseology: Third International Conference, Europhras 2019, Malaga, Spain, September 25–27, 2019, Proceedings 3*. Springer, pages 15–30.
- Michael Carl and Moritz Jonas Schaeffer. 2017. Why translation is difficult: A corpus-based study of non-literality in post-editing and from-scratch translation. *Hermes (Denmark)* 56:43–57.
- Lionel Fontan, Isabelle Ferrané, Jérôme Farinas, Julien Pinquier, and Xavier Aumont. 2016. Using phonologically weighted levenshtein distances for the prediction of microscopic intelligibility. In *Annual conference Interspeech (INTERSPEECH 2016)*, page 650.
- Charlotte Gooskens and Femke Swarte. 2017. Linguistic and extra-linguistic predictors of mutual intelligibility between germanic languages. *Nordic Journal of Linguistics* 40:123–147.
- Klara Jagrova, Tania Avgustinova, Irina Stenger, and Andrea Fischer. 2018. Language models, surprisal and fantasy in slavic intercomprehension. *Computer Speech & Language* 53.
- Machteld Meullemans and Alice Fiorentino. 2018. What is intercomprehension and what is it good for? In François Grin, Manuel Célio Conceição, Peter A. Kraus, László Marác, Žaneta Ozoliņa, Nike K. Pokorn, and Anthony Pym, editors, *The MIME vademecum: Mobility and inclusion in multilingual Europe*, Artgraphic Cavin SA, pages 146–147. Hal-02497697.
- Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Hybrid), pages 578–585.
- Irina Stenger, Klára Jágrová, Andrea Fischer, Tania Avgustinova, Dietrich Klakow, and Roland Marti. 2017. Modeling the impact of orthographic coding on czech-polish and bulgarian-russian reading intercomprehension. *Nordic Journal of Linguistics* 40(2):175–199.
- Peter Trudgill. 2003. *Mutual intelligibility*, Edinburgh University Press, Edinburgh, page 91.
- Yuxiang Wei. 2022. Entropy as a measurement of cognitive load in translation. In *AMTA 2022 - 15th Conference of the Association for Machine Translation in the Americas, Proceedings - Workshop on Empirical Translation Process Research*, volume 1, pages 75–86.
- Iuliia Zaitova, Irina Stenger, Muhammad Umer Butt, and Tania Avgustinova. 2024a. Cross-linguistic processing of non-compositional expressions in slavic languages. In *Proceedings of the Workshop on Cognitive Aspects of the Lexicon@ LREC-COLING 2024*, pages 86–97.
- Iuliia Zaitova, Irina Stenger, Wei Xue, Tania Avgustinova, Bernd Möbius, and Dietrich Klakow. 2024b. Cross-linguistic intelligibility of non-compositional expressions in spoken context. In *Proc. Interspeech 2024*, pages 4189–4193.
- Jian Zhu, Cong Zhang, and David Jurgens. 2022. Byt5 model for massively multilingual grapheme-to-phoneme conversion. In *Annual conference Interspeech (INTERSPEECH 2022)*.
- Dmitry Zmitrovich, Aleksandr Abramov, Andrey Kalmykov, Vitaly Kadulin, Maria Tikhonova, Ekaterina Taktasheva, Danil Astafurov, Mark Baushenko, Artem Snegirev, Tatiana Shavrina, et al. 2024. A Family of Pretrained Transformer Language Models for Russian. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 507–524.

Appendix A. List of predictors with correlation analysis outcome

Table 4: Association between translation task features and response variables. Asterisks indicate statistically significant results at the confidence level of 0.05. The tables are sorted to have the features with significant results across more languages on top. Features with the same number of significant results are sorted alphabetically.

(a) Pearson correlation coefficient between predictors and entropy of translation variants.

#	feature	Czech	Polish	Bulgarian	Belarusian	Ukrainian
1	pwld_stim_gold	0.314*	0.08	0.331*	0.333*	0.556*
2	surprisal_lit	-0.001	-0.088	0.043	0.385*	0.328*
3	pwld_stim_lit	0.144	0.042	0.238	0.3*	0.481*
4	cosine_stim_lit	0.067	0.02	-0.316*	-0.16	-0.327*
5	cosine_stim_gold	0.001	-0.094	-0.382*	-0.139	-0.377*
6	eval_lit	0.014	-0.046	-0.114	-0.284*	-0.099
7	surprisal_stim_sent	0.045	0.104	0.298*	0.258	0.076
8	surprisal_gold	0.123	-0.076	-0.08	0.177	0.393*
9	surprisal_stim	0.008	0.141	0.347*	0.101	0.18
10	surprisal_lit_sent	-0.197	-0.095	0.172	0.229	0.012
11	qe_lit	-0.126	-0.035	0.176	-0.107	-0.088
12	pwld_gold_lit	0.051	0.0	0.236	0.079	0.137
13	qe_gold	-0.091	-0.058	0.198	0.044	-0.008
14	surprisal_gold_sent	-0.066	-0.081	0.11	0.019	-0.019

(b) Pearson correlation coefficient between predictors and intelligibility scores.

#	feature	Czech	Polish	Bulgarian	Belarusian	Ukrainian
1	pwld_stim_gold	-0.305*	-0.375*	-0.432*	-0.384*	-0.594*
2	pwld_stim_lit	-0.304*	-0.29*	-0.469*	-0.211	-0.418*
3	cosine_stim_gold	0.008	0.233	0.438*	0.272*	0.438*
4	surprisal_stim_sent	-0.263*	-0.366*	-0.258	-0.426*	-0.083
5	eval_lit	0.16	0.276	0.268*	0.439*	-0.003
6	pwld_gold_lit	-0.153	-0.293*	-0.396*	-0.178	-0.099
7	surprisal_lit	-0.223	-0.09	-0.318*	-0.415*	-0.186
8	cosine_stim_lit	-0.099	0.129	0.224	0.213	0.388*
9	qe_lit	0.171	0.153	-0.069	0.304*	0.024
10	surprisal_gold	-0.039	-0.026	-0.004	-0.226	-0.308*
11	surprisal_lit_sent	-0.183	-0.135	-0.188	-0.37*	0.072
12	qe_gold	0.151	0.093	-0.097	0.111	-0.027
13	surprisal_gold_sent	-0.072	-0.001	-0.032	-0.137	0.034
14	surprisal_stim	0.2	-0.174	-0.236	-0.251	-0.187

Table 5: Language-specific selections of best intelligibility predictors (by RFE, N=5)

	feature names
Czech	surprisal_lit, surprisal_gold, surprisal_stim_sent, pwld_stim_lit, qe_gold
Polish	surprisal_stim_sent, surprisal_lit_sent, surprisal_gold_sent, cosine_stim_gold, pwld_stim_gold
Bulgarian	surprisal_stim, surprisal_lit, cosine_stim_gold, pwld_stim_lit, pwld_stim_gold
Belarusian	surprisal_stim, surprisal_gold, surprisal_lit_sent, surprisal_gold_sent, pwld_stim_gold
Ukrainian	surprisal_lit_sent, surprisal_gold_sent, pwld_stim_gold, qe_gold, qe_lit

Train More Parameters But Mind Their Placement: Insights into Language Adaptation with PEFT

Jenny Kunz

Dept. of Computer and Information Science

Linköping University

jenny.kunz@liu.se

Abstract

Smaller LLMs still face significant challenges even in medium-resourced languages, particularly when it comes to language-specific knowledge – a problem not easily resolved with machine-translated data. In this case study on Icelandic, we aim to enhance the generation performance of an LLM by specialising it using unstructured text corpora. A key focus is on preventing interference with the models’ capabilities of handling longer context during this adaptation. Through ablation studies using various parameter-efficient fine-tuning (PEFT) methods and setups, we find that increasing the number of trainable parameters leads to better and more robust language adaptation. LoRAs placed in the feed-forward layers and bottleneck adapters show promising results with sufficient parameters, while prefix tuning and (IA)³ are not suitable. Although improvements are consistent in 0-shot summarisation, some adapted models struggle with longer context lengths, an issue that can be mitigated by adapting only the final layers.

1 Introduction

LLMs have strong multilingual capabilities and top the leaderboards even for less-represented languages (Nielsen et al., 2024). However, smaller LLMs still struggle with these languages, hampering fast and resource-efficient inference. Instruction tuning on machine-translated data can improve performance compared to English-only tuning (Muennighoff et al., 2023; Chen et al., 2024a) but models still fall short when evaluated on native benchmarks, likely due to missing language-specific knowledge (Chen et al., 2024b). While collecting large amounts of native instruction-tuning

data could address this issue, this can be costly or infeasible. This makes techniques for adapting models using unstructured text data valuable.

In this paper, we perform ablations with parameter-efficient fine-tuning (PEFT) methods for language adaptation with unstructured text data *after* instruction alignment. This diverges from the standard setup for fine-tuning a model: Unlike typical fine-tuning, where the adaptation data closely matches the expected output format, the data we use is closer to the expected output in language but likely further from the target task format. Therefore, the setup risks interference with the original instruction-tuning objectives, possibly leading to *catastrophic forgetting* (McCloskey and Cohen, 1989). In addition, hardware constraints made us choose a maximum context length smaller than the one used in pre-training, risking further performance degradation.

Therefore, we aim to identify setups that do not interfere with previously learned abilities. We attempt to avoid catastrophic forgetting with PEFT methods that leave the majority of or all model parameters unchanged: LoRA (Hu et al., 2022), IA³ (Liu et al., 2022), bottleneck adapters (Houlsby et al., 2019) and prefix tuning (Li and Liang, 2021). We experiment with the number of learnable parameters, the placement of LoRA matrices in different Transformer modules and layers, as well as the training corpus used for adaptation.

We use the smallest instruction-tuned LLaMA 3.2 model (LlamaTeam, 2024) with 1B parameters and adapt it to Icelandic, evaluating performance on text summarisation. Our findings are that:

- LoRA and bottleneck adapters show improvements especially in 0-shot settings, though simply adding target-language task demonstrations also improves the performance substantially.
- A higher number of trainable parameters is better.
- LoRAs in the feed-forward layers are the best-performing setup, followed by bottleneck

adapters. LoRA in the attention layers works less well, particularly considering the number of trainable parameters. We therefore conclude that feed-forward modules are the most promising target in language adaptation.

- Prefix tuning hurts the model’s capabilities.
- Some setups with few trainable parameters negatively impact 5-shot performance, possibly due to smaller context lengths at adaptation time compared to pre-training time. This can be resolved by restricting adapter placement to the top layers.

2 Experimental Setup

2.1 Models

We use *Llama-3.2-1B-Instruct*, the newest and smallest Llama model at the time of writing, with 1B, 16 layers, and a hidden size of 2048. This model has been tuned with instruction fine-tuning (Wei et al., 2022) and reinforcement learning with human feedback (Ouyang et al., 2024)¹.

2.2 Adaptation Data

Our main dataset for adaptation is the Icelandic portion of CC100 (Conneau et al., 2020) that has been processed with CCNet filtering (Wenzek et al., 2020) to increase data quality. We randomly select 250,000 text chunks, with a maximum length of 1,024 tokens, resulting in 12.5M tokens. This data was likely seen during pre-training, i.e., the model is not exposed to new data but *primed* towards Icelandic. As web-crawled corpora are reportedly of lower quality for smaller languages (Kreutzer et al., 2022; Artetxe et al., 2022), we perform ablations with the curated Icelandic Gigaword Corpus (IGC) (Steingrímsson et al., 2018; Barkarson et al., 2022), using sections from its subsets *Books*, *Wiki*, *Social*, and *Journals*. Even here we use 250,000 chunks, resulting in 12M tokens. As the *Social* subset is by far the largest and we aim to have a large portion of highly curated text, we undersample it by using only 10%, resulting in a dataset composition of 9% *Books*, 17% *Wiki*, 22% *Journals*, and 52% *Social*.

2.3 Adaptation Methods and Setups

The code, prompt generator and adapters used for the experiments in this paper can be found at github.com/jekunz/peft-la. We use the Transformers (Wolf et al., 2020) and Adapters (Poth et al., 2023) libraries, a learning rate of 5e-5,

¹Ablations with the base model *Llama-3.2-1B* showed inferior performance with and without adaptation.

a linear learning rate scheduler, and a batch size of 4.² All adapters are trained with a causal language modeling objective. We test the following methods and setups:

LoRA is a widespread adaptation technique for generative LLMs. In the most common setup, it adds low-rank decomposition matrices to the model’s self-attention modules and trains only those. The matrices can be merged into the weights, removing the inference overhead. For LoRA in the attention module, we test ranks 1024, 256, 128, 32 and 8 and apply LoRA to the query and value matrices, which is reportedly the most stable setup (Fomenko et al., 2024). We also test LoRA in the feed-forward module and place LoRAs in all matrices using ranks 256, 128, 64, 32 and 8. For both module setups and all ranks, we use $\alpha = 2r$.

IA³ is the most parameter-efficient among the methods tested. It multiplies activations in the model’s attention (key and value) and feed-forward matrices with learned vectors, adding hardly any overhead.

Bottleneck adapters add smaller intermediate layers with a down- and up-projection in between the model’s layers. While popular for encoder model, bottleneck adapters are less common for generative LLMs as they increase the number of parameters and depth even during inference. We train Housby adapters with reduction factors of 64, 16 and 4.

Prefix tuning prepends a sequence of learnable prefix vectors to the input sequences, allowing the model to attend to the prefix vectors when generating the subsequent tokens. As the vectors add to the sequence length, even prefix tuning slows down inference. We use a prefix length of 30 tokens.

2.4 Evaluation

To assess generative performance, we evaluate abstractive text summarisation with the RÚV Radio News (RRN) dataset (Sverrisson and Einarsson, 2023) in the *main* \rightarrow *intro* setup, i.e., generating the introduction from the main body of the article. We filter out articles missing one of these fields.

We evaluate the summaries using BERTScore (Zhang et al., 2020) (base model: *bert-base-multilingual-uncased*) to measure the representational similarity between the output and the reference, and ROUGE-L (Lin, 2004) for surface overlap, based on the longest common subsequence.

²As the learning rate and scheduler are crucial in continued pre-training (Ibrahim et al., 2024), we also tested 1e-5 and 1e-4 and a cosine scheduler but did not observe large differences.

The models are evaluated in 0-shot, 1-shot and 5-shot setups with minimal prompts in Icelandic³ that instruct the model to summarise the article in one paragraph and include markers for the start of both the article and the summary.

3 Results and Discussion

3.1 PEFT Methods

	0-shot	1-shot	5-shot
No Adapter	53.37 / 04.09	64.68 / 10.26	64.01 / 11.37
LoRA-qv-1024	63.61 / 08.57	66.53 / 11.70	65.50 / 12.06
LoRA-qv-256	63.27 / 08.32	65.55 / 11.05	62.97 / 10.56
LoRA-qv-128	62.55 / 07.63	64.51 / 10.78	62.23 / 10.54
LoRA-qv-32	61.06 / 06.62	62.68 / 08.98	55.42 / 05.60
LoRA-qv-8	60.45 / 05.21	61.53 / 08.23	56.62 / 06.42
LoRA-ff-256	65.60 / 09.72	69.06 / 13.89	69.10 / 15.48
LoRA-ff-128	64.67 / 08.87	69.10 / 13.86	68.36 / 14.55
LoRA-ff-64	63.72 / 07.86	67.72 / 12.60	67.46 / 13.65
LoRA-ff-32	62.94 / 07.19	67.61 / 12.18	67.42 / 13.76
LoRA-ff-8	61.69 / 06.36	64.85 / 10.39	62.66 / 10.09
(IA) ³	56.70 / 04.56	64.07 / 09.47	61.74 / 10.37
Bottlen.-4	63.78 / 08.15	66.75 / 11.74	66.74 / 13.21
Bottlen.-16	63.33 / 08.38	67.77 / 13.11	65.80 / 12.36
Bottlen.-64	60.66 / 05.16	64.79 / 09.96	61.32 / 08.59
Prefix	55.84 / 02.02	54.56 / 01.73	49.86 / 00.67

Table 1: Comparing adaptation methods. BERTScore F1 / ROUGE-L.

As shown in Table 1, language adaptation consistently improves 0-shot summarisation scores. However, for 1-shot and 5-shot setups, the results are more mixed, and in some setups decrease compared to the baseline without adaptation. That the 1-shot setup without adaptation already shows comparable performance to many adaptation setups implies that in-context learning, where possible, can be an alternative to language adaptation for this model.

The best-performing method are LoRAs in the feed-forward layers. Even bottleneck adapters with a reduction factor of 16 or 4 consistently increase scores, although there is a noticeable difference in performance to feed-forward LoRA. As illustrated in Figure 1, feed-forward LoRA also results in the highest BERTScores relative to the number of parameters added, followed by bottleneck adapters. LoRA in the attention matrices requires substantially more parameters to reach a comparable performance. These results show that the placement of the PEFT modules in the Transformer architecture

³We also tested English instructions, which led to slightly worse results, except for the *no adapters* model, where English instruction slightly improved the 0-shot performance.

plays a crucial role even if the number of trainable parameters is the same.

Some setups interfere with the model’s ability to operate on longer inputs as the performance especially in the 5-shot setup decreases. We hypothesise this is a result of limiting the context length to 1,024 tokens during the adaptation process. LoRA in the attention module is the most heavily affected setup, suggesting that the effectiveness of self-attention when processing longer contexts is harmed.

We observe that performance improves as the LoRA rank increases or the bottleneck reduction factor decreases, indicating that sufficient learning capacity is necessary for better results in language adaptation. This is in line with the underwhelming performance of (IA)³, which introduces the fewest parameters. Designed as an alternative to in-context learning for task adaptation, (IA)³ does not transfer well to language adaptation.

Prefix tuning with textual data decreases the performance substantially for the 1- and 5-shot setups. We assume that as prefixes have a direct impact on the generation, prefixes that diverge from the expected output format harm the model’s abilities to match the latter. For this reason, prefix-tuning an instruction-tuned model on unlabelled text does not work, whereas prefix-tuning on specific tasks like summarisation, or instruction tuning in general, works well as shown by Zhang et al. (2024a).

3.2 Ablation 1: LoRA Modules

	0-shot	1-shot	5-shot
q,v	63.27 / 08.32	65.55 / 11.05	62.97 / 10.56
ff	65.60 / 09.72	69.06 / 13.89	69.10 / 15.48
ff + q,v	65.44 / 09.61	68.44 / 13.14	68.89 / 15.17

Table 2: Comparing LoRA module placement. BERTScore F1 / ROUGE-L; LoRA rank 256

We have a closer look at the module placement of LoRAs and compare LoRA in the self-attention module, LoRA in the feed-forward module, and LoRA both in the self-attention and the feed-forward module.

In the results given in Table 2, we see that for the same rank, LoRA in the feed-forward module is better than in the attention module. Moreover, it is slightly better than LoRA in both the attention and the feed-forward modules. We find this surprising given that the latter option has the most trainable parameters and conclude that having LoRA even

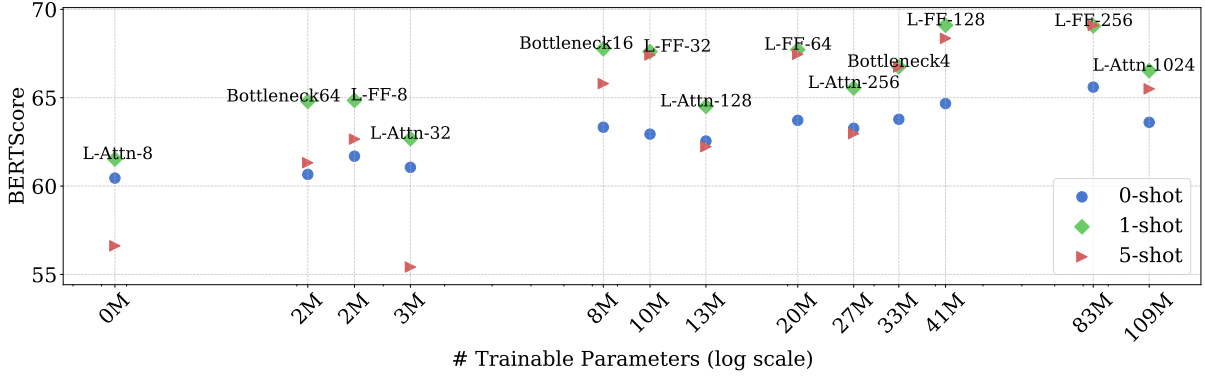


Figure 1: Number of trainable parameters plotted against BERTScores. Prefix tuning (34M parameters) and (IA)³ (49K parameters) are excluded.

in the attention matrices is at best unnecessary.

3.3 Ablation 2: Layer Exclusion

	0-shot	1-shot	5-shot
No Adapter	53.37 / 04.09	64.68 / 10.26	64.01 / 11.37
All Layers	61.06 / 06.62	62.68 / 08.98	55.42 / 05.60
All but last 2	59.39 / 04.83	60.26 / 07.37	57.73 / 06.51
All but last 4	59.89 / 04.93	62.37 / 08.70	58.29 / 07.02
Only last 2	59.64 / 03.64	63.55 / 08.37	65.40 / 12.42
Only last 4	58.78 / 04.56	62.20 / 08.22	61.94 / 10.57

Table 3: Layer Exclusion experiments. BERTScore F1 / ROUGE-L; Self-attention (qv) LoRA rank 32.

Fine-tuning primarily affects the final layers of a model (Merchant et al., 2020; Mosbach et al., 2020; Zhou and Srikumar, 2022). We explore two strategies focusing on these layers: (1) *excluding* the final layers during adaptation to preserve the instruction-tuning capabilities while focusing on general language learning, which is likely stored in earlier layers, and (2) adapting *only* the final layers, as this may be sufficient and could maintain the model’s robustness with respect to the limited context length used in our adaptation process (a key issue highlighted in Section 3.1).

We test the two hypotheses using self-attention LoRA with rank 32 as this configuration shows strong 0-shot performance but suffers in the 5-shot setup. The results in Table 3 show that the first hypothesis does not hold; excluding the last layers does not improve the performance and, in some cases, degrades it. The second hypothesis, however, appears plausible: restricting LoRA modules to the last two layers yields the best 5-shot results among all setups in Table 3, outperforming the baseline without adaptation. However, this comes at the

expense of a slight decrease in 0-shot performance. We are hopeful that these insights can guide us in developing customised methods for language adaptation.

3.4 Ablation 3: Training Corpora

	0-shot	1-shot	5-shot
CCNet	63.27 / 08.32	65.55 / 11.05	62.97 / 10.56
IGC	60.80 / 05.75	61.02 / 06.48	58.31 / 06.17
CCNet	65.60 / 09.72	69.06 / 13.89	69.10 / 15.48
IGC	63.66 / 08.10	66.19 / 10.46	66.37 / 12.00
CCNet	63.78 / 08.15	66.75 / 11.74	66.74 / 13.21
IGC	61.39 / 05.58	64.95 / 09.79	65.24 / 11.54

Table 4: Comparing text corpora for adaptation. BERTScore F1 / ROUGE-L; LoRA-qv-256 (above), LoRA-ff-256 (middle) and bottleneck reduction factor 4 (below).

In Table 4, we do not observe a benefit of training on the IGC; on the contrary, the performance is consistently lower. While this is in line with previous research (Artetxe et al., 2022; van Noord et al., 2024), note that we do not test on any task where high-quality generation is important but on text summarisation, which can rely on copying chunks of text. We also note that CCNet is probably more diverse, and that different mixes from the IGC may lead to different results. We therefore believe that it is worthwhile to continue testing on curated data.

3.5 Future Work

In order to test whether our findings generalise, we plan to extend our approach to other languages, larger models and adapters trained on more data, and to explore the effect of training on longer con-

texts. Based on our experiments on the placement and training of adapters in Section 3.3, we hope to find a sweet spot for language adaptation where no relevant information is overwritten but generation performance is improved. Inspiration could be taken from methods that automatically detect, and assign more parameters to, layers of particular importance (Zhang et al., 2023; Yao et al., 2024).

A common approach to mitigate interference is episodic memories – mixing in examples from previous tasks (Chaudhry et al., 2019), in our case, instruction-tuning data. This has shown promise in other works (Jiang et al., 2024; Parmar et al., 2024), making it worthwhile to incorporate.

One challenge in evaluating language adaptation methods is that automatic metrics for generative performance provide limited and potentially misleading insights. While running extensive human evaluations for all ablations in this paper is impractical, a human study of model outputs for the most promising setups, across a diverse set of prompts, should be included in future evaluations.

4 Related Work

Razumovskaia et al. (2024) find that LoRA language adaptation with unstructured text data improves the linguistic quality of generated texts in human ratings but usefulness and performance on a (translated) natural language inference benchmark remain low. Their study indicates that benchmark evaluation could underestimate the usefulness of language adaptation in chat and generation setups.

Work on testing other PEFT architectures than LoRA for language adaptation of LLMs has been sparse. While bottleneck-style language adapters trained on text corpora are a common setup for cross-lingual transfer with encoder models (Pfeiffer et al., 2020; He et al., 2021; Faisal and Anastasopoulos, 2022), they have been largely overlooked for generative models, likely due to the inference overhead that can be avoided with LoRA, as the latter works equally well for task fine-tuning. Our experiments show that similar findings hold for language adapters: Bottleneck adapters perform well but there are LoRA setups that reach the same performance or are better while avoiding the overhead.

Recent language adaptation works have focused on target-language instruction fine-tuning, often with machine-translated data (Muennighoff et al., 2023; Chen et al., 2024a; Holmström and Doostmohammadi, 2023). In cross-lingual transfer, mul-

tilingual instruction tuning has shown promise, particularly for generative tasks (Kew et al., 2023) and for larger models (Chen et al., 2024a). However, models trained on machine-translated data may perform well on translated evaluation sets but struggle on native benchmarks (Chen et al., 2024b).

5 Conclusion

We tested a range of PEFT methods for language adaptation using unstructured text corpora, finding that LoRA in the feed-forward modules yielded the most promising results, followed by bottleneck adapters. LoRA in the attention modules performed less well, was less robust to larger context lengths and needed more parameters for a comparable performance. Combining LoRAs in both the attention and feed-forward modules did not improve over feed-forward LoRAs only, and may even lead to slightly decreased performance. Prefix tuning and (IA)³ were not suitable at all.

Our results show that across architectures, more trainable parameters lead to better scores, showing, perhaps unsurprisingly, that sufficient learning capacity is crucial for language adaptation.

Some adaptation setups led to a decline in performance as contexts get longer; possibly a result of restricted context lengths during adaptation. However, this issue can be mitigated by training only the last layers. Notably, we did not observe any positive effects from using higher-quality pre-training data sourced from narrower domains.

Moving forward, with a higher resource investment, we see the potential that more training data, possibly with instruction data in the mix, and longer context lengths improve the performance further. However, to truly assess the potential of these methods, we need more diverse, language-native evaluation data, as well as fine-grained human evaluations that assess various aspects of generated language quality and content.

Limitations

The meaningfulness of automated text summarisation metrics when using news text summaries as references has been questioned and is highly dependent on the dataset (Zhang et al., 2024b). While our search for effective setups yielded conclusive results with BERTScore and ROUGE-L, moving forward, it will be crucial to incorporate human evaluations and more diverse tasks to accurately

assess performance across a broader and better-interpretable range of criteria.

As we have discussed in Section 5, we see a critical need for more language-native evaluation data, in particular datasets that incorporate significant language-specific knowledge (Chen et al., 2024b). Testing on a limited set of language-native tasks most of which are classification tasks, or on machine-translated data, may give a limited picture of the effect of language adaptation.

Due to computational constraints, we were unable to include larger models or more than one language in this study. As a result, it remains unclear whether our findings apply to other languages, especially those that are typologically more different from or closer to English.

Acknowledgments

I thank my colleagues Kevin Glocker, Kättriin Kukk, Julian Schlenker, Marcel Bollmann, Noah-Manuel Michael and Romina Oji for valuable discussions at all stages of this project and feedback on earlier drafts, and the anonymous reviewers for their constructive feedback and insightful suggestions.

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. It was supported by TrustLLM funded by Horizon Europe GA 101135671. The computations were enabled by the Berzelius resource provided by the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre and by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

References

Mikel Artetxe, Itziar Aldabe, Rodrigo Agerri, Olatz Perez-de Viñaspre, and Aitor Soroa. 2022. Does corpus quality really matter for low-resource languages? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7383–7390, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Starkaður Barkarson, Steinþór Steingrímsson, and Hildur Hafsteinsdóttir. 2022. Evolving large text corpora: Four versions of the Icelandic Gigaword corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2371–2381, Marseille, France. European Language Resources Association.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K. Dokania, Philip H. S. Torr, and Marc’Aurelio Ranzato. 2019. On tiny episodic memories in continual learning.

Pinzhen Chen, Shaoxiong Ji, Nikolay Bogoychev, Andrey Kutuzov, Barry Haddow, and Kenneth Heafield. 2024a. Monolingual or multilingual instruction tuning: Which makes a better alpaca. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1347–1356, St. Julian’s, Malta. Association for Computational Linguistics.

Pinzhen Chen, Simon Yu, Zhicheng Guo, and Barry Haddow. 2024b. Is it good data for multilingual instruction tuning or just bad multilingual evaluation for large language models?

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.

Fahim Faisal and Antonios Anastasopoulos. 2022. Phylogeny-inspired adaptation of multilingual models to new languages. 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 434–452, Online only. Association for Computational Linguistics.

Vlad Fomenko, Han Yu, Jongho Lee, Stanley Hsieh, and Weizhu Chen. 2024. A note on lora.

Ruidan He, Linlin Liu, Hai Ye, Qingyu Tan, Bosheng Ding, Liying Cheng, Jiawei Low, Lidong Bing, and Luo Si. 2021. On the effectiveness of adapter-based tuning for pretrained language model 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 2208–2222, Online. Association for Computational Linguistics.

Oskar Holmström and Ehsan Doostmohammadi. 2023. Making instruction finetuning accessible to non-English languages: A case study on Swedish models. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 634–642, Tórshavn, Faroe Islands. University of Tartu Library.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference*

- on *Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.
- 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*.
- Adam Ibrahim, Benjamin Thérien, Kshitij Gupta, Mats Leon Richter, Quentin Gregory Anthony, Eugene Belilovsky, Timothée Lesort, and Irina Rish. 2024. Simple and scalable strategies to continually pre-train large language models. *Transactions on Machine Learning Research*.
- Zhengbao Jiang, Zhiqing Sun, Weijia Shi, Pedro Rodriguez, Chunting Zhou, Graham Neubig, Xi Lin, Wen-tau Yih, and Srinu Iyer. 2024. Instruction-tuned language models are better knowledge learners. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5421–5434, Bangkok, Thailand. Association for Computational Linguistics.
- Tannon Kew, Florian Schottmann, and Rico Sennrich. 2023. Turning english-centric llms into polyglots: How much multilinguality is needed?
- Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsara Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmongkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, Iroko Orife, Kelechi Ogueji, Andre Niyongabo Rubungo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. 2022. Quality at a Glance: An Audit of Web-Crawled Multilingual Datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.
- Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. 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 4582–4597, 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.
- Haokun Liu, Derek Tam, Muqeeth Mohammed, Jay Mohata, Tenghao Huang, Mohit Bansal, and Colin Raffel. 2022. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In *Advances in Neural Information Processing Systems*.
- LlamaTeam. 2024. The llama 3 herd of models.
- Michael McCloskey and Neil J. Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. *The Psychology of Learning and Motivation*, 24:104–169.
- Amil Merchant, Elahe Rahimtoroghi, Ellie Pavlick, and Ian Tenney. 2020. What happens to BERT embeddings during fine-tuning? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 33–44, Online. Association for Computational Linguistics.
- Marius Mosbach, Anna Khokhlova, Michael A. Hedderich, and Dietrich Klakow. 2020. On the interplay between fine-tuning and sentence-level probing for linguistic knowledge in pre-trained transformers. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 68–82, Online. Association for Computational Linguistics.
- Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, Toronto, Canada. Association for Computational Linguistics.
- Dan Saattrup Nielsen, Kenneth Enevoldsen, and Peter Schneider-Kamp. 2024. Encoder vs decoder: Comparative analysis of encoder and decoder language models on multilingual nlu tasks.
- Rik van Noord, Taja Kuzman, Peter Rupnik, Nikola Ljubešić, Miquel Esplà-Gomis, Gema Ramírez-Sánchez, and Antonio Toral. 2024. Do language models care about text quality? evaluating web-crawled corpora across 11 languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5221–5234, Torino, Italia. ELRA and ICCL.
- 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. 2024. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing*

- Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.
- Jupinder Parmar, Sanjev Satheesh, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Reuse, don't retrain: A recipe for continued pretraining of language models.
- Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.
- Clifton Poth, Hannah Sterz, Indraneil Paul, Sukannya Purkayastha, Leon Engländer, Timo Imhof, Ivan Vulić, Sebastian Ruder, Iryna Gurevych, and Jonas Pfeiffer. 2023. Adapters: A unified library for parameter-efficient and modular transfer learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 149–160, Singapore. Association for Computational Linguistics.
- Evgeniia Razumovskaia, Ivan Vulić, and Anna Korhonen. 2024. Analyzing and adapting large language models for few-shot multilingual nlu: Are we there yet?
- Steinþór Steingrímsson, Sigrún Helgadóttir, Eiríkur Rögnvaldsson, Starkaður Barkarson, and Jón Guðnason. 2018. Risamálheild: A very large Icelandic text corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Þór Sverrisson and Hafsteinn Einarsson. 2023. Abstractive text summarization for Icelandic. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 17–31, Tórshavn, Faroe Islands. University of Tartu Library.
- 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*.
- Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.
- 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.
- Kai Yao, Penlei Gao, Lichun Li, Yuan Zhao, Xiaofeng Wang, Wei Wang, and Jianke Zhu. 2024. Layer-wise importance matters: Less memory for better performance in parameter-efficient fine-tuning of large language models.
- Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. 2023. Adaptive budget allocation for parameter-efficient fine-tuning. In *The Eleventh International Conference on Learning Representations*.
- Renrui Zhang, Jiaming Han, Chris Liu, Aojun Zhou, Pan Lu, Yu Qiao, Hongsheng Li, and Peng Gao. 2024a. LLaMA-adapter: Efficient fine-tuning of large language models with zero-initialized attention. In *The Twelfth 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 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown, and Tatsunori B. Hashimoto. 2024b. Benchmarking large language models for news summarization. *Transactions of the Association for Computational Linguistics*, 12:39–57.
- Yichu Zhou and Vivek Srikumar. 2022. A closer look at how fine-tuning changes BERT. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1046–1061, Dublin, Ireland. Association for Computational Linguistics.

SweSAT-1.0: The Swedish University Entrance Exam as a Benchmark for Large Language Models

Murathan Kurfali¹ Shorouq Zahra¹ Evangelia Gogoulou¹
Luise Dürlich^{1,2} Fredrik Carlsson¹ Joakim Nivre^{1,2}

¹RISE Research Institutes of Sweden, Sweden

²Uppsala University, Sweden

Abstract

This introduces SweSAT-1.0, a new benchmark dataset created from the Swedish university entrance exam (Högskoleprovet) to assess large language models in Swedish. The current version of the benchmark includes 867 questions across six different tasks, including reading comprehension, mathematical problem solving, and logical reasoning. We find that some widely used open-source and commercial models excel in verbal tasks, but we also see that all models, even the commercial ones, struggle with reasoning tasks in Swedish. We hope that SweSAT-1.0 will facilitate research on large language models for Swedish by enriching the breadth of available tasks, offering a challenging evaluation benchmark that is free from any translation biases.

1 Introduction

The recent progress in language modeling has significantly expanded the generalization capabilities of large language models (LLMs). Models such as Llama 3.1 (Dubey et al., 2024), Gemma (Team et al., 2024), and GPT-4 (Achiam et al., 2023) have demonstrated remarkable performance across a wide range of NLP tasks, exceeding the expectations researchers held just a few years ago. Consequently, many existing benchmarks are found to be inadequate due to their task-specific nature, focusing narrowly on traditional classification problems and failing to capture the full spectrum of language understanding capabilities of modern LLMs. Benchmarks such as SuperGLUE (Wang et al., 2019) and XTREME (Hu et al., 2020) predominantly assess specific NLP tasks, limiting their ability to evaluate the broader, more generalized language capabilities that contemporary LLMs are seemingly capable of.

This issue is even more crucial for languages other than English, which are often evaluated on translated benchmarks that are prone to numerous biases and quality issues. To address this gap, we follow the tradition of using standardized exams (Hendrycks et al., 2020; Achiam et al., 2023) and introduce the first version of a Swedish benchmark called SweSAT-1.0¹ sourced from the Swedish university entrance exam, *Swedish Scholastic Aptitude Test* (‘Högskoleprovet’ in Swedish). The exam encompasses both verbal and quantitative reasoning tests across several sub-categories, such as reading comprehension, mathematical problem solving, and logical reasoning.

SweSAT-1.0 is sourced from the last eight exams over the past five years. The benchmark is prepared through automatic parsing of the exam files, followed by manual checks to correct any parsing errors. It currently comprises 867 questions and has the following advantages over most existing benchmarks: i) it is free from translation biases and culturally irrelevant content; ii) it allows researchers to control for data contamination² as the exact administration dates of the exams are known; iii) it broadens the range of tasks available for evaluation in Swedish; and iv) it indirectly allows a comparison against the real exam takers as the results are publicly available.

In addition to presenting the benchmark, we evaluate a wide range of popular multilingual and Swedish-oriented LLMs. The results show that while multilingual LLMs outperform their Swedish-oriented counterparts, even the commercial models fail at solving the reasoning tasks in Swedish, highlighting a crucial shortcoming of the existing LLMs. We hope that the benchmark will

¹The dataset can be accessed here: <https://github.com/NLP-RISE/swesat>

²Data contamination occurs when some or all of the test data is inadvertently included in the training set (Li et al., 2024).

Section	Total # questions	Description
ORD	160	<i>Vocabulary</i> : Tests the understanding of in-domain words and synonyms.
LÄS	160	<i>Reading comprehension</i> : Assesses the ability to make inference from a text.
MEK	160	<i>Sentence completion</i> : Assesses the ability to complete sentences via cloze tests.
XYZ	157	<i>Mathematical problem-solving</i> : Tests arithmetic, algebra, geometry, statistics, and functions.
KVA	140	<i>Quantitative comparisons</i> : Measures the ability to compare quantities in math concepts.
NOG	90	<i>Data sufficiency</i> : Evaluates the ability to determine if data is sufficient for solving a problem.

Table 1: Overview of exam sections in SweSAT-1.0, with total number of questions per section.

contribute to the evaluation of LLM performance in Swedish and encourage further research and development in multilingual contexts.

2 Related Work

There are a few benchmarks specifically designed to evaluate NLP models in Swedish, with SuperLim (Berdičevskis et al., 2023) and ScandEval (Nielsen, 2023) being the most prominent examples. Created as the Swedish counterpart to SuperGLUE, SuperLim is a comprehensive test suite that consists of 15 tasks, such as word analogy, pronoun resolution, and text summarization.³ If not adapted from English through translation, the featured datasets are either constructed by reformatting pre-existing tasks or created from scratch using pre-existing corpora. The reliance on pre-existing datasets raises concerns about data contamination, and the use of translation could introduce bias, which signals the need for new and complementary evaluation datasets. ScandEval (Nielsen, 2023; Nielsen et al., 2024), on the other hand, provides a multilingual evaluation suite spanning a subset of North Germanic languages, among them Swedish. Despite broad task coverage, the majority of ScandEval datasets are revisited versions of existing datasets, which again raises concerns about whether data contamination and the use of machine translation could undermine the evaluation process.

3 Dataset Description

SweSAT-1.0 is a benchmark dataset sourced from the publicly available *Swedish Scholastic Aptitude Test*,⁴ a standardized Swedish university entrance exam. The exam is written and administered by the Swedish Council for Higher Education and used for admission to higher education in Sweden.

³We note that one word-level task in SuperLim is directly taken from the *ORD* section of SweSAT (see Table 1).

⁴<https://www.studera.nu/hogskoleprov>

The exam consists of two main parts: verbal and quantitative, each containing four sections. Each exam includes 160 multiple-choice questions taken over a single day, lasting almost 8 hours (including breaks). This exam has been selected for its high quality; since it is written specifically to assess students’ verbal and quantitative reasoning skills in Swedish, we eliminate the risk of cultural and linguistic biases.

Sample questions can be found in Appendix C. We refer interested readers to Stage and Ögren (2004) for more detailed information on the exam.

3.1 Dataset Construction

The dataset was constructed through a semi-automatic process. Although the exam files are available in PDF format, extracting the content correctly proved challenging due the documents’ structure and formatting. For the verbal part, we employed pdfplumber,⁵ a popular Python library for PDF parsing. This approach worked well for extracting plain text but struggled with recognizing and preserving the format of mathematical expressions in the quantitative sections. Therefore, we adopted a different method for quantitative questions: we first converted each page into a high-resolution image, then performed OCR using GPT-4o (2024-08-06) with a detailed prompt (see Appendix A) to accurately capture both the text and mathematical formulas. The latter were represented in LaTeX in a consistent format, following common practice (Wang et al., 2023; Zhang et al., 2023). Despite our best efforts, we discovered that there were various errors in the final output, such as improper handling of hyphenated words at line breaks, italicized words jumping onto the wrong lines, or LaTeX formatting issues. Therefore, each exam was manually checked and corrected for errors to ensure accuracy and consistency.

⁵<https://github.com/jsvine/pdfplumber>

Model	ORD	LAS	MEK	XYZ	KVA	NOG	Average
Aya-23-8B	43.12	40.00	40.94	18.75	18.75	10.42	28.66
Gemma-2-9b	85.62	82.50	86.25	31.77	30.31	31.77	58.04
Gemma-2-27b	91.56	90.62	90.94	37.50	36.25	32.29	63.19
GPT-SW3-1.3b	16.88	22.50	25.94	18.23	21.25	9.38	19.03
GPT-SW3-6.7b-v2	20.00	21.25	25.62	17.19	19.38	11.46	19.15
GPT-SW3-20b	21.56	30.63	30.31	18.75	22.50	12.50	22.71
AI-Sweden/Llama-3-8B	71.25	56.25	59.69	21.88	20.31	13.02	40.40
Llama-3-8B	68.44	65.00	55.62	18.75	26.25	25.00	43.18
Llama-3.1-8B	80.31	69.38	58.75	20.83	31.25	18.23	46.46
GPT-4o-mini (2024-07-18)	97.50	84.38	96.25	32.29	38.12	35.42	63.99
GPT-4o (2024-08-06)	100.0	92.50	99.38	47.40	45.62	45.83	71.79

Table 2: Average performance of baseline models across question types on the entire SweSAT 1.0.

3.2 The SweSAT-1.0 Dataset

SweSAT-1.0 includes the last five years of the exam (from 2020 to 2024) held over eight different sessions.⁶ Following our primary focus on evaluating text-based language models in Swedish, SweSAT-1.0 includes only the verbal and quantitative reasoning sections that do not require multimodal inputs, thus omitting the entire section of DTK (Diagrams, Tables, and Maps) as well as any question that requires visual information to solve. The ELF (English Reading Comprehension) section is also excluded from the dataset since our primary focus is on Swedish.⁷ The dataset currently comprises 867 questions, covering six question types, as shown in Table 1. All questions are in the multiple-choice format: ORD and NOG sections have five options whereas the remaining sections have only four.

Alongside the questions, we prompt the models using the official exam instructions to simulate the real exam-taking scenario. The original instructions include an explanation of the exam section, and one sample question and answer for five of the sections included in this dataset: ORD, MEK, KVA, NOG, and XYZ. However, the sample question and its answer in the XYZ section is excluded as it contains figures incompatible with the benchmark setup.

This results in a mix of one-shot and zero-shot prompts. We exclude the sample questions and answers in order to conduct the experiments using a zero-shot version of the exam instructions. The

⁶At the time when the dataset was constructed, the 2024 fall exam has not yet been held, and only the spring exam is available for 2020.

⁷Note that the ELF sections are not publicly available, so these questions could not be included in the benchmark.

zero-shot version of these instructions (which excludes any example questions) as well as the mixed one-shot version are both included in the dataset release to facilitate a standardized evaluation across all sections.

4 Baselines

In this section, we evaluate the performance of a range of LLMs, each with different levels of Swedish coverage during training, on SweSAT-1.0. The primary purpose of this baseline evaluation is to evaluate the dataset itself by analyzing how some of the most popular LLMs perform on its tasks to ensure that the dataset is sufficiently challenging and valuable as a benchmark. By doing so, we also provide reference scores for future studies while exploring the current capabilities of LLMs in Swedish. Our evaluation includes a range of instruction-tuned open-source models such as GPT-SW3 (Ekgren et al., 2024), Gemma-2 (Team et al., 2024), Aya (Üstün et al., 2024), Llama 3 and 3.1 (Dubey et al., 2024), as well as the commercial GPT-4o-mini (2024-08-06) and GPT-4o (2024-07-18) models (Achiam et al., 2023). The entire model list can be found in Appendix B.

4.1 Experimental Setup

We use the original exam instructions (excluding the sample questions) as zero-shot prompts to assess the models’ performance under authentic exam-taking conditions. To ensure adherence to these instructions, we add a brief directive⁸ at

⁸*Svara endast med bokstaven på det rätta alternativet utan någon förklaring* (‘Answer only with the letter of the correct option without any explanation’).

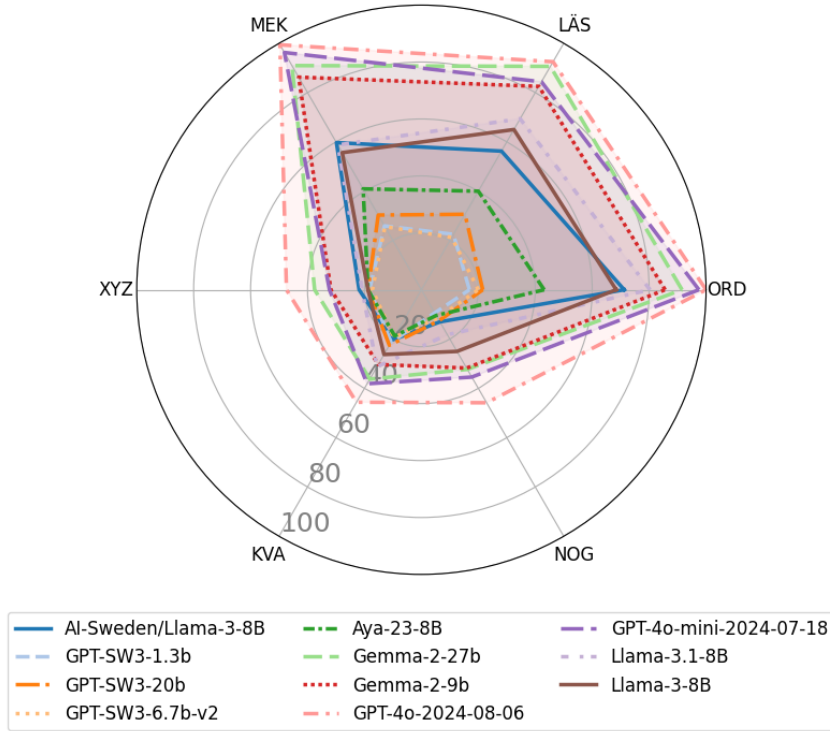


Figure 1: Radar chart comparing model performances across different question types

the end of each question, prompting the models to return a single letter as the desired response format. The models may at most output a single token; any output that does not exactly match one of the allowed answer letters (A, B, C, D, or E; depending on the question type) is discarded. Questions are presented one by one, with reading passages repeated in the prompt before each LÄS question. Answers are generated using greedy decoding⁹ to ensure a deterministic output, hence reproducibility, by selecting the highest probability response at each step. As all questions are multiple-choice, we use accuracy as our evaluation metric.

4.2 Results

The average performance of baseline models across eight exams is shown in Table 2. All models perform markedly better on the verbal sections (MEK, LÄS, and ORD), with Gemma models achieving around 90% accuracy and GPT-4o achieving almost a perfect score in the MEK and ORD sections. On the other hand, quantitative sections yield significantly lower scores, with even GPT-4o failing on the majority of questions. Swedish-oriented models — all models in the GPT-SW3 family in addition to a fine-tuned Llama 3 version — con-

sistently show lower accuracy across all question types. To note a special case, we find that the aforementioned Llama 3 instruct-variant, fine-tuned on The Nordic Pile (Öhman et al., 2023),¹⁰ exhibits better performance than all evaluated GPT-SW3 models. Yet, it achieves slightly lower average accuracy than the original Llama 3 on five of the eight exams. This raises questions on whether continued pre-training on a mix of Scandinavian languages is useful for this task, or whether it may depend on the nature of the selected dataset.

The differences among models across question types are further illustrated in Figure 1. The results suggest that current LLMs have significant limitations in quantitative reasoning tasks in Swedish. Furthermore, we also analyze the patterns in the way models provide answers through confusion matrices (see Appendix D). GPT-SW3 models are observed to frequently select the same options (e.g., consistently choosing A or alternating between A and D in the case of the 20B version), which highlights potential shortcomings in following instructions. However, the selected options for other models are more evenly distributed across the potential answers, suggesting better task understanding,

⁹For GPT-4o models, we set the temperature to $1e^{-9}$.

¹⁰A dataset comprised of a mix of Scandinavian languages and English.

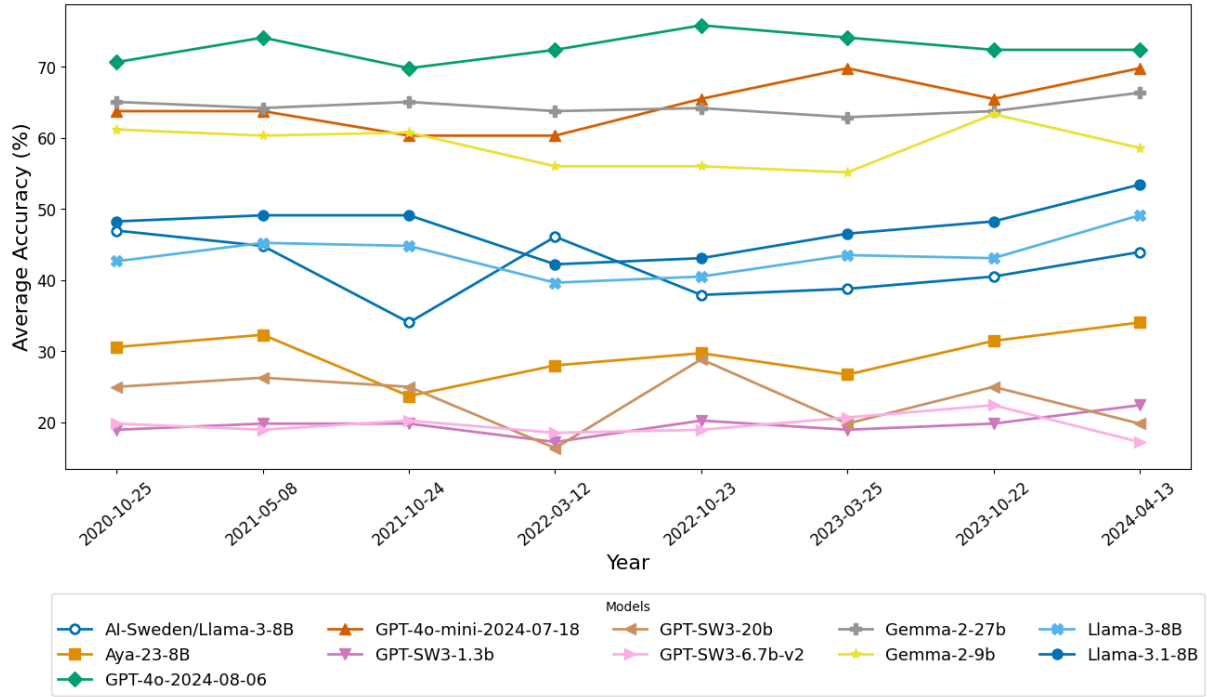


Figure 2: Average performance of baseline models across years

even though the correct option is not consistently identified. Yet, it should also be noted that this evaluation represents a particularly challenging setting where we require the models to produce only the correct answer without using techniques like chain-of-thought prompting (Wei et al., 2022) and without model-specific prompt engineering.

Finally, we investigate the potential impact of data contamination on LLM performance. As shown in Figure 2, all models exhibit a highly consistent performance across exam years, with an average standard deviation of only 2.8% in accuracy. This suggests that the contamination effect is absent and that the exam difficulty is consistent across years.

5 Conclusion

In this paper, we present a comprehensive benchmark to evaluate LLMs’ various abilities in Swedish, using the university entrance exam. We believe our benchmark provides a consistent framework for testing LLM performance across a range of tasks detailed above, with an option to control for data contamination in model training through exam timestamps. Our baseline evaluations reveal the high accuracy of multilingual models across verbal tasks compared to their Swedish-centric counterparts – but also the overall weakness of all tested

models on the reasoning tasks.

Acknowledgments

We gratefully acknowledge the support of the Swedish Research Council (grant no. 2022-02909). The experiments with the open-source LLMs were enabled by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 202206725 and 2024-01506. We thank NAISS for providing computational resources under Project 2024/22-211. We also thank the reviewers for their valuable feedback and suggestions.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Aleksandrs Berdičevskis, Gerlof Bouma, Robin Kurtz, Felix Morger, Joey Öhman, Yvonne Adesam, Lars Borin, Dana Dannélls, Markus Forsberg, Tim Isbister, et al. 2023. Superlim: A Swedish language understanding evaluation benchmark. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8137–8153.

- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Ariel Ekgren, Amaru Cuba Gyllensten, Felix Stoltenwerk, Joey Öhman, Tim Isbister, Evangelia Gogoulou, Fredrik Carlsson, Judit Casademont, and Magnus Sahlgren. 2024. GPT-SW3: An autoregressive language model for the Scandinavian languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7886–7900, Torino, Italia. ELRA and ICCL.
- 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*.
- Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. In *International Conference on Machine Learning*, pages 4411–4421. PMLR.
- Yucheng Li, Yunhao Guo, Frank Guerin, and Chenghua Lin. 2024. An open-source data contamination report for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 528–541, Miami, Florida, USA. Association for Computational Linguistics.
- Dan Nielsen. 2023. ScandEval: A benchmark for Scandinavian natural language processing. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 185–201.
- Dan Saattrup Nielsen, Kenneth Enevoldsen, and Peter Schneider-Kamp. 2024. Encoder vs decoder: Comparative analysis of encoder and decoder language models on multilingual NLU tasks. *arXiv preprint arXiv:2406.13469*.
- Joey Öhman, Severine Verlinden, Ariel Ekgren, Amaru Cuba Gyllensten, Tim Isbister, Evangelia Gogoulou, Fredrik Carlsson, and Magnus Sahlgren. 2023. The Nordic Pile: A 1.2tb Nordic dataset for language modeling. *arXiv preprint arXiv:2303.17183*.
- Christina Stage and Gunilla Ögren. 2004. *The Swedish Scholastic Assessment Test (SweSAT): Development, Results, and Experiences*. EM nr 49. Umeå University, Department of Educational Measurement, Umeå, Sweden.
- Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.
- Ahmet Üstün, Viraat Aryabumi, Zheng-Xin Yong, Wei-Yin Ko, Daniel D’souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, et al. 2024. Aya model: An instruction finetuned open-access multilingual language model. *arXiv preprint arXiv:2402.07827*.
- 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* 32.
- Xiaoxuan Wang, Ziniu Hu, Pan Lu, Yanqiao Zhu, Jieyu Zhang, Satyen Subramaniam, Arjun R Loomba, Shichang Zhang, Yizhou Sun, and Wei Wang. 2023. SciBench: Evaluating college-level scientific problem-solving abilities of large language models. *arXiv preprint arXiv:2307.10635*.
- 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. In *Advances in Neural Information Processing Systems* 35, pages 24824–24837.
- Wenxuan Zhang, Mahani Aljunied, Chang Gao, Yew Ken Chia, and Lidong Bing. 2023. M3Exam: A multilingual, multimodal, multilevel benchmark for examining large language models. In *Advances in Neural Information Processing Systems* 36, pages 5484–5505.

A Parsing the Quantitative Questions

Parsing the quantitative part of the exams, i.e., sections containing mathematical expressions and visual elements, proved to be very challenging for standard PDF parsing libraries in accurately recovering complex mathematical equations. Therefore, we performed OCR on each page separately using GPT-4o with the following prompt:

The image contains an exam sheet with both text-based and visual-based questions. Your task is to extract only the text-based questions and answers and format them into the following JSON structure. If a question contains any actual visual (such as a diagram, shape, figure, graph, or table visible in the image), set "is_accompanied_with_visual" to "yes" and specify the type of the visual in the "visual_type" field (e.g., "diagram", "graph", etc.). In this case, set the "question" field to "Visual required to solve this question" and leave the "answers" field blank. If the question describes geometrical objects (like lines, points, or coordinates) but does not include an actual visible diagram, treat it as a text-only question. Set "is_accompanied_with_visual" to "no" and fully extract the question and answers, preserving all formulas and numbers exactly as shown.

JSON Format:

```
[
  {
    "question_number": <number>,
    "question": "<question.text.with.formulas>",
    "answers": {
      "a": "<option a>",
      "b": "<option b>",
      "c": "<option c>",
      "d": "<option d>"
    },
    "is_accompanied_with_visual": "<yes/no>",
    "visual_type": "<visual.type>",
    "question.type": "<XYZ/KVA/NOG/DTK>"
  }
]
```

Further Instructions:

- Fully extract text-based questions and answers exactly as shown without modification.
- Do not simplify or paraphrase any part of the question. Classify the question as XYZ, KVA, NOG, or DTK.
- All the math formulas must be represented as LaTeX code, surrounded by \$ (e.g., $\frac{1}{3}$). Convert special math notations, such as $\sqrt{\quad}$, into the corresponding LaTeX format. Wrap all the formulas with \$ symbols.
- Pay extra attention to capturing exponents correctly. Be aware that there may be fractional exponents, such as $(x^{\frac{5}{15}})^{\frac{1}{5}}$.
- Distinguish clearly between similar characters, particularly "2" and "5" and "6" and "8", to avoid confusion.
- Pay close attention to capturing nested exponents and grouping symbols accurately. When encoding expressions, make sure to wrap exponents and nested exponents within braces {} to maintain the correct mathematical hierarchy. For example, $\left(x^7\right)^{\frac{1}{2}}$.
- Validate the resulting LaTeX expression by ensuring it visually matches the intended structure of the original mathematical notation.

- Pay special attention to minus signs. Ensure that all minus signs are correctly included and accurately placed.
- Always encode expressions properly in LaTeX. Make sure to use `\` for LaTeX commands and wrap **all formulas** with \$ symbols.
- Ensure all LaTeX functions, such as `\times` and `\text`, are used only within math mode (i.e., surrounded by `...`).

B Baseline models

Table 3 provides the repository names of the baseline models on <https://huggingface.co/>, alongside their simplified names used throughout the text. As for the OpenAI models, we used the (2024-07-18) release of GPT-4o-mini and the (2024-08-06) release of GPT-4o.

Simplified Name	HuggingFace model repository
Aya-23-8B	CohereForAI/aya-23-8B
Gemma-2-27b	google/gemma-2-27b-it
Gemma-2-9b	google/gemma-2-9b-it
GPT-SW3-1.3b	AI-Sweden-Models/gpt-sw3-1.3b-instruct
GPT-SW3-20b	AI-Sweden-Models/gpt-sw3-20b-instruct
GPT-SW3-6.7b-v2	AI-Sweden-Models/gpt-sw3-6.7b-v2-instruct
AI-Sweden/Llama-3-8B	AI-Sweden-Models/Llama-3-8B-instruct
Llama-3.1-8B	meta-llama/Llama-3.1-8B-Instruct
Llama-3-8B	meta-llama/Meta-Llama-3-8B-Instruct

Table 3: HuggingFace model repository names of the baseline models

C Example Questions

Figures 3 and 4 show sample questions from the NOG and KVA question types (respectively), as shown in the exam sheet.

D Confusion Matrices

Figure 5 presents confusion matrices summarizing model predictions over the entire SweSAT-1.0 dataset.

28. Vad är medelvärdet av a och b ?

- (1) Medelvärdet av $(a + 5)$ och $(b + 9)$ är lika med 10,5.
(2) Medelvärdet av a , $(b - 1)$ och 3 är lika med 3.

Tillräcklig information för lösningen erhålls

- A i (1) men ej i (2)
B i (2) men ej i (1)
C i (1) tillsammans med (2)
D i (1) och (2) var för sig
E ej genom de båda påståendena

What is the mean of a and b ?

1. The mean of $(a + 5)$ and $(b + 9)$ is equal to 10.5.
2. The mean of a , $(b - 1)$ and 3 is equal to 3.

Sufficient information for solving the problem is obtained:

- A from (1) but not from (2)
B from (2) but not from (1)
C from (1) and (2) together
D from both (1) and (2) each by itself
E not from the two statements

Figure 3: A sample from the NOG question type in Swedish (top) and translated to English (bottom)

13. Medelvärdet av de tre talen x , y och z är 12. Summan av y och z är 30.

Kvantitet I: x

Kvantitet II: 9

- A I är större än II
B II är större än I
C I är lika med II
D informationen är otillräcklig

The mean value of the three numbers x , y and z is 12. The sum of y and z is 30.

1. Quantity I: x
2. Quantity II: 9

- A I is greater than II
B II is greater than I
C I is equal to II
D The information is insufficient

Figure 4: A sample from the KVA question type in Swedish (top) and translated to English (bottom)

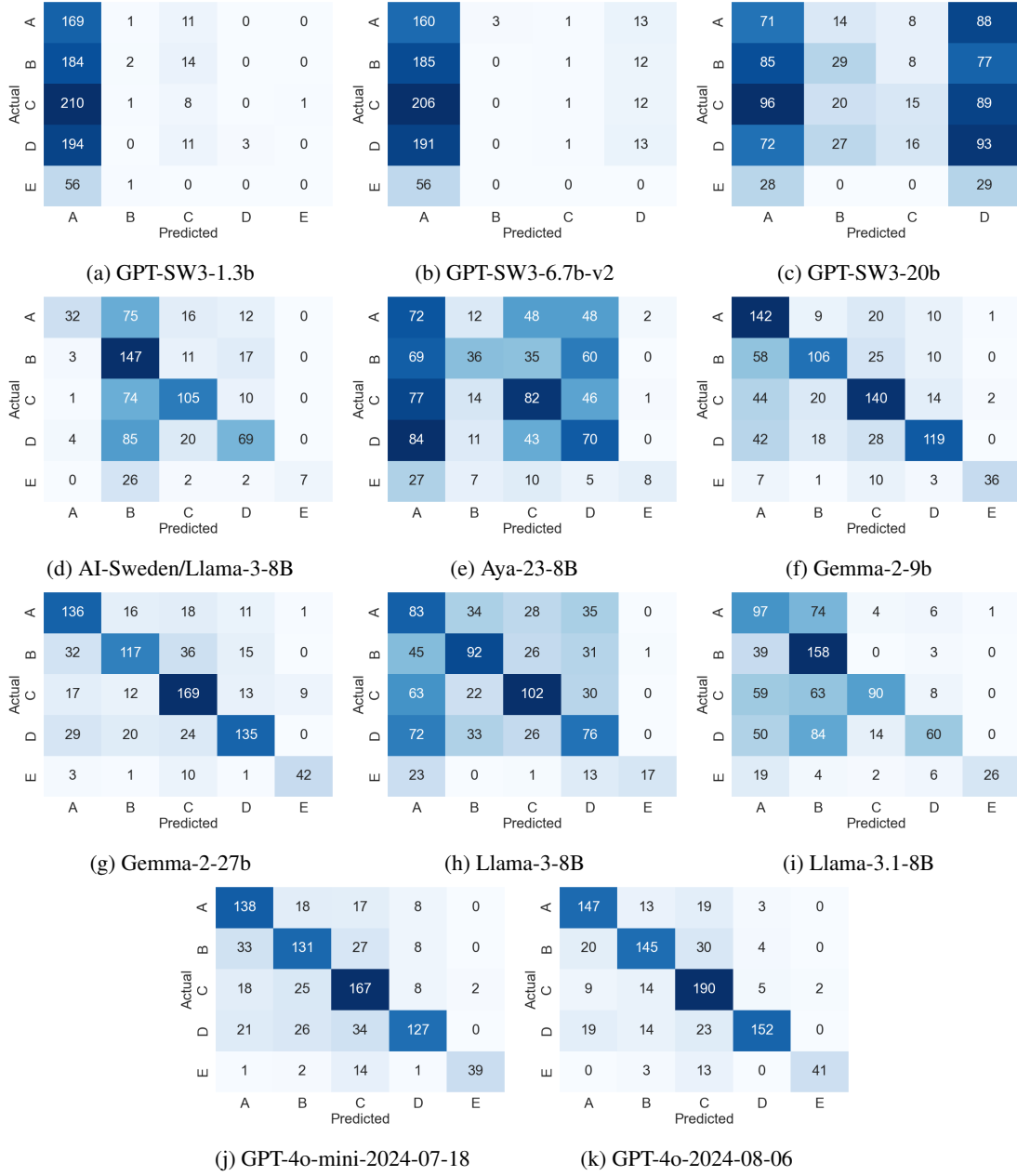


Figure 5: Confusion matrices for the baseline models on SweSAT 1.0. Note that only two sections feature five options, hence the lower frequency of option E.

How Well do LLMs know Finno-Ugric Languages? A Systematic Assessment

Hele-Andra Kuulmets Taido Purason Mark Fishel

Institute of Computer Science

University of Tartu

{hele-andra.kuulmets, taido.purason, mark.fisel}@ut.ee

Abstract

We present a systematic evaluation of multilingual capabilities of open large language models (LLMs), specifically focusing on five Finno-Ugric (FiU) languages. Our investigation covers multiple prompting strategies across several benchmarks and reveals that Llama 2 7B and Llama 2 13B perform weakly on most FiU languages. In contrast, Llama 3.1 models show impressive improvements, even for extremely low-resource languages such as Võro and Komi, indicating successful cross-lingual knowledge transfer inside the models. Finally, we show that stronger base models outperform weaker, language-adapted models, thus emphasizing the importance of the choice of the base model for successful language adaptation.

1 Introduction

Large language models (LLMs) have recently made significant advances in multilingual settings. For instance, GPT-4 achieves 80.9% accuracy for Latvian and 76.5% for Icelandic on the 3-shot MMLU benchmark (OpenAI et al., 2024). For some time, strong multilingual capabilities were mainly limited to proprietary models, such as ChatGPT¹ and Claude², whose weights, training details, and inference processes are kept private. These models outperformed open LLMs³ like Llama 2 models (Touvron et al., 2023), on non-English tasks. However, open-weight LLMs have recently begun to close this gap (Dubey et al., 2024; Jiang et al., 2024), even though the officially

supported languages of these models remain limited and the primary focus is on those with significantly more data available than for Finno-Ugric (FiU) languages.

On the other hand, it has been observed that even models optimized solely for English, such as the Llama 2 family models (Touvron et al., 2023), demonstrate some understanding of a wide range of languages beyond their intended use (Holtermann et al., 2024). In experiments conducted by Holtermann et al. (2024), the Llama 2 7B chat model correctly answered 14% and 40% of basic open-ended questions in Estonian and Finnish, respectively, even though only 0.03% of the Llama 2 training data was in Finnish and less than 0.005% in Estonian (Touvron et al., 2023).

This work evaluates the multilingual capabilities of open LLMs on five FiU languages: Finnish, Estonian, Livonian, Võro, and Komi. Among these, Finnish and Estonian are the most well-resourced, making it easier to adapt existing LLMs for these languages through continued pretraining (Kuulmets et al., 2024; Luukkonen et al., 2023). In contrast, Võro, Livonian, and Komi are extremely low-resource languages, making language-specific adaptation considerably more challenging.

The aim of this work is to clarify the capabilities of open LLMs in understanding FiU languages. While it is evident that open LLMs can understand these languages to some degree (Holtermann et al., 2024), their proficiency and comparative performance across models remain largely unexplored. We focus on Llama models, which have demonstrated state-of-the-art performance and competitiveness with proprietary models (Dubey et al., 2024; Touvron et al., 2023) and have been widely used in non-English adaption (Kuulmets et al., 2024; Etxaniz et al., 2024; Lin et al., 2024; Fujii et al., 2024; Dima et al., 2024; Basile et al., 2023). Another reason for focusing on Llama

¹<https://openai.com/index/chatgpt/>

²<https://www.anthropic.com/claude>

³Models that have publicly accessible weights available for use, modification, and research.

models is that the newer Llama 3.1 models are natively multilingual, potentially improving performance on unsupported languages as well. For further insights, we compare Llama models with Mistral NeMo (Jiang et al., 2024), another natively multilingual open model shown to be competitive with Llama 3.1 model of the same size.

We evaluate only base models rather than chat-optimized models, as most knowledge is acquired during pretraining (Zhou et al., 2023; Lin et al., 2023). In other words, a stronger base model offers greater potential for developing a strong chat model. Consequently, the performance of base models on different FiU languages can serve as a relative estimate of the chat model’s quality.

The evaluation is conducted using several existing benchmarks that include one or more Finno-Ugric languages. We examine both the zero-shot and few-shot capabilities of these models. Additionally, we explore whether chain-of-thought prompting, which involves first translating the input to English, could improve results on Finno-Ugric languages. In summary, we seek to answer the following research questions:

1. How well can open LLMs solve tasks in Finno-Ugric languages?
2. What is the expected improvement from few-shot prompting over zero-shot prompting in solving tasks in Finno-Ugric languages?
3. Can chain-of-thought prompting, where the model first translates the input into English, improve the performance of open LLMs on Finno-Ugric languages?

2 Related Work

2.1 Multilingual LLMs

While state-of-the-art LLMs are typically trained on English-centric data, they exhibit some multilingual capabilities (Brown et al., 2020; Holtermann et al., 2024), even for languages with minimal representation in the training data (Holtermann et al., 2024; Touvron et al., 2023). This suggests that knowledge transfer from high-resource languages to low-resource languages must occur at least to some extent within the model. These multilingual capabilities can be further enhanced through continued pretraining in the target languages, even with just a few billion tokens of data (Pires et al., 2023; Cui et al., 2024; Kuulmets et al., 2024; Etxaniz et al., 2024).

Recent open LLMs such as Llama 3.1 (Dubey et al., 2024), Mistral NeMo (Jiang et al., 2024), and Tower (Alves et al., 2024) are specifically optimized for multilingual performance. For example, Llama 3.1 models officially support seven non-English languages (Dubey et al., 2024), Mistral NeMo is particularly strong in ten languages other than English (Jiang et al., 2024), and Tower is trained on a multilingual dataset consisting of ten languages, including English. According to Dubey et al. (2024), the strong performance in non-English languages is achieved by increasing the proportion of multilingual data in the pretraining dataset and incorporating high-quality target language instructions into the instruction-tuning data.

However, neither Mistral NeMo nor Llama 3.1 models officially support Finno-Ugric languages. The amount of Finno-Ugric data in their pretraining corpora is unknown but is likely very limited. For example, Purason et al. (2024) presented experiments on adapting LLMs to FiU languages, but gathered only 2.6 million characters of pretraining data for Livonian, 14 million for Võro, and 579 million for Komi.

2.2 In-context Learning

In-context learning (ICL) (Brown et al., 2020) is a method where a pretrained language model *learns* to generate the desired output for a given task from the context of the prompt, without any gradient updates. One of the most common applications of ICL is few-shot prompting, where a few example question-answer pairs are provided in the prompt to guide the model in solving the task.

2.2.1 Chain-of-thought Prompting

Chain-of-thought (CoT) prompting (Wei et al., 2023) is a prompting technique that improves upon few-shot prompting. With CoT, the example demonstrations provided in the prompt include a series of intermediate reasoning steps that conclude with an answer as opposed to being just question-and-answer pairs. While initially proposed to improve English reasoning in LLMs, Shi et al. (2022) showed that CoT prompting turns English-centric PaLM and GPT-3 into multilingual reasoners, achieving strong results even in languages whose proportion in the training data is as small as 0.01%. Notably, they achieve an accuracy of 91% on the Estonian subset of the multilingual commonsense reasoning benchmark XCOPA

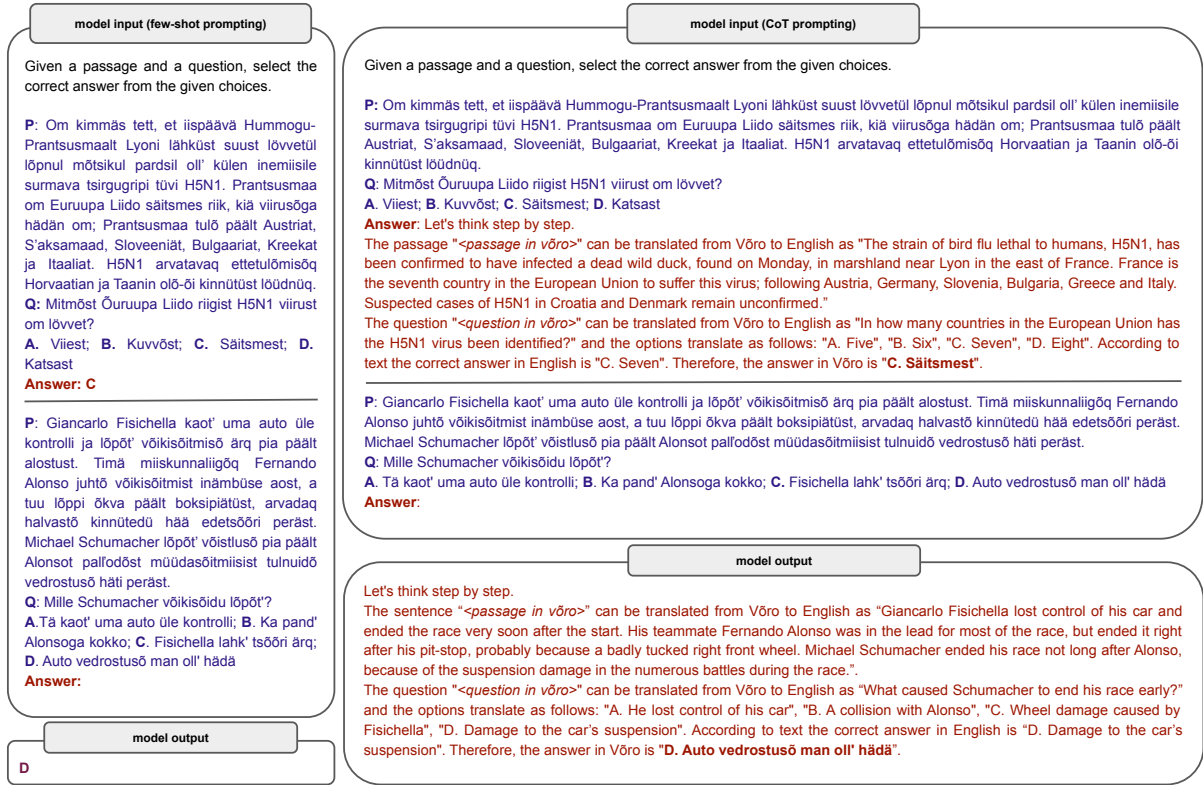


Figure 1: Model input and expected output for few-shot prompting (left) and for CoT prompting where the intermediate step involves translating the input from the source language (Võro) to English. The example is taken from the Belebele benchmark.

(Ponti et al., 2020) (average accuracy 89.9%) with PaLM. Their observation that there is no strong correlation between performance and language frequency in the training corpora leads them to suggest that, to some extent, language models can transfer knowledge from high-resource to low-resource languages, and that this ability is mainly facilitated by scale.

2.3 English as Pivot Improves Multilingual Capabilities of LLMs

One of the findings of Shi et al. (2022) is that CoT prompting with intermediate reasoning steps in English outperforms native CoT prompting with steps in the target language. Huang et al. (2023) show that conversational models such as ChatGPT and Llama-2 also benefit from using English as a pivot language – asking the model to first retell the request in English improves performance on non-English tasks. Notably, this strategy eliminates the need for few-shot examples, meaning that the ability to translate between English and the target language must have been learned during (pre)training rather than from parallel exam-

ples provided in the context. Zhang et al. (2024) instruction-tune pretrained LLMs to first process instructions in the pivot language English and then produce responses in the target language.

The phenomenon has been explicitly studied by Zhang et al. (2023), who show that ChatGPT behaves similarly to subordinate bilinguals whose representation of knowledge is strongly biased toward English and, as a consequence, translates all non-English inputs to English. Wendler et al. (2024) investigate the latent representations of token embeddings of LLaMA 2 and find that in the middle layers, these are closer to English tokens, and only in the final layers shift towards target language tokens. They interpret this result as the "concept space" being closer to English.

3 Datasets

The selection of benchmark tasks is determined by the availability of datasets for our target languages. In total, we evaluate the models on five tasks using nine datasets. These datasets primarily originate from cross-lingual benchmarks that include multiple languages. For our experiments, we

task	datasets	est	fin	vro	kpv	liv
machine translation	FLORES-200 (NLLB Team, 2022), SMUGRI-FLORES (Yankovskaya et al., 2023)	✓	✓	✓	✓	✓
multiple choice QA	Belebele (Bandarkar et al., 2024), Belebele-smugri (Purason et al., 2024)	✓	✓	✓	✓	✓
text classification	SIB-200 (Adelani et al., 2024), SIB-smugri (Purason et al., 2024)	✓	✓	✓	✓	✓
extractive QA	EstQA (Käver, 2021), TyDiQA (Clark et al., 2020)	✓	✓			
commonsense reasoning	XCOPA (Ponti et al., 2020)	✓				

Table 1: Tasks and datasets used for benchmarking the models.

use only the subsets that correspond to the selected target languages. A summary of the datasets, tasks and their language coverage is provided in Table 1.

Machine Translation (MT) Our evaluation includes translation tasks from low-resource FiU languages to English. For this purpose, we use the FLORES-200 benchmark (NLLB Team, 2022), which includes Estonian and Finnish, and the FLORES-SMUGRI dataset (Yankovskaya et al., 2023), which translates the first 250 sentences from FLORES-200 to ten low-resource FiU languages, including Komi, Võro, and Livonian. To ensure consistency, we use only the first 250 sentences of FLORES-200 for Estonian and Finnish as well.

Multiple choice QA This task involves selecting the correct answer from a set of options, given a passage, a question, and possible answer choices. We use the Belebele dataset (Bandarkar et al., 2024), which augments paragraphs from the FLORES-200 benchmark with corresponding questions and answer choices. Among its 122 languages, Belebele includes Estonian and Finnish. Purason et al. (2024) further extend the dataset to cover Võro, Livonian, and Komi, resulting in a total of 127 examples per language. For consistency, we use the same number of examples for Estonian and Finnish.

Topic classification We use the massively multilingual text classification benchmark SIB-200 (Adelani et al., 2024), which bases on the FLORES-200 benchmark and comprises 125 examples per language. This benchmark involves classifying sentences from FLORES-200 into seven categories. Purason et al. (2024) extend it to include Võro, Livonian, and Komi.

Extractive QA It is a task in which the objective is to identify a snippet from a given passage

that answers a given question. There exists an Estonian dataset for this task, EstQA (Käver, 2021) which includes 603 test examples, each potentially featuring multiple golden answers. In our evaluation, however, we consider only the first answer for each example. Finnish is included into the multilingual dataset TyDiQA (Clark et al., 2020) covering eight typologically diverse languages. Both of these datasets are translation-free, meaning they are created directly in the target language rather than translated from English. In our experiments, we use Finnish samples from the `secondary-task` subset of TyDiQA, where the task format is similar to EstQA. This subset contains 782 Finnish test examples.

Commonsense reasoning Reasoning skills have been observed to be less trivially transferable across languages than question-answering abilities (Kuulmets et al., 2024; Zhu et al., 2024; Huang et al., 2023). To avoid creating a misleading impression of the models’ capabilities, it is essential to include reasoning datasets in our evaluation benchmarks. To the best of our knowledge, only one such benchmark incorporates a Finno-Ugric language: XCOPA (Ponti et al., 2020), which includes Estonian. XCOPA requires models to identify which of two answer choices most plausibly represents the cause or effect of a given premise. The test dataset comprises 500 examples.

4 Methodology

For tasks that do not require open-ended text generation (e.g., Belebele, SIB, XCOPA), performance is evaluated by calculating the log likelihood of each possible answer choice and selecting the most likely one as the prediction. In contrast, tasks requiring open-ended text generation, such as FLORES, extractive QA, we use greedy decoding to generate predictions.

We report the results both in zero-shot and few-shot setting where we add either 1, 3 or 5 input-output pairs to the prompt to provide the model with task-specific guidance. Additionally, we investigate the impact of CoT prompting, which guides the model to generate intermediate reasoning steps before producing the final answer. Drawing inspiration from Shi et al. (2022), the intermediate steps require translating the input into English, identifying the answer in English, and translating it back to the target language. CoT prompting can also be used both in zero-shot⁴ and few-shot settings. In the zero-shot setting, the prompt ends with *“Let’s think step-by-step”* (Kojima et al., 2022), while in the few-shot setting, this is followed by explicit reasoning steps. Figure 1 illustrates model input and output in one-shot setting with and without CoT.

We use regexes to extract answers from the generated text in tasks requiring decoding. Although this approach may occasionally produce false negatives, the models generally adhere well to the output format in few-shot settings. We implement all evaluation strategies with `lm-eval-harness` framework (Gao et al., 2024) and make the task configurations publicly available.⁵

5 Results

5.1 Main Results

Table 2 shows 5-shot results (without CoT) across all tasks and models. In general, Llama 2 7B and Llama 2 13B perform significantly worse on the observed FiU languages than the Llama 3.1 family models. The exception is Finnish, on which the Llama 2 models are notably better than on the other FiU languages. This may be due to the larger amount of Finnish data in the Llama 2 training dataset (Touvron et al., 2023) when compared to data in other FiU languages. However, both Llama-2 7B and Llama 2 13B still appear weak on Finnish when compared to other models.

Llama-2 70B shows notable improvements over Llama 2 7B and Llama 2 13B on Estonian and Finnish across all tasks. The results for Belebele and SIB also indicate improvement for Võro, though the improvement in machine translation (FLORES) is less pronounced. Additionally, SIB appears to be generally too easy of a benchmark for the models, as Llama 2 7B already achieves

86% accuracy for Finnish. For other languages, the benchmark saturates with Llama 2 70B. For this reason, we exclude SIB from further analysis. Finally, we observe that Llama 2 models are the weakest on Komi and Livonian.

	L2-7B	L2-13B	L2-70B	L3.1-8B	L3.1-70B
SIB					
liv	64.8	61.6	83.2	74.4	77.6
kpv	68.0	59.2	83.2	77.6	87.2
vro	64.8	59.2	85.6	86.4	86.4
est	69.6	68.0	88.8	89.6	89.6
fin	85.6	81.6	91.2	87.2	89.6
Belebele					
liv	26.23	35.25	36.89	37.70	42.62
kpv	27.87	31.15	34.43	52.46	73.77
vro	27.05	32.79	44.26	50.82	73.77
est	28.69	36.07	66.39	68.03	88.52
fin	44.26	54.92	86.89	74.59	91.80
XCOPA					
est	49.2	51.8	67.6	69.2	92.6
FLORES (FiU → En)					
liv	6.8	9.3	12.0	10.5	16.1
kpv	5.4	6.0	7.3	10.3	21.9
vro	7.8	9.1	12.9	16.7	30.3
est	12.6	17.8	26.9	35.3	41.0
fin	29.6	31.9	34.6	32.0	37.1
Extractive QA					
<i>exact match</i>					
est	21.89	34.33	49.25	50.75	52.74
fin	51.66	48.34	53.45	58.31	47.06
<i>F1</i>					
est	35.35	51.39	66.72	70.87	73.76
fin	70.63	70.36	74.65	75.44	72.98
<i>BERTScore F1</i> (Zhang* et al., 2020)					
est	76.88	82.95	88.86	91.76	93.02
fin	88.50	87.95	89.60	90.63	88.67

Table 2: 5-shot results on all tasks. Accuracy is reported for SIB, Belebele and XCOPA. BLEU is reported for FLORES. BERTScore F1 was calculated using `bert-base-multilingual-cased`.

We notice that on Estonian and Finnish, Llama 2 70B is competitive with Llama 3.1 8B despite the latter being nearly nine times smaller, although Llama-3.1 8B appears to slightly underperform on Finnish, as indicated by the results of Belebele and FLORES.

When comparing Llama-3.1 8B to Llama-3.1 70B, the larger model clearly outperforms the smaller one on Belebele, FLORES, and XCOPA.

⁴We leave zero-shot CoT for future research.

⁵<https://github.com/TartuNLP/smugri-lm-eval-configs>

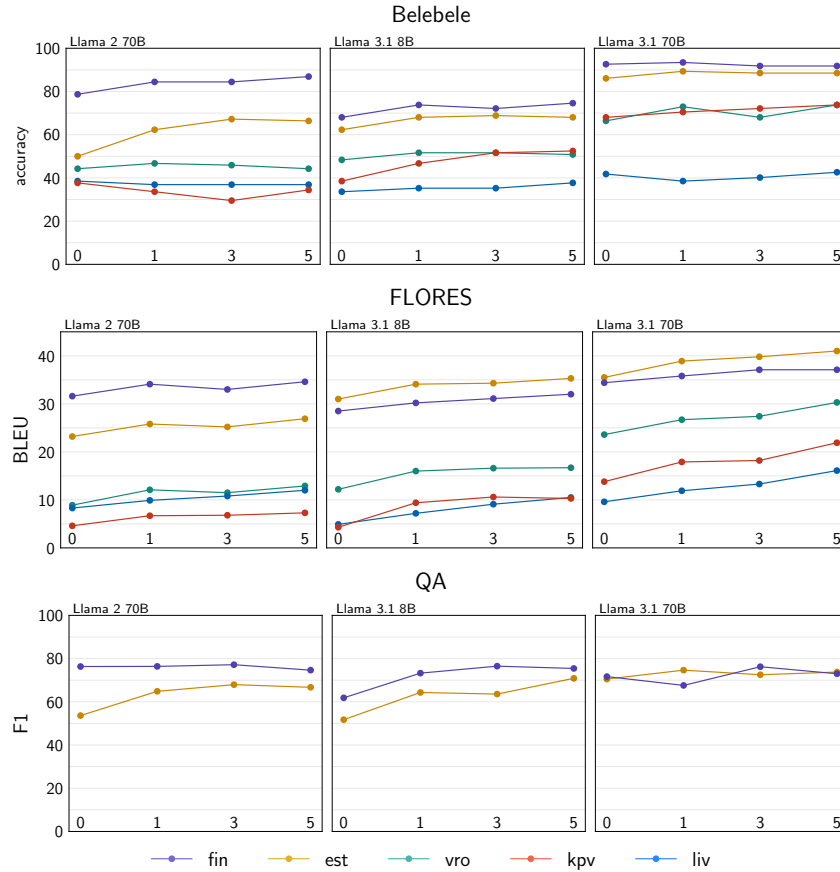


Figure 2: Effect of few-shot examples in 0, 1, 3 and 5-shot setting.

For Estonian and Finnish, the Llama-3.1 70B achieves nearly 90% accuracy on Belebele and XCOPA, along with very strong BLEU scores on the FLORES dataset. The improvements are also significant for extremely low-resource languages Võro, Komi and Livonian.

5.2 The Effect of Few-Shot Examples

We analyze the impact of few-shot examples on the models’ ability to solve tasks in FiU languages. We limit this analysis to three models: Llama 2 70B, Llama 3.1 8B, and Llama 3.1 70B due to their superior performance.

Figure 2 illustrates the results. For Belebele and QA tasks, one-shot prompting generally improves performance compared to zero-shot prompting. However, the gains from adding three or five examples vary significantly across tasks and languages. Notably, the improvements from few-shot examples are particularly inconsistent on the Finnish QA task with Llama-3.1 70B.

In contrast, on FLORES benchmark, the improvements are more consistent as the number of examples increases. Notably, Llama-3.1 70B

shows substantial gains when translating from Võro, Livonian, and Komi to English, with improvements of 6.6 BLEU points for Võro, 6.6 for Livonian, and 8.1 for Komi when using five examples compared to zero-shot prompting.

To conclude, few-shot prompting can yield notable gains in some cases—such as a 17% improvement for Estonian on Belebele with three examples and using Llama 2 70B as the base model. However, these gains are inconsistent and smaller compared to the improvements achieved by using a stronger base model. For instance, the zero-shot performance for Estonian on Belebele with Llama 3.1 70B surpasses the 3-shot performance of Llama 2 70B. This highlights the greater potential of stronger base models over prompt engineering the weaker models.

5.3 The Effect of CoT Prompting

We analyze the impact of CoT prompting across three tasks: Belebele, QA, and XCOPA. Due to the significant increase in the input length with additional examples, we only compare one-shot prompting with one-shot CoT prompting for Bele-

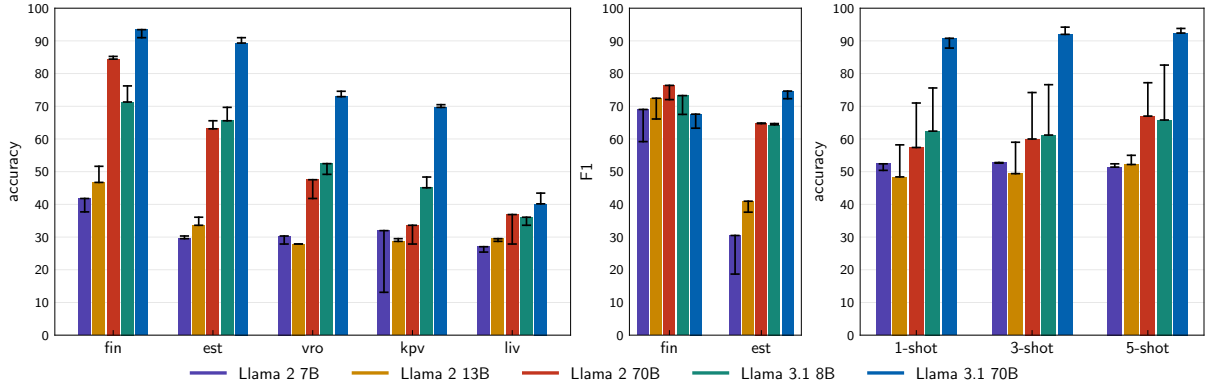


Figure 3: Comparison of CoT prompting and few-shot prompting on Belebele (left, 1-shot), QA (middle, 1-shot) and XCOPA (right, 1-, 3- and 5-shot). The bars shows the scores with few-shot prompting. Horizontal line (–) indicates the score with few-shot CoT prompting with the same number of shots.

bele and QA. For XCOPA we consider 1-, 3-, and 5-shot scenarios.

Figure 4 shows the results. In Belebele task, Llama 2 13B, Llama 2 70B and Llama 3.1 8B benefit from CoT prompting in case of Estonian and Finnish. With the same models the effect of CoT prompting to Võro, Livonian and Komi is mostly negative. Llama 2 7B shows negative or minimal positive gains on all languages. This can be explained with the weak translation skills of Llama 2 7B. On the other hand, Llama 3.1 70B has very strong translation skills, yet CoT prompting yields smaller positive improvement than weaker models. This suggests the strong cross-lingual capabilities of Llama 3.1 70B that mitigate the need for CoT prompting.

For the QA task, CoT prompting consistently results in lower performance. This could be attributed to the nature of the extractive QA task, which requires the output to precisely match the correct text snippet. The intermediate translation steps involved in CoT prompting may lead to slight alterations in the morphological form of the answer, causing a mismatch with the expected output.

In XCOPA, we see mostly positive improvements from CoT prompting, with even Llama 2 13B benefiting, while Llama 2 7B does not. The average improvement across all shots for Llama 2 70B and Llama 3.1 8B is 14%. However, the benefit of CoT prompting decreases significantly for Llama 3.1 70B, following the trend observed in the Belebele task.

These observations naturally raise the question of whether there is a correlation between a model’s

translation capability and its ability to benefit from CoT prompting. To answer that question, we plot the 1-shot BLEU scores of FiU → English translation direction against the gains from 1-shot CoT prompting over 1-shot prompting (Figure 4). As shown in the plot, there is no strong correlation between machine translation quality and CoT gains. Interestingly, CoT prompting can provide improvements over few-shot prompting, even for models with weak translation capabilities. However, it also appears that CoT prompting is more likely to degrade performance than enhance it.

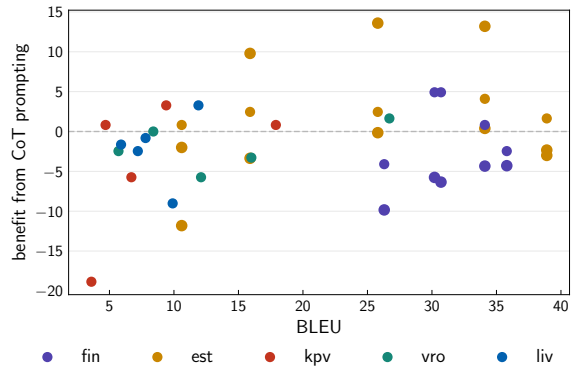


Figure 4: 1-shot BLEU scores for FiU → English translation (x-axis) compared with gains from 1-shot CoT prompting over 1-shot prompting (y-axis). Each dot represents a specific Llama model on a specific task and language. Tasks include Belebele, QA, and XCOPA.

Our findings align with Sprague et al. (2024), whose experiments and extensive meta-analysis of existing studies show that CoT provides significant benefits on tasks involving math and logic but offers much smaller gains for other types of tasks.

	Belebele			FLORES			XCOPA			QA		
	L2	Lam	L3.1	L2	Lam	L3.1	L2	Lam	L3.1	L2	Lam	L3.1
liv	26.23	23.77	37.70	6.76	7.70	10.50	-	-	-	-	-	-
vro	27.05	31.97	50.82	7.83	16.23	16.72	-	-	-	-	-	-
kpv	27.87	24.59	52.46	5.36	3.64	10.32	-	-	-	-	-	-
est	28.69	36.89	68.03	12.65	34.29	35.28	49.20	68.20	69.00	35.35	63.76	70.87
fin	44.26	27.87	74.59	29.63	18.36	31.97	-	-	-	70.63	56.32	75.44
avg	30.82	29.02	56.72	12.44	16.04	20.96	49.20	68.20	69.00	52.99	60.04	73.16

Table 3: Comparison of five-shot results of Llama 2 7B, Llammas-base and Llama 3.1 8B. F1 score is reported for QA.

6 Comparison With Other Models

6.1 Mistral NeMo

We compare Llama 3.1 8B with its competitor, the 12B-parameter model Mistral NeMo (Jiang et al., 2024), across all tasks except SIB. Both models are evaluated in zero-shot and five-shot settings to assess their ability to perform with and without examples. Results for the zero-shot setting are shown in Table 4, while the five-shot results are presented in Table 5. Note that zero-shot results for the QA task are not reported, as this task is typically evaluated in a few-shot setting due to significantly lower performance in zero-shot scenarios.

	Belebele		FLORES		XCOPA	
	L3.1	MN	L3.1	MN	L3.1	MN
liv	33.61	35.25	4.91	5.85	-	-
vro	48.36	50.82	12.19	8.18	-	-
kpv	38.52	36.89	8.18	3.45	-	-
est	62.30	74.59	31.00	33.04	56.80	56.40
fin	68.03	74.59	28.54	30.39	-	-
avg	50.16	54.43	16.96	16.18	56.80	56.40

Table 4: Comparison of zero-shot results of Llama-3.1 8B and Mistral NeMo.

	Belebele		FLORES		XCOPA		QA	
	L3.1	MN	L3.1	MN	L3.1	MN	L3.1	MN
liv	37.70	37.70	10.50	10.10	-	-	-	-
vro	50.82	50.00	16.72	12.55	-	-	-	-
kpv	52.46	34.43	10.32	6.01	-	-	-	-
est	68.03	83.61	35.28	32.28	69.20	71.60	70.87	71.86
fin	74.59	78.69	31.97	33.24	-	-	75.44	77.39
avg	56.72	56.89	20.96	18.83	69.20	71.60	73.16	74.63

Table 5: Comparison of five-shot results of Llama-3.1 8B and Mistral NeMo. F1 score is reported for QA.

The results show that Mistral NeMo and Llama

3.1 8B perform similarly on FiU languages in the zero-shot setting, though Mistral NeMo is over 4% better on the Belebele task. In the five-shot setting, Mistral NeMo outperforms Llama 3.1 8B on three out of four tasks, except for machine translation, where Llama 3.1 8B demonstrates a stronger ability to learn from examples. Overall, Mistral NeMo excels in Finnish and Estonian, while Llama 3.1 8B appears slightly stronger in extremely low-resource FiU languages. Notably, Llama 3.1 8B consistently outperforms Mistral NeMo in Komi, which, unlike the other languages, uses the Cyrillic script.

	Belebele			FLORES			XCOPA		
	L2	Lam	L3.1	L2	Lam	L3.1	L2	Lam	L3.1
liv	24.59	38.52	33.61	4.74	4.62	4.91	-	-	-
vro	23.77	33.61	48.36	4.61	9.92	12.19	-	-	-
kpv	26.23	29.51	38.52	2.88	1.44	8.18	-	-	-
est	22.95	39.34	62.30	8.53	28.90	31.0	48.80	56.60	56.60
fin	32.79	34.43	68.03	27.16	11.57	28.54	-	-	-
avg	26.07	35.08	50.16	9.59	11.29	16.96	48.80	56.60	56.60

Table 6: Comparison of zero-shot results of Llama 2 7B, Llammas-base and Llama 3.1 8B.

6.2 Llammas

We compare Llama 2 7B with Llammas (Kuulmets et al., 2024), which is an adaptation of Llama 2 7B to Estonian with additional pretraining of 5B tokens of Estonian-centric data. We also include comparative size Llama 2.1 8B in this comparison. The results are presented in Table 6 and Table 3.

Unsurprisingly, Llammas outperforms Llama 2 7B on Estonian by a significant margin; however, its performance on Finnish, in general, decreases substantially. As indicated in the tables presented in Section 5.1, Llama 2 7B already demonstrates some capability in solving tasks in Finnish, unlike

in other FiU languages. This suggests that continued pretraining on Estonian notably damages this capability.

Llammas consistently outperforms Llama 2 7B on Võro, which is not surprising given the linguistic similarities between Võro and Estonian. The comparison between Livonian and Komi is less clear in determining which model performs better. However, Llama 3.1 8B surpasses both models by a large margin, except on the Belebele task in Livonian. Notably, Llama 3.1 8B outperforms Llammas even on Estonian, demonstrating that language-specific adaptation of a weaker base model cannot compete with a stronger, unadapted base model.

7 Conclusion

We evaluated the Llama 2 and multilingual Llama 3.1 family models on five Finno-Ugric languages with varying amounts of available resources. Our results show that Llama 2 7B and 13B perform poorly on most languages, except for Finnish, where they achieve moderate results. In contrast, the Llama 3.1 family models demonstrate impressive performance, even for extremely low-resource languages like Võro and Komi.

The comparison of zero-shot and few-shot prompting indicates that few-shot prompting is beneficial across all languages. However, increasing the number of examples does not always lead to better performance. Similarly, few-shot CoT prompting brings substantial benefits for tasks like commonsense reasoning but negatively affects others, such as QA. Notably, the strongest model, Llama 3.1 70B, benefits less from CoT prompting on tasks where it helps weaker models, suggesting that strong cross-lingual capabilities reduce reliance on CoT prompting.

Outstanding results in MT, XCOPA, and Belebele for Estonian and Finnish highlight the need for stronger benchmarks to better assess the capabilities and limitations of these models. The surprisingly strong results from Llama 3.1 70B on Komi and Võro, despite extremely limited resources, demonstrate effective cross-lingual knowledge transfer and reduce the dependence on large target-language datasets for reasonable performance.

Finally, our comparison with Mistral NeMo suggests that the latter outperforms Llama 3.1 8B in Estonian and Finnish. Furthermore, our analy-

sis of Llama models versus Llammas shows that a stronger, general-purpose base model consistently outperforms a weaker base model adapted to a specific language, emphasizing the critical role of the base model in successful language adaptation.

Acknowledgements

This work was partially supported by the Estonian Research Council grant PRG2006 as well as the National Programme of Estonian Language Technology grant EKTB104. All computations were performed in the High Performance Computing Center of the University of Tartu.

References

- David Adelani, Hannah Liu, Xiaoyu Shen, Nikita Vassilyev, Jesujoba Alabi, Yanke Mao, Haonan Gao, and En-Shiun Lee. 2024. SIB-200: A simple, inclusive, and big evaluation dataset for topic classification in 200+ languages and dialects. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 226–245, St. Julian’s, Malta. Association for Computational Linguistics.
- Duarte M. Alves, José Pombal, Nuno M. Guerreiro, Pedro H. Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G. C. de Souza, and André F. T. Martins. 2024. Tower: An open multilingual large language model for translation-related tasks.
- Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2024. The belebele benchmark: a parallel reading comprehension dataset in 122 language variants. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775, Bangkok, Thailand. Association for Computational Linguistics.
- Pierpaolo Basile, Elio Musacchio, Marco Polignano, Lucia Siciliani, Giuseppe Fiameni, and Giovanni Semeraro. 2023. Llamantino: Llama 2 models for effective text generation in italian language.
- 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.
- Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. Tydi qa: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*.
- Yiming Cui, Ziqing Yang, and Xin Yao. 2024. Efficient and effective text encoding for chinese llama and alpaca.
- George-Andrei Dima, Andrei-Marius Avram, Cristian-George Craciun, and Dumitru-Clementin Cercel. 2024. RoQLlama: A lightweight Romanian adapted language model. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4531–4541, Miami, Florida, USA. Association for Computational Linguistics.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnston, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kam-badur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chat-terji, Olivier Duchenne, Onur Çelebi, Patrick Al-rassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sa-hana Chennabasappa, Sanjay Singh, Sean Bell, Seo-hyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Rapparthi, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collet, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vig-nesh Ramanathan, Viktor Kerkez, Vincent Gouget, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyan Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuwei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwon Song, Yuchen Zhang, Yue Li, Yun-ing Mao, Zacharie Delphierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boes-steinberg, Alex Vaughan, Alexei Baevski, Allie Fein-stein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yaz-dan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhar-gavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Sto-jkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkan Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanaz-

- eri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damla, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabza, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vitor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojuan Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. 2024. The llama 3 herd of models.
- Julen Etxaniz, Oscar Sainz, Naiara Miguel, Itziar Aldabe, German Rigau, Eneko Agirre, Aitor Ormazabal, Mikel Artetxe, and Aitor Soroa. 2024. Latxa: An open language model and evaluation suite for Basque. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14952–14972, Bangkok, Thailand. Association for Computational Linguistics.
- Kazuki Fujii, Taishi Nakamura, Mengsay Loem, Hiroki Iida, Masanari Ohi, Kakeru Hattori, Hirai Shota, Sakae Mizuki, Rio Yokota, and Naoaki Okazaki. 2024. Continual pre-training for cross-lingual llm adaptation: Enhancing japanese language capabilities.
- Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac’h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024. A framework for few-shot language model evaluation.
- Carolyn Holtermann, Paul Röttger, Timm Dill, and Anne Lauscher. 2024. Evaluating the elementary multilingual capabilities of large language models with MultiQ. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 4476–4494, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.
- Haoyang Huang, Tianyi Tang, Dongdong Zhang, Xin Zhao, Ting Song, Yan Xia, and Furu Wei. 2023. Not all languages are created equal in LLMs: Improving multilingual capability by cross-lingual-thought prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12365–12394, Singapore. Association for Computational Linguistics.
- Albert Jiang, Alexandre Sablayrolles, Alexis Tacnet, Alok Kothari, Antoine Roux, Arthur Mensch, Audrey Herblin-Stoop, Augustin Garreau, Austin Birky, Bam4d, Baptiste Bout, Baudouin de Monicault, Blanche Savary, Carole Rambaud, Caroline Feldman, Devendra Singh Chaplot, Diego de las Casas, Eleonore Arcelin, Emma Bou Hanna, Etienne Metzger, Gaspard Blanchet, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Harizo Rajaona, Henri Roussez, Hichem Sattouf, Ian Mack, Jean-Malo Delignon, Jessica Chudnovsky, Justus Murke, Kartik Khandelwal, Lawrence Stewart, Louis Martin, Louis TERNON, Lucile Saulnier, L  lio Renard Lavaud, Margaret Jennings, Marie Pellet, Marie Torelli, Marie-Anne Lachaux, Marjorie Janiewicz, Micka  l Seznec, Nicolas Sch  hl, Niklas Muhs, Olivier de Garrigues, Patrick von Platen,

- Paul Jacob, Pauline Buche, Pavan Kumar Reddy, Perry Savas, Pierre Stock, Romain Sauvestre, Sagar Vaze, Sandeep Subramanian, Saurabh Garg, Sophia Yang, Szymon Antoniak, Teven Le Scao, Thibault Schueller, Thibaut Lavril, Thomas Wang, Théophile Gervet, Timothée Lacroix, Valera Nemychnikova, Wendy Shang, William El Sayed, and William Marshall. 2024. Mistral-nemo-base-2407.
- Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213. Curran Associates, Inc.
- Hele-Andra Kuulmets, Taïdo Purason, Agnes Luhtaru, and Mark Fishel. 2024. Teaching llama a new language through cross-lingual knowledge transfer. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3309–3325, Mexico City, Mexico. Association for Computational Linguistics.
- Anu K  ver. 2021. Extractive question answering for estonian language. Master’s thesis, Tallinn University of Technology (TalTech).
- Bill Yuchen Lin, Abhilasha Ravichander, Ximing Lu, Nouha Dziri, Melanie Sclar, Khyathi Chandu, Chandra Bhagavatula, and Yejin Choi. 2023. The unlocking spell on base llms: Rethinking alignment via in-context learning.
- Peiqin Lin, Shaoxiong Ji, J  rg Tiedemann, Andr   F. T. Martins, and Hinrich Sch  tze. 2024. Mala-500: Massive language adaptation of large language models.
- Risto Luukkonen, Ville Komulainen, Jouni Luoma, Anni Eskelinen, Jenna Kanerva, Hanna-Mari Kupari, Filip Ginter, Veronika Laippala, Niklas Muennighoff, Aleksandra Piktus, Thomas Wang, Noumane Tazi, Teven Scao, Thomas Wolf, Osm   Suominen, Samuli Sairanen, Mikko Merioksa, Jyrki Heinenen, Aija Vahtola, Samuel Antao, and Sampo Pyys  lo. 2023. FinGPT: Large generative models for a small language. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2710–2726, Singapore. Association for Computational Linguistics.
- James Cross Onur   lebi Maha Elbayad Kenneth Heafield Kevin Heffernan Elahe Kalbassi Janice Lam Daniel Licht Jean Maillard Anna Sun Skyler Wang Guillaume Wenzek Al Youngblood Bapi Akula Loic Barrault Gabriel Mejia Gonzalez Prangthip Hansanti John Hoffman Semarley Jarrett Kaushik Ram Sadagopan Dirk Rowe Shannon Spruit Chau Tran Pierre Andrews Necip Fazil Ayan Shruti Bhosale Sergey Edunov Angela Fan Cynthia Gao Vedanuj Goswami Francisco Guzm  n Philipp Koehn Alexandre Mourachko Christophe Ropers Safiyyah Saleem Holger Schwenk Jeff Wang NLLB Team, Marta R. Costa-juss  . 2022. No language left behind: Scaling human-centered machine translation.
- OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, 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, Sim  n 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,   ukasz 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,   ukasz 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 M  ly, Ashvin Nair, Reiichiro Nakano, Rameev 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, Poko-

- rny, 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.
- Ramon Pires, Hugo Abonizio, Thales Sales Almeida, and Rodrigo Nogueira. 2023. *Sabiá: Portuguese Large Language Models*, page 226–240. Springer Nature Switzerland.
- Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376, Online. Association for Computational Linguistics.
- Taido Purason, Hele-Andra Kuulmets, and Mark Fishel. 2024. LLMs for extremely low-resource finno-ugric languages.
- Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2022. Language models are multilingual chain-of-thought reasoners.
- Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. 2024. To cot or not to cot? chain-of-thought helps mainly on math and symbolic reasoning.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shrutli 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.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models.
- Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in English? on the latent language of multilingual transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394, Bangkok, Thailand. Association for Computational Linguistics.
- Lisa Yankovskaya, Maali Tars, Andre Tättar, and Mark Fishel. 2023. Machine translation for low-resource finno-ugric languages. In *The 24rd Nordic Conference on 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*.
- Xiang Zhang, Senyu Li, Bradley Hauer, Ning Shi, and Grzegorz Kondrak. 2023. Don’t trust ChatGPT when your question is not in English: A study of multilingual abilities and types of LLMs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7915–7927, Singapore. Association for Computational Linguistics.
- Zhihan Zhang, Dong-Ho Lee, Yuwei Fang, Wenhao Yu, Mengzhao Jia, Meng Jiang, and Francesco Barbieri. 2024. PLUG: Leveraging pivot language in cross-lingual instruction tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7025–7046, Bangkok, Thailand. Association for Computational Linguistics.
- Chunting Zhou, Pengfei Liu, Puxin Xu, Srinu 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.

Wenhao Zhu, Shujian Huang, Fei Yuan, Shuaijie She, Jiajun Chen, and Alexandra Birch. 2024. Question translation training for better multilingual reasoning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 8411–8423, Bangkok, Thailand. Association for Computational Linguistics.

Mapping Faroese in the Multilingual Representation Space: Insights for ASR Model Optimization

Dávid í Lág

University of the Faroe Islands
J. C. Svabosgøta 14,
100 Tórshavn
davidl@setur.fo

Barbara Scalvini

University of the Faroe Islands
J. C. Svabosgøta 14,
100 Tórshavn
barbaras@setur.fo

Jón Gunason

Reykjavik University
Menntavegur 1
101 Reykjavik
jb@ru.is

Abstract

ASR development for low-resource languages such as Faroese faces significant challenges due to the scarcity of large, diverse datasets. Although fine-tuning multilingual models using related languages is common practice, there is no standardized method for selecting these auxiliary languages, leading to a computationally expensive trial-and-error process. By analyzing the positioning of Faroese among other languages in wav2vec2’s multilingual representation space, we find that Faroese’s closest neighbors are influenced not only by linguistic similarity but also by historical, phonetic, and cultural factors. These findings open new avenues for auxiliary language selection to improve Faroese ASR and underscore the potential value of data-driven factors in ASR fine-tuning.

1 Introduction

Low-resource languages, such as Faroese, face unique challenges in ASR development, primarily due to the lack of sufficiently large and varied datasets. Recent advances in multilingual ASR models have provided a promising avenue for cross-linguistic transfer, leveraging similarities between languages to enhance the performance of those with limited resources. It is common practice to fine-tune multilingual models for a target language by incorporating similar, closely related languages (Juan et al., 2014; Juan, 2015; Ivan Froiz-Míguez, 2023). However, currently there is no standardized procedure for selecting these languages. ASR researchers often train multiple models with different language combinations to find the best set to enhance target language performance, a trial-and-error approach that is computationally costly as models grow larger. This

underscores the need for more efficient methods. In this study, we focus on Faroese, a low-resource Insular Scandinavian language. We explore its representation in Meta’s wav2vec2 XLSR 53 model (Alexis Conneau, 2020), and seek out its neighbors in this space, with the aim of extracting new insight for selection of auxiliary languages. Our approach analyzes how languages are encoded within the model’s multilingual representation space by measuring the distance between Faroese and 102 languages from the Google Fleurs dataset (Alexis Conneau, 2022) at each model layer. Since Faroese is absent from Google Fleurs, we incorporated recordings from the Ravnursson data set (Hernández Mena and Simonsen, 2022), currently the only ASR-suitable Faroese dataset, to better understand how the model perceives Faroese in relation to other languages and to improve multilingual fine-tuning strategies.

2 Background and related work

2.1 Advances in Transformer and Self-Supervised Models for ASR

In 2019, the wav2vec model was introduced as a self-supervised model that learns speech representations without labeled data and can be fine-tuned for ASR, reducing the need for extensive labeled datasets (A. Baevski and Auli, 2020). While initially trained only on English, later versions support multiple languages (Alexis Conneau, 2020). The architecture of the wav2vec 2.0 model enables cross-lingual transfer in ASR through multilingual quantized speech representations, allowing latent speech units to capture key features of speech (Alexei Baevski, 2020). Transfer learning with related languages has been shown to improve ASR for low-resource languages by leveraging high-dimensional embeddings from the wav2vec2.0 XLSR-53 model (Akbayan Bekarystankyzy, 2024; J. Cho and Hori, 2018; Vishwa Gupta, 2022). Re-

search demonstrates the model’s ability to capture language similarities by clustering embeddings using K-Means (Alexis Conneau, 2020).

2.2 ASR for Faroese

The effort towards digitalization of Faroese speech has led to the creation of a Basic Language Resource Kit for Faroese (A. Simonsen and Henrichsen, 2022) in the context of the Ravnur project.¹ This project involved the collection of both text corpora and audio recordings finalized in the creation of ASR systems. The Ravnur audio data set contains 100 hours of training data, which is a balanced collection of high-quality recordings, including different dialects and speakers of different ages. The availability of such data has allowed researchers to test strategies to produce ASR models for Faroese. One such strategy was the fine-tuning of multilingual models such as wav2vec2, which led to the creation of the very first ASR model specifically targeting Faroese (Hernandez Mena, 2022).

3 Method

3.1 Dataset

To assess the relationship between Faroese and other languages, we used Meta’s wav2vec2 XLS-R 53 Large model² with 25 layers to generate hidden representations for all of the 102 Google Fleurs³ (Alexis Conneau, 2022) languages in addition to Faroese. The model is trained on 56k hours of speech data for 53 languages. Of the Scandinavian languages, only Swedish is included in the model. We performed inference with the model using the same number of sentences per language in the Google Fleurs dataset for the 102 languages. Faroese is not in Google Fleurs, and therefore we instead take 900 random sentences from the Ravnursson ASR corpus⁴.

3.2 Distance calculation

We calculate the distance between Faroese and 102 other languages in the hidden representation space of wav2vec 2.0, analyzing across different

layers. The pipeline for the distance calculation can be summarized as follows. First, we obtain a sentence-level representation by applying average pooling to all hidden representations across the sentence. Then, we compute the overall representation by averaging the sentence-level representations for all sentences for each language l and layer j ,

$$\mu_{l,j} = \frac{1}{N} \sum_{i=1}^N R_{l,i,j}, \quad (1)$$

where $R_{l,i,j}$ is the representation vector for sentence s_{li} at layer j . $S_l = s_{l1}, s_{l2}, \dots, s_{lN}$ is a set of $N = 900$ sentences for language $l \in L$ where L is a set of languages with $|L| = 103$. The layer index is $j = 0, 1, \dots, 24$.

3.3 Clustering and visualization

K-means clustering was used on the computed representations after performing dimensionality reduction using Principal Component Analysis (PCA) (Jolliffe, 2002), t-distributed stochastic neighbor embedding (t-SNE) (T. Tony Cai, 2021) and Uniform Manifold Approximation and Projection (UMAP) (Leland McInnes, 2018). Each layer in the wav2vec2 XLS-R 53 model contributes to the model’s overall functionality. Ankita Pasad (2021) explored which type of speech information is predominantly encoded in each of the 25 layers of the wav2vec2 model, in terms of local acoustic features, phone identity, word identity, and word meaning. We take inspiration from their results and identify three main layer groupings:

- Layers 1 to 11: The first few layer representations (0-5) are dominated by local acoustic features, which gradually decrease, leaving gradually room for language-specific features such as phone and word identity.
- Layers 12 to 19: In these layers, word identity and word meaning dominate the representations, capturing more abstract linguistic features essential for understanding syntax and semantics. There is a sharp decrease in phone identity representation around layer 15, followed by a sharp increase.
- Layers 20 to 24: We observe an overall decrease in all linguistic properties, with phone identity, however, remaining more prominent than the other characteristics.

¹<https://mtd.setur.fo/en/resource/ravnur-blark-1-0/>

²<https://huggingface.co/facebook/wav2vec2-large-xlsr-53>

³<https://huggingface.co/datasets/google/fleurs>

⁴https://huggingface.co/datasets/carlosdanielhernandezmena/ravnursson_asr

We use this information for interpretations of the results and layer selection during qualitative clustering analysis. Specifically, we will focus on layers 18 - 20, as we expect word identity and phone identity information to be at their highest in these layers.

3.4 Experiments

The key steps involved in our methodology are outlined as follows:

1. **Data selection:** Since Icelandic had the fewest sentences in the Google Fleurs dataset, with 924 sentences, we set the number of sentences per language for the analysis at 900.
2. **Hidden representation extraction:** For each language, we ran inference with the wav2vec2 XLS-R 53 model on the selected 900 sentences, extracting the hidden representation for each of the 25 hidden layers as described in Sec 3.3. We processed the representations as follows:
 - Calculating the mean of all layer-wise 25 hidden representations per language
 - Grouping the layers into intervals of five: 0-4, 5-9, 10-14, 15-19, 20-24, and computing the mean interval representation for each language.
3. **Distance between languages:** To explore the relationships between Faroese and the other languages, we calculate the Euclidean distance in the original representation vector space.
4. **Clustering:** We apply K-Means after reducing dimensions down to 2 using PCA, t-SNE, and UMAP. This choice was made in order to facilitate visualization and qualitative analysis.

4 Results and Discussion

4.1 Quantitative analysis: top nearest neighbors in the representation space for Faroese

For each layer interval, we calculated the Euclidean distance between Faroese and the 102 languages in the Google Fleurs dataset. Table 1 presents the top eight nearest neighbors to Faroese

in descending order for each layer interval. Interesting patterns emerge from these results. The top nearest neighbor across all layer intervals is either Welsh or Irish, with Welsh being the closest when all layers (0–24) are combined. Welsh and Irish belong to the Celtic language family, in contrast to Faroese, which is a Scandinavian language. However, Faroese phonetics is known to have been significantly influenced by contact with Scottish Gaelic-speaking communities from the neighboring British Isles. German ranks as the second closest neighbor in the early layers (0–9), while Scandinavian languages emerge as neighbors in the later layers: Swedish in layers 10–14, and Norwegian in layers 20–24 and overall. Beyond this, the composition of nearest neighbors does not reveal any clear pattern in terms of linguistic families.

4.2 Qualitative analysis: dimensionality reduction and clustering

The internal representation space of multilingual models is highly multidimensional and often challenging to interpret. To clarify the results of our quantitative analysis and provide a visual interpretation of the distances in this space, we performed dimensionality reduction on the combined representation space of layers 18–20. In these layers, we anticipate clustering among languages from the same linguistic families due to shared phonetic, syntactic, or acoustic characteristics. If a language clusters separately from its family, it may indicate unique linguistic traits. Examining outliers and mixed clusters could also uncover cross-family influences or reveal features such as geographic convergence. Figure 1 shows clusters of languages in the same language family for six different regions. Clustering was performed using K-Means following dimensionality reduction to two dimensions. Of the three-dimensionality reduction techniques tested, t-SNE most closely aligned with results from the original high-dimensional space, as shown in Table 2. In this analysis, Irish appears as the closest neighbor to Faroese, with Swedish positioned farther within the neighborhood (see Figure 1). Overall, we observe a representation of Germanic/Scandinavian languages in the clusters (English, German, Luxembourgish, Swedish), along with non-Indo-European languages that are part of the Nordic cultural sphere, such as Finnish.

Layers 0-4	5-9	10-14	15-19	20-24	0-24
1 Irish (10.8)	Irish (13.4)	Irish (13.4)	Irish (16.2)	Welsh (31.7)	Welsh (14.8)
2 German (11.3)	German (15.4)	Estonian (15.4)	Croatian (17.0)	Turkish (34.7)	Turkish (17.5)
3 Romanian (11.6)	Estonian (16.0)	Croatian (15.8)	Estonian (17.4)	Punjabi (47.4)	Punjabi (22.6)
4 Estonian (11.8)	Croatian (16.2)	Lithuanian (15.9)	Lithuanian (17.5)	Slovak (104.0)	Slovak (25.2)
5 Simplified Chinese (11.8)	Romanian (16.2)	Welsh (16.1)	Polish (17.7)	Georgian (110.1)	Georgian (25.8)
6 Catalan (12.0)	English (16.2)	Romanian (16.1)	Georgian (17.9)	Amharic (112.7)	Amharic (27.4)
7 Korean (12.1)	Welsh (16.4)	Polish (16.5)	Romanian (18.0)	Norwegian (126.4)	Norwegian (29.8)
8 Armenian (12.3)	Lithuanian (16.4)	Swedish (16.6)	Slovenian (18.0)	Vietnamese (145.8)	Armenian (32.5)

Table 1: *Closest languages to Faroese measured in Euclidean distance in the original representation vector space*

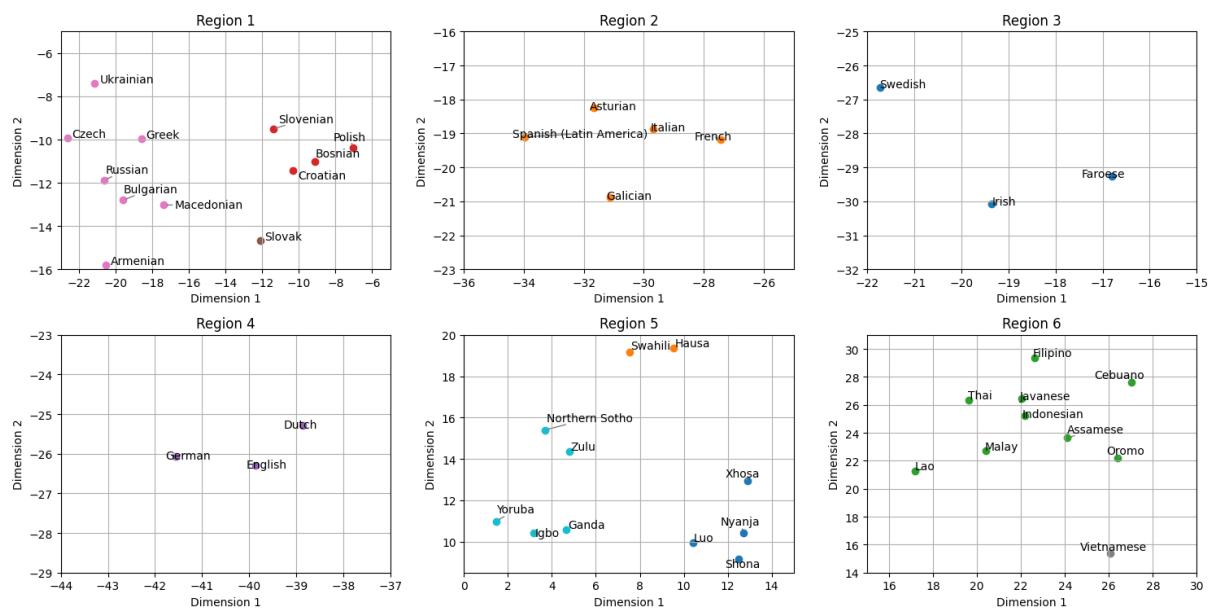


Figure 1: *Clusters of closely related languages for layers 18-20 with t-SNE and K-Means with 18 clusters*

PCA	t-SNE	UMAP
Romanian (1.72)	Irish (2.68)	Croatian (0.42)
French (3.29)	Maori (4.03)	Catalan (0.47)
English (5.51)	Swedish (5.58)	Romanian (0.55)
German (5.64)	Finnish (6.00)	Maori (0.77)
Luxembourgish (9.83)	Latvian (8.14)	Georgian (0.79)

Table 2: *Languages in the same cluster as Faroese in layers 18-20 using K-Means with 18 clusters after dimensional reduction with PCA, t-SNE, and UMAP*

5 Conclusion

In conclusion, the representation spaces in wav2vec2 indicate that languages tend to cluster, as evidenced through nearest-neighbor analy-

sis, clustering, and dimensionality reduction techniques. This analysis places Faroese in proximity to Gaelic languages, alongside Germanic and Nordic languages. The prominence of Gaelic languages as close neighbors suggests that limiting comparisons to only the closest family members may overlook valuable insights, possibly related to historical phonetic and linguistic influences. Such consideration will be further investigated in future work.

6 Limitations

This exploration of the representation of Faroese is based on a single model and may therefore vary with other models, as language representations are influenced by the specific language distribution

within the training data. Additionally, we only evaluated language proximity using one dataset, FLEURS, which may have limited speaker representation. The metric used, Euclidean distance, is just one approach for vector comparison and has its limitations. For instance, it is susceptible to the curse of dimensionality and may not be optimal in highly multidimensional spaces. Alternative metrics, such as cosine similarity, could yield slightly different results. Despite these limitations, our analysis provides a foundation for a more comprehensive characterization of language similarity within model representation spaces, with potential applications in language selection for low-resource ASR training.

References

- A. Mohamed A. Baevski, H. Zhou and M. Auli. 2020. <http://arxiv.org/abs/2006.11477> wav2vec 2.0: A framework for self-supervised learning of speech representations.
- I. N. Debess A. Simonsen, S. S. Lamhauge and P. J. Henrichsen. 2022. <https://aclanthology.org/2022.lrec-1.495/> Creating a basic language resource kit for faroese.
- Mateus Mendes Anar Fazylzhanova Muhammad As-sam Akbayan Bekarystankyzy, Orken Mamyrbayev. 2024. <https://www.nature.com/articles/s41598-024-64848-1> Multilingual end-to-end asr for low-resource turkic languages with common alphabets.
- Michael Auli Alexei Baevski, Alexis Conneau. 2020. <https://ai.meta.com/blog/wav2vec-20-learning-the-structure-of-speech-from-raw-audio/> Wav2vec 2.0: Learning the structure of speech from raw audio.
- Ronan Collobert Abdelrahman Mohamed Michael Auli Alexis Conneau, Alexei Baevski. 2020. <https://arxiv.org/abs/2006.13979> Unsupervised cross-lingual representation learning for speech recognition.
- Simran Khanuja Yu Zhang Vera Axelrod Sid-dharth Dalmia Jason Riesa Clara Rivera Ankur Bapna Alexis Conneau, Min Ma. 2022. <https://arxiv.org/abs/2205.12446> Fleurs: Few-shot learning evaluation of universal representations of speech.
- Karen Livescu Ankita Pasad, Ju-Chieh Chou. 2021. <https://arxiv.org/abs/2107.04734> Layer-wise analysis of a self-supervised speech representation model.
- Carlos Daniel Hernandez Mena. 2022. <https://huggingface.co/carlosdanielhernandezmena/wav2vec2-large-xlsr-53-faroese-100h> Acoustic model in faroese: wav2vec2-large-xlsr-53-faroese-100h.
- Carlos Daniel Hernández Mena and Annika Simon-sen. 2022. <http://hdl.handle.net/20.500.12537/276> Ravnursson faroese speech and transcripts.
- Paula Fraga-Lamas Diego Fustes Carlos Dafonte Javier Pereira Tiago M. Fernandez-Carames Ivan Froiz-Miguez, Oscar Blanco-Novoa. 2023. <https://doi.org/10.29007/1ppr> Design and evaluation of a cross-lingual ml-based automatic speech recognition system fine-tuned for the galician language. *Kalpa publications in computing*.
- R. Li M. Wiesner S. H. Mallidi N. Yalta M. Karafiat S. Watanabe J. Cho, M. K. Baskar and T. Hori. 2018. <https://arxiv.org/abs/1810.03459> Multilingual sequence-to-sequence speech recognition: architecture, transfer learning, and language modeling. 2018 *IEEE Spoken Language Technology Workshop (SLT)*, arXiv:1810.03459.
- Ian T Jolliffe. 2002. *Principal component analysis for special types of data*. Springer.
- Sarah Flora Samson Juan. 2015. <https://api.semanticscholar.org/CorpusID:33165732> Exploiting resources from closely-related languages for automatic speech recognition in low-resource languages from malaysia.
- Sarah Flora Samson Juan, Laurent Besacier, Benjamin Lecouteux, and Tien Ping Tan. 2014. <https://api.semanticscholar.org/CorpusID:8620301> Using closely-related language to build an asr for a very under-resourced language: Iban. 2014 *17th Oriental Chapter of the International Committee for the Co-ordination and Standardization of Speech Databases and Assessment Techniques (COCOSDA)*, pages 1–5.
- James Melville Leland McInnes, John Healy. 2018. <https://arxiv.org/abs/1802.03426> Umap: Uniform manifold approximation and projection for dimension reduction.
- Rong Ma T. Tony Cai. 2021. <https://arxiv.org/abs/2105.07536> Theoretical foundations of t-sne for visualizing high-dimensional clustered data.
- Gilles Boulianne Vishwa Gupta. 2022. <https://aclanthology.org/2022.lrec-1.689.pdf> Progress in multilingual speech recognition for low resource languages kurmanji kurkish, cree and inuktut.

Towards a Derivational Semantics Resource for Latvian

Ilze Lokmane, Mikus Grasmanis, Agute Klints, Gunta Nešpore-Bērzkalne,
Pēteris Paikens, Lauma Pretkalniņa, Laura Rituma, Madara Stāde, Evelīna Tauriņa

Institute of Mathematics and Computer Science

University of Latvia

Raiņa bulvāris 29, Riga, Latvia

ilze.lokmane@lu.lv, (mikus.grasmanis, agute.klints,
gunta.nespore, peteris.paikens, lauma.pretkalnina,
laura.rituma, madara.stade)@lumii.lv, taurina.evelina@gmail.com

Abstract

In this paper, we describe the implementation of the first structured resource of semantic derivational links for Latvian, basing it on the largest online dictionary Tēzaurs.lv and linking it to the Latvian WordNet. We separate two kinds of derivational links: semantic derivation links between senses and morphological derivation links between lexemes. Semantic links between senses are defined as a pair of semantic labels assigned to both ends of the link. The process of semantic linking involves revising the sense inventory of both the base word and the derivative, defining semantic labels for lexemes of four basic word classes – nouns, verbs, adjectives, and adverbs, and adding the appropriate labels to the corresponding senses. We exemplify our findings with a detailed representation of the sense relations between a base verb and its nominal derivatives.

Keywords: morphosemantic relations, derivational semantics, polysemous words, WordNet, Latvian

1 Introduction

So far, no derivational semantics resource has been created for the Latvian language. The idea for its creation grew out of the desire to extend the Latvian WordNet (Paikens et al., 2023) because regular derivatives are an essential part of the lexicon, and they also have semantic relations both with their base words and with each other, for example, two derivatives can be synonyms. Latvian WordNet is planned to be supplemented with derivational links, similar to what Princeton WordNet (Mititelu et al., 2021) and others have imple-

mented (e.g. Turkish (Bilgin et al., 2004), Bulgarian (Dimitrova et al., 2014), Romanian (Mititelu, 2012), Czech (Rambousek et al., 2018), Polish (Piasecki et al., 2012)). We consider derivational semantics resources relevant for NLP applications because the behavior of current large language model chat agents for less resourced languages like Latvian shows a misunderstanding of meaning of derived words, so the application of lexical resources has value even in the era of large pre-trained models.

Latvian WordNet has been developed manually for the past four years (Paikens et al., 2023). As of Autumn 2024, Latvian WordNet contains 8756 synsets which cover the meanings of the 2000 most frequently used words in The Balanced Corpus of Modern Latvian (Levāne-Petrova and Dargis, 2018) and their related synsets. The inventory of words and senses is based on the Tēzaurs.lv online dictionary (Spektors et al., 2023; Grasmanis et al., 2023), which is a large (approximately 405000 entries in the last release in September 2024) digital compilation of legacy dictionaries. Latvian WordNet is developed and maintained on the Tēzaurs.lv lexicographic platform, and the data are available in dictionary entries of words whose senses are included in WordNet. This lexical resource also contains links between Latvian WordNet and Princeton WordNet (Fellbaum, 1998). The important thing is that the Latvian WordNet is created between separate word senses, and we also want to create the semantics of derivatives separately for each word sense, so we think these resources will be well integrated. Currently, the semantics of derivatives is a network parallel to Latvian WordNet, the word sense inventory being the unifying element which is involved in both networks. In the future, that will help to integrate one resource into the other.

Up until now, according to the traditions of lexicography, the regular derivatives listed in the

Tēzaurs.lv dictionary had their own entries only if they had a specific sense which was far removed from the senses of the base word. In order to represent the diversity of derivational relations, we are currently creating new entries for the most frequently used regular derivatives.

In Latvian linguistics little or no attention has been paid to semantic relations between the senses of a polysemous base word and the senses of its derivatives, as only general semantics of derivational formatives has been studied and described referring to the basic sense of the base word (Kalnača and Lokmane, 2021; Soida, 2009). In order to improve Tēzaurs.lv and Latvian WordNet, it should be verified whether these relations exist between all senses of the base word and the derivative (in more detail in Chapter 3.3). Therefore, we have chosen to employ two kinds of derivational links: morphological derivation links between lexemes and semantic derivation links between exact word senses (described in more detail in Chapters 2 and 3). A morphological derivation link contains information about the formatives used in word formation, while a semantic link is formed as a pair of semantic labels that describe both linked senses.

The choice of word pairs for annotating is determined by their frequency of use in The Balanced Corpus of Modern Latvian (Levāne-Petrova and Dargis, 2018). First, the derivatives of the most frequently used verbs, which are already included in Latvian WordNet, are marked to enrich the lexical information of these words as much as possible. Second, the most frequently used derivations in each derivation group are selected, for example, the most frequently used adjectives derived from nouns. The following word pairs are annotated in this phase of the project: a) verbs – deverbal nouns, b) nouns – denominal verbs, c) nouns – denominal adjectives, d) adjectives – deadjectival adverbs. Such groups were chosen to cover the four main word classes of the Latvian language involved in word formation processes. Other patterns of derivational links will be annotated as the project progresses, including patterns when a derivative is of the same word class as the base word. The processed data set currently includes 1000 morphological links and 1600 semantic links.

To ensure a reliable resource for future research, the dataset is developed manually. However, we assume that in future some semi-automatic meth-

ods could also be applied to unambiguous words to ensure a larger coverage, which is essential for NLP applications of this dataset.

2 Morphological Derivational Links

A morphological derivation link between lexemes connects the base word entry to the derived word entry. This link contains two attributes: a derivational stem base and a derivational formative. The stem indicates which part and form of the base word the derivative is formed from. The formative is the means by which a new word is made; it can be a single morpheme, such as a prefix or a suffix, or a combination of morphemes, such as a suffix and an ending, that together form a complex formative. For example, the noun *skrējējs* ‘runner’ is formed by adding formatives *-ēj-* and *-s* to the past tense stem of the verb *skriet* ‘to run’; and the adjective *mākoņains* ‘cloudy’ is formed by adding formatives *-ain-* and *-s* to the plural stem of the noun *mākonis* ‘cloud’.

Since the Latvian language has an extremely rich inflectional and derivational morphology (Kalnača and Lokmane, 2021), new words can be made from various stems, e.g., the present, past, infinitive or participle stems of verbs and singular or plural stems of nouns, using prefixes, suffixes, endings, and interfixes. Therefore, information about the derivational stem seems to be crucial in describing Latvian derivational morphology.

In addition, this information will help in further studies regarding the semantic properties that derivatives obtain with certain derivative formatives. Although Latvian grammars (e.g., (Kalnača and Lokmane, 2021; Soida, 2009)) provide general information of the semantic aspects of such formatives, wider language material could potentially lead to new insights, assist in determining previously undescribed peculiarities of derivative senses, and specify derivational stem bases.

However, our aim does not include dividing the entire word into morphemes; the internal composition of Latvian words is the objective of another project, “Database of Latvian Morphemes and Derivational Models” (see <https://www.dlmdm.lu.lv>). Instead, we only indicate the morphemes involved in the derivative process.

In most cases, the derivational direction between two words is clear, i.e., the base word and the derivative can be discerned by consulting the already described models of word formation.

However, there are derivational relations in which it is not obvious which of the two is the base word and which is the derivative (e. g., *kontrolēt* ‘to control’ – *kontrolē* ‘control’; *spēlēt* ‘to play (a game)’ – *spēle* ‘a game’). This problem arises mainly (but not exclusively) in pairs of loan words where it is not possible to establish which of the words was introduced into Latvian first; this means that both derivational paths are possible in such cases, as both models of word formation are possible in Latvian. A noun can be derived from a verb (e.g., *atsaukties* ‘to refer’ – *atsauce* ‘a reference’; *aizstāvēt* ‘to defend’ – *aizstāvis* ‘a defender’), and a verb can be derived from a noun (e.g., *skaips* ‘Skype’ – *skaipot* ‘to communicate via Skype’, *balva* ‘an award’ – *apbalvot* ‘to reward’). There are also more recent loan word pairs that are clearly derivationally linked, but are probably not derived from each other (e.g., *bioloģija* ‘biology’ – *bioloģisks* ‘biological’; *demokrātija* ‘democracy’ – *demokrātisks* ‘democratic’). In such instances, the solution is to label the link between lexemes as ‘derivationally related’ without specifying which is the base word and which is the derivative; information on the stem base and the formatives is also not provided.

3 Semantic Derivational Links

Due to the fact that semantic relations between the senses of a polysemous base word and the senses of its derivatives are yet to be studied in depth in Latvian linguistics, a new system for annotating such instances had to be devised. This chapter describes the process of preparing entries for linking, creating semantic derivation links between the senses of the base word and its derivatives, semantic labels for each word class combination and more detailed observations of the relations between the senses of polysemous words.

3.1 Revising the Senses of the Base Word and the Derivative

First step for derivational link creation is revising dictionary entries and word senses. The Tēzaurs.lv entries come from various dictionaries, therefore, the criteria for dividing meanings may vary across different entries. We strive to standardize them according to the current criteria for distinguishing senses in the Tēzaurs.lv (see (Lokmane et al., 2021)) and based on the current situation in the

language.

Derivatives mostly do not have entries in the Tēzaurs.lv because regular derivatives have not been included in the dictionary until now. Therefore, they need to be created anew. We strive to align the derivative’s entry with the entry of the base word (sequence of senses, their granularity), but we try to not create “artificial” meanings for derivatives just to align the entry symmetrically with the base word entry. The verification of the sense is based on corpora data mentioned below. If the word is used in corpora in a particular sense, the sense has to be created and added to the word entry.

Usage examples from several corpora of the Latvian National Corpora Collection (Saulīte et al., 2022) are added to the senses of base words and derivatives (examples must be short, clear, of simple syntactic constructions, in examples the word appears in various constructions). The examples also guide the creation and distinction of senses – if in many examples it is not possible to determine in which meaning the word is used, the division of senses should be reconsidered. We add several examples for each sense, but one example is enough to conclude that the sense is being used, therefore it is relevant to entry.

3.2 Semantic Labels

Semantic links between senses are formed as a pair of semantic labels, which are given to both ends of the link. It seems important to record not only the semantics of the derivative, as most grammars do, but also the semantic characteristics of the base word. For example, the sense ‘to be lying down’ of the base verb *gulēt* labeled as *toBe-InState* is linked to the sense ‘sleeping place’ of the derived noun *guļa* labeled as *location*. Similarly, the sense ‘group’ of the base noun *kopa* labeled as *abstract notion* is linked to the sense ‘used by several or many’ of the derived adjective *kopējs* labeled as *related to*. Such an approach will allow future studies of word-formation processes not only from the perspective of the derivative, but also from the perspective of the base word.

Each of the four word classes discussed so far has a different number of semantic labels (see Table 1). Choosing and defining semantic labels is a labor-intensive process, because there are no ready-made samples that can be used without improvements. It should also be emphasized that

Word class	Semantic label	Description
verb	toBeInProcess toBeInState toDo	to undergo a change of a condition or a state to experience a state or a condition to perform an action
noun	abstract notion action agent animal body part cause device experiencer feature instrument location member of a profession mythical creature natural phenomenon patient person physical phenomenon process resource result state thing time (noun)	a non-concrete concept or idea something that the verb argument does or performs participant who initiates and carries out an action a living being except humans any part of an organism such as an organ or extremity the non-volitional causer of the event an object or machine used to perform an action participant experiencing some state or process property of an entity the entity that is manipulated by the agent and with which an action is performed the place in which something is situated or takes place a person who works in a specified professional activity a supernatural creature that does not exist in real life a physical event that occurs in atmosphere or on the ground participant undergoing the effect of some action a human being a natural phenomenon involving the physics of matter and energy a change in condition or state of the argument the entity by which an action is performed and which is used up during the action entity that comes into existence through the event the state or condition of the argument an inanimate material object the period or moment during which something exists or continues
adjective	evaluative property including measurable possessing similar to related to	based on or relating to an assessment expressing a general property like colour, shape etc. including the entity named by base word expressing a measurable property possessing the entity named by base word similar to the entity named by base word related to the entity named by base word
adverb	degree frequency manner place time (adv.)	specifying the degree to which a property applies describing how often something happens describing how something happens describing location in which something is situated or takes place describing when or how long something takes place

Table 1: Semantic labels for senses linked by a relation

the list of labels can be linguistically specific, although some labels are, of course, universal (Mititelu et al., 2021). Thus, the selection of semantic labels takes into account the experience of creat-

ing electronic resources of other languages (Bilgin et al., 2004; Piasecki et al., 2012) and linguistic studies of both word class semantics and derivational semantics (Wierzbicka, 1988; Raskin and

Nirenburg, 1995; Soida, 2009; Kalnača and Lokmane, 2021).

The noun has most semantic labels. Firstly, this is due to the fact that nouns are included in several pairs of derivationally linked word classes – as derivatives of verbs and as base words of both denominal verbs and adjectives. Secondly, the semantics of nouns are generally more specific and easier to classify than, for example, the semantics of adjectives (Wierzbicka, 1988). Verbs have only three semantic labels despite being both base words and derivatives in relation to nouns. However, the list of semantic labels is being constantly enriched as we proceed with new lexical groups. In the future, there might be a need for a more detailed semantic division of verbs, e.g., names of motion, communication, cognition etc. Adverbs have been assigned five semantic labels traditionally described in grammars.

One of the most difficult problems so far has been the semantic classification of adjectives, since they, being attributes, derive at least part of their semantics from the noun they are attached to. We have chosen to assign three rather general semantic labels to qualitative (descriptive) adjectives and four semantic labels to relational (denominal) adjectives (for a similar solution, see (Raskin and Nirenburg, 1995)). Qualitative adjectives are morphologically simpler than relational ones. The latter, being more complex formally, derive their semantics from their base words.

In each pair of word classes considered so far, the set of semantic labels is different, to best capture the specific semantics of derivative in relation to the base word.

3.3 Relations between Senses of Polysemous Words

Even within the boundaries of one word and its derivatives, there can be a large variety of semantic relations between them, especially when all the senses of a word are considered. This is exemplified by the verb *atgādināt*, which has 4 senses and 4 noun derivatives (see Figure 1).

The first sense, *atgādināt*₁ ‘to prompt, to remind, to cue (something forgotten or imperfectly learned)’, has two narrower subsenses, *atgādināt*_{1.1} ‘to give a reminder (by device)’ and *atgādināt*_{1.2} ‘to bring back a memory (of something)’. The second sense, *atgādināt*₂ ‘to resemble’ has no subsenses. The 4 derivatives that

have been linked to the verb (i.e., the base word) through morphological and semantic links are examined in more detail in the following paragraphs; the links between the base word and these derivatives are visualised in Figure 1. *Atgādināšana* names the action derived from the verb “to remind”. It was created by one of the linguists of the project as it did not previously exist as a separate dictionary entry. This derivative contains two senses: *atgādināšana*₁ ‘the act of reminding’ and *atgādināšana*_{1.1} ‘the act of reminding (by device)’ which are linked symmetrically to the base word senses 1 and 1.1 using the semantic link “toDo – action”, where “toDo” and “action” are roles for both ends of the link from Table 1.

Atgādinājums denotes the result of the act of reminding; it has three senses: *atgādinājums*₁ ‘a reminder (written or spoken)’, *atgādinājums*_{1.1} ‘a written reminder (incl. by device)’, and *atgādinājums*_{1.2} ‘a reminder of a fact, event’. All three semantic derivation links to these senses are of the “toDo – result” type, however, they are not symmetrical (see Figure 1). E.g., first sense of the derivative and its subsense are both linked to *atgādināt*₁. The reason can be both word meaning peculiarities and previous reviewing and amendment of the entries.

Atgādne ‘a reminder (usually written)’ is a more specific term for a general reminder. The entry only has one sense, which is linked to the first sense of the base word by the “toDo – instrument” semantic link type.

Atgādinātājs ‘someone/something that reminds’ is another three-sense derivative, but in this case, the semantic link distribution with the base word is symmetrical. It is the variety of semantic derivation links that stands out in this case: each derivative sense is ascribed a different role (“agent”, “device”, “cause”), whilst the roles of the base word are either “toDo” or “toBeInState”, demonstrating the wide range of meanings that even relatively simple derivatives may contain.

It is worth noting that the second sense of the base word *atgādināt*₂ is not linked to any of the senses of the derivatives, which further highlights the complex, irregular semantic link structures between the senses of derived words.

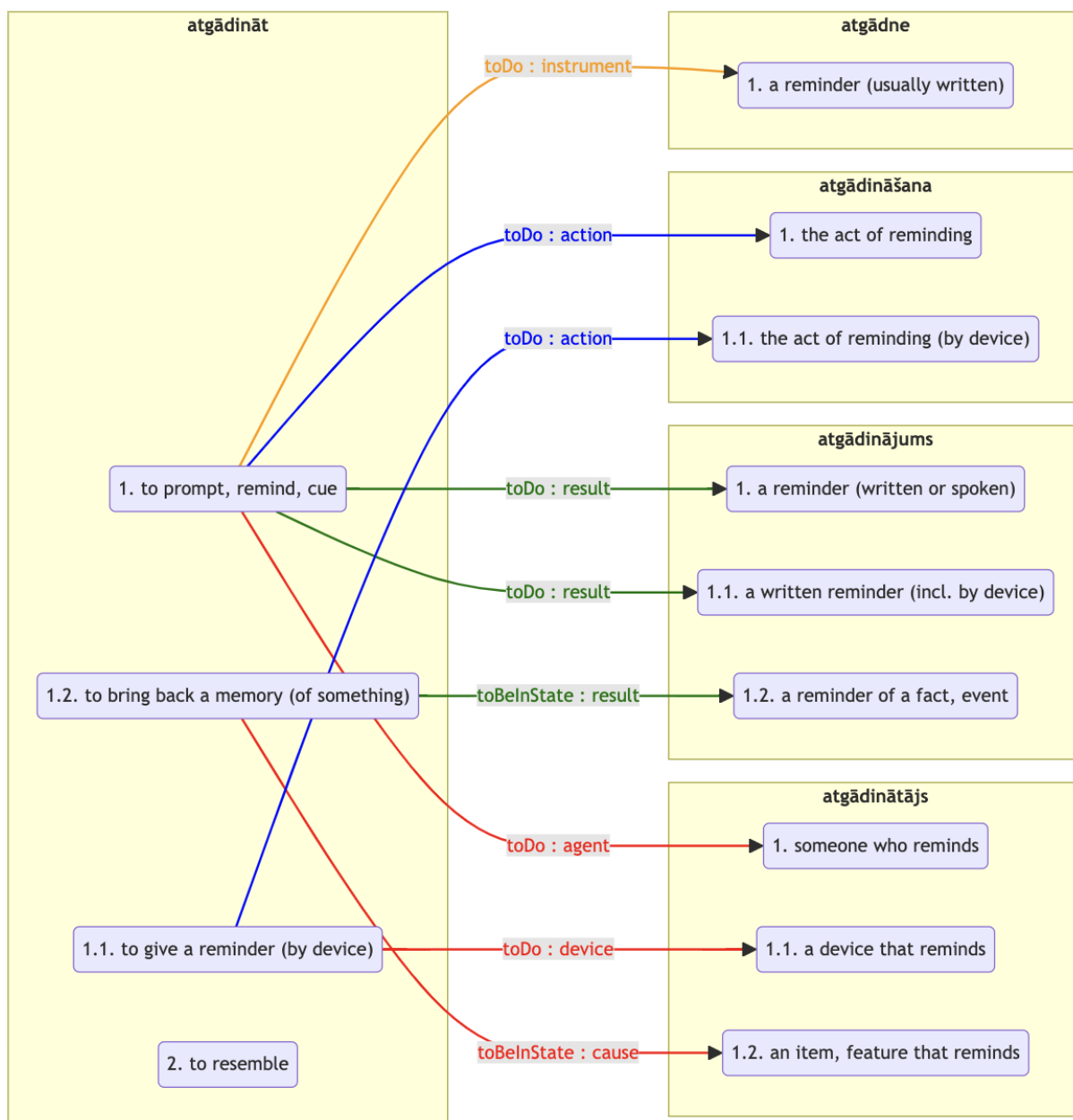


Figure 1: Relations between the senses of the verb *atgādināt* and its derivatives

3.4 Problematic Cases and Solutions

Polysemous derivatives can sometimes pose a challenge for annotation due to their gradual shifts in meaning. There are certain cases when the basic and usually most general sense of a derivative may be lost or rarely used, as the derivative has developed more specific senses over time. This is illustrated by the noun *laidiens* ‘a release’ derived from the verb *laist* ‘to let’ or the noun *darījums* ‘a transaction’ derived from *darīt* ‘to do’. The solution for annotating such cases may be twofold depending on corpus data – either to include the

basic sense in the entry with a tag ‘rarely’, or not to include it at all. In the latter case, the general derivational semantics exist only as a potential and remain unrevealed in semantic derivational links.

Due to diverse sense granularity of the base word and the derivative, attempts to obtain symmetry between the two might lead to an unnecessarily fine-grained distinction of senses. Instead, two following linking patterns can additionally be employed: (a) one sense of the base word is linked to several senses of the derivative (*plānot* ‘to plan’ is linked to two senses of the derivative *plānotājs*

‘a planner’: those of an agent and of a device), (b) several senses of the base word are linked to a single sense of the derivative (two senses ‘to know (how to)’ and ‘to be able to’ of the base word *mācēt* are linked to the single sense of the derivative *māka* ‘a skill’) (on a similar asymmetry between word senses in English see (Mititelu, 2018)).

4 Conclusions and Future Work

The creation of derivational semantics resource has been started, the first such open-access resource for the Latvian language. To reflect the possible difference in derivational semantics between the senses of one polysemous word, two types of links are created in the resource - a morphological link between lexemes and a semantic link between word meanings. A semantic link is formed by a pair of labels assigned to each linked sense. This results in a more informative resource than the general models of derivational semantics described in grammar alone. The first processed data consist of approx. 1000 morphological links and 1600 semantic links and the data is available in the autumn release of Tēzaurs.lv, and from the winter release, it will also be available in the public version of Tēzaurs.lv in the entries of the processed words.

In the future, first of all, it is planned to cover other pairs of word classes involved in Latvian derivation, including derivation pairs within the same word class. Secondly, it is planned to automate part of the process – to find the existing entries of derivatives in the dictionary according to templates, to check in the corpus what kind of derivatives are used for a certain base word and compare with the dictionary data to create the missing entries. Thirdly, it is planned to create a good search system in the data, so that we can further study which derivatives form which semantics. We would like to pay special attention to the semantic relations of polysemous words with their derivatives. Plans for further work also include the integration of the derivational links within Latvian WordNet, as there is a difference between synset-to-synset WordNet links and the derivational links that apply to specific words within that synset, and more study is needed to determine the proper representation for that interaction.

Acknowledgments

This research was funded by Latvian Council of Science project “Advancing Latvian computational lexical resources for natural language understanding and generation” (LZP2022/1-0443).

References

- Orhan Bilgin, Ozlem Cetinoglu, and Kemal Oflazer. 2004. Morphosemantic relations in and across wordnets: A study based on Turkish. In *Proceedings of the Global Wordnet Conference*.
- Tsvetana Dimitrova, Ekaterina Tarpomanova, and Borislav Rizov. 2014. Coping with derivation in the Bulgarian Wordnet. In *Proceedings of the Seventh Global Wordnet Conference*, pages 109–117, Tartu, Estonia. University of Tartu Press.
- Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Language, Speech and Communication. Mit Press.
- Mikus Grasmanis, Peteris Paikens, Lauma Pretkalnina, Laura Rituma, Laine Strankale, Arturs Znotins, and Normunds Gruzitis. 2023. Tēzaurs.lv – the experience of building a multifunctional lexical resource. In *Electronic lexicography in the 21st century (eLex 2023): Invisible Lexicography. Proceedings of the eLex 2023 conference*, pages 400–418. Lexical Computing CZ s.r.o.
- Andra Kalnača and Ilze Lokmane. 2021. *Latvian Grammar*. University of Latvia Press, Riga.
- Kristīne Levāne-Petrova and Roberts Dargis. 2018. Balanced corpus of modern Latvian (LVK2018).
- Ilze Lokmane, Laura Rituma, Madara Stāde, and Agute Klints. 2021. The Latvian WordNet and word sense disambiguation: Challenges and findings. In *7th Biennial Conference on Electronic Lexicography (eLex)*, pages 232–246.
- Verginica Mititelu, Svetlozara Leseva, and Ivelina Stoyanova. 2021. Semantic analysis of verb-noun derivation in Princeton WordNet. In *Proceedings of the 11th Global Wordnet Conference*, pages 108–117, University of South Africa (UNISA). Global Wordnet Association.
- Verginica Barbu Mititelu. 2012. Adding morphosemantic relations to the Romanian Wordnet. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, pages 2596–2601, Istanbul, Turkey. European Language Resources Association (ELRA).
- Verginica Barbu Mititelu. 2018. Investigating English affixes and their productivity with Princeton WordNet. In *Global WordNet Conference*.

- Pēteris Paikens, Agute Klints, Ilze Lokmane, Lauma Pretkalniņa, Laura Rituma, Madara Stāde, and Laine Strankale. 2023. Latvian WordNet. In *Proceedings of the Twelfth Global Wordnet Conference*, pages 187–196, University of the Basque Country, Donostia - San Sebastian, Basque Country. Global Wordnet Association.
- Maciej Piasecki, Radosław Ramocki, and Marek Maziarz. 2012. Recognition of Polish derivational relations based on supervised learning scheme. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 916–922, Istanbul, Turkey. European Language Resources Association (ELRA).
- Adam Rambousek, Aleš Horák, and Karel Pala. 2018. Sustainable long-term WordNet development and maintenance: Case study of the Czech WordNet. *Cognitive Studies — Études cognitive*, 18:75–81.
- Victor Raskin and Sergei Nirenburg. 1995. Lexical semantics of adjectives: A microtheory of adjectival meaning. *MCCS report 95*, 288.
- Baiba Saulite, Roberts Dargis, Normunds Gruztis, Ilze Auzina, Kristīne Levāne-Petrova, Lauma Pretkalniņa, Laura Rituma, Pēteris Paikens, Arturs Znotiņš, Laine Strankale, Kristīne Pokratniece, Ilmārs Poikāns, Guntis Barzdins, Inguna Skadiņa, Anda Baklāne, Valdis Saulespurāns, and Jānis Ziediņš. 2022. Latvian national corpora collection – Korpuss.lv. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5123–5129, Marseille, France. European Language Resources Association.
- Emīlija Soida. 2009. *Vārdarināšana*. University of Latvia Press, Riga.
- Andrejs Spektors, Lauma Pretkalniņa, Normunds Grūzītis, Pēteris Paikens, Laura Rituma, Baiba Saulīte, Gunta Nešpore-Bērzkalne, Ilze Lokmane, Agute Klints, Madara Stāde, Mikus Grasmanis, Laine Strankale, Ilze Auziņa, Artūrs Znotiņš, Roberts Dargis, and Guntis Bārzdiņš. 2023. Tēzaurs.lv 2023 (summer edition). CLARIN-LV digital library at IMCS, University of Latvia.
- Anna Wierzbicka. 1988. *The Semantics of Grammar*. Companion series. J. Benjamins Publishing Company.

Poro 34B and the Blessing of Multilinguality

Risto Luukkonen^{1,2} Jonathan Burdge² Elaine Zosa² Aarne Talman³
Ville Komulainen¹ Väinö Hatanpää⁴ Peter Sarlin² Sampo Pyysalo¹

¹TurkuNLP, University of Turku, Finland ²Silo AI, Finland

³University of Helsinki, Finland ⁴CSC – IT Center for Science, Finland

risto.m.luukkonen@utu.fi jonathan.burdge@silo.ai

peter@silo.ai sampo.pyysalo@utu.fi

Abstract

The pretraining of state-of-the-art large language models now requires trillions of words of text, which is orders of magnitude more than available for the vast majority of languages. While including text in more than one language is an obvious way to acquire more pretraining data, multilinguality is often seen as a curse, and most model training efforts continue to focus near-exclusively on individual large languages. We believe that multilinguality can be a blessing: when the lack of training data is a constraint for effectively training larger models for a target language, augmenting the dataset with other languages can offer a way to improve over the capabilities of monolingual models for that language. In this study, we introduce Poro 34B, a 34 billion parameter model trained for 1 trillion tokens of Finnish, English, and programming languages, and demonstrate that a multilingual training approach can produce a model that substantially advances over the capabilities of existing models for Finnish and excels in translation, while also achieving competitive performance in its class for English and programming languages. We release the model parameters, scripts, and data under open licenses at <https://huggingface.co/LumiOpen/Poro-34B>.

1 Introduction

Neural language models based on the transformer architecture (Vaswani et al., 2017) have led to substantial advances in natural language processing. Encoder-only transformer models such as BERT (Devlin et al., 2019) have advanced the state of the art in a broad range of classification tasks, while

decoder-only models such as GPT (Radford et al., 2018) have redefined what can be achieved by generative models, opening new areas of study in prompting and in-context learning. The success of these models is related in substantial part to their scaling properties: training larger models on more data leads to better results and even entirely new capabilities (Brown et al., 2020). Studies refining our understanding of the optimal balance of model size and training steps have increased the demands on data (Hoffmann et al., 2022b), and many recent models optimize further for inference-time efficiency by training smaller models on more data (Sardana and Frankle, 2023).

These developments have introduced increasing demands for textual data, with many recent models pretrained on a trillion tokens or more (e.g. Touvron et al., 2023; Almazrouei et al., 2023; MosaicML, 2023; Li et al., 2023; Lozhkov et al., 2024; Groeneveld et al., 2024). While such resources can still be assembled from internet crawls for a few of the languages best represented online, for the vast majority of human languages we have already run out of data for training the largest of language models (Joshi et al., 2020; Villalobos et al., 2022). While it is standard to repeat training data, repetition can lead to reduced sample efficiency and degradation of performance (Hernandez et al., 2022): Muenighoff et al. (2024) estimate that the value of repetition starts to diminish rapidly after four epochs and that repetition ceases to add information around 40 epochs. The availability of data is thus currently a limit for monolingual training for all but a few of the highest-resourced languages.

Multilingual training offers one obvious solution for increasing the amount of training data available, and a large number of multilingual transformer models have been introduced (e.g. Conneau et al., 2020; Lin et al., 2022b; Le Scao et al., 2022; Wei et al., 2023). However, despite the intuitive appeal of augmenting training data with texts in other nat-

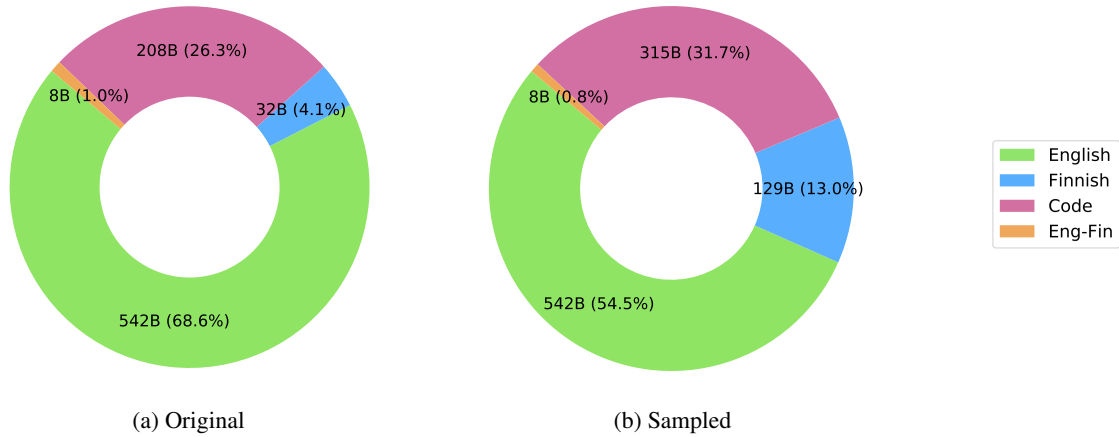


Figure 1: Pretraining data distribution.

ural languages, multilinguality is frequently seen as a negative – commonly referred to as the *curse of multilinguality* (Conneau et al., 2020). While there have been studies of the tradeoffs between monolingual and multilingual training (Fujinuma et al., 2022; Chang et al., 2023) as well as efforts to enhance models specifically for multilinguality (Pfeiffer et al., 2022) and to introduce additional language capabilities to existing models (Gogoulou et al., 2023; Kew et al., 2023; Zhao et al., 2024; Ibrahim et al., 2024), state-of-the-art generative models are still frequently trained near-exclusively on large languages such as English, with only limited efforts specifically focusing on optimizing performance for smaller languages. In this study, we explore how to lift data limitations to create state-of-the-art large generative models from scratch for smaller languages, drawing on the understanding emerging in recent studies on how to make the most of limited data and assure that multilinguality is a blessing rather than a curse. Some key lessons from previous work include 1) **limited multilinguality** instead of a large number of languages (Conneau et al., 2020; Chang et al., 2023) 2) **matching scripts** (e.g., Latin) (Fujinuma et al., 2022) and 3) **matching language families** (Pyysalo et al., 2021), 4) incorporating a **cross-lingual signal** using translation pairs (Anil et al., 2023; Wei et al., 2023), 5) **oversampling target language** data up to four epochs (Muennighoff et al., 2024) and 6) augmenting natural language with **programming language data** (Madaan et al., 2022; Aryabumi et al., 2024).

We chose to specifically target the Finnish language, which is an interesting case for study as it is a Uralic language with no large close neighbours

in its language family, necessitating more distant transfer than, for example, between English and another Germanic language. While the language is natively spoken by under six million people, its resources are still sufficient to consider a monolingual training approach for larger generative models. In a recent study, Luukkonen et al. (2023) combined several web crawls and curated sources of Finnish to create a dataset of approximately 40B tokens and introduced the monolingual FinGPT models trained from scratch for 300B tokens. With approximately 8 epochs, the repetition of data is expected to show diminishing returns (Muennighoff et al., 2024), and the largest of these models show signs of data limitations, with the 8B parameter model outperforming the 13B in benchmarks. We believe it should be possible to overcome these limitations by applying the lessons listed above. While we cannot match language families, we train for four epochs over the Finnish data and augment it with both English and programming language data as well as an explicit cross-lingual signal from translation pairs. We pursue this approach to create Poro 34B, training a 34B parameter model for a total of 1T tokens – 25 times more than the available Finnish data – and evaluate the model in detail on Finnish, English, and programming language tasks. We find that the model not only achieves the goal of substantially advancing over the performance of existing Finnish models, but is also competitive in its class of open models on English and code as well as remarkably strong in translation tasks.

2 Pretraining data

For pretraining Poro 34B, we rely on datasets that have been previously preprocessed to remove

low-quality texts and boilerplate, filter toxic context, and deduplicate repeated texts. We illustrate the pretraining data distribution in Figure 1 and describe the data briefly in the following. Data sources are detailed in Table 4 in the Appendix.

Finnish For Finnish pretraining data, we draw on the resources recently introduced by Luukkonen et al. for creating the FinGPT model family. We exclude the *ePub* and *Lehdet* resources provided by the National Library of Finland for that work as they could not be shared due to copyright limitations, but use the remaining sources of data, totalling to a 32B token monolingual corpus. The majority of the Finnish data originates from web crawls (approx. 84%) complemented with news sources (approx. 2%), Project Lönnrot, the Finnish equivalent of Project Gutenberg copyright-free book corpus (approx. 0.5%), Wikipedia (approx. 0.5%) and Finnish online discussion forum contents from Reddit and Suomi24 (approx. 13%). Following the rule of thumb proposed by Muenighoff et al. (2024), we upsample the 32B tokens of Finnish so that four epochs over the data are made during training. Consequently, approximately 13% of the total tokens seen in pretraining are Finnish.

English For English pretraining data, we primarily use SlimPajama (Soboleva et al., 2023), a cleaned and deduplicated subset of the RedPajama corpus¹ (Together Computer, 2023), from which we excluded data from the books category due to their copyright status. We supplemented this dataset with the Project Gutenberg public domain books data from the Dolma corpus² (Soldaini et al., 2024). We train for one epoch over the 542B tokens of the English data, which thus represents slightly over half of the 1T total training tokens.

Programming Languages To introduce data representing various programming languages (referred to hereinafter as “code” for short) into our pretraining, we make use of the Starcoder corpus (Li et al., 2023), a processed subset of The Stack corpus³ (Kocetkov et al., 2023). The original corpus consists of 208B tokens, which we oversample 1.5x so that approximately a third of the tokens seen during pretraining represent code.

¹<https://huggingface.co/datasets/togethercomputer/RedPajama-Data-1T>

²<https://huggingface.co/datasets/allenai/dolma>

³<https://huggingface.co/datasets/bigcode/the-stack>

Cross-lingual data We introduce a cross-lingual signal into pretraining by including translation examples from OPUS (Tiedemann, 2009). Specifically, we use the English-Finnish examples from the Tatoeba dataset (Tiedemann, 2020) to generate instruction-formatted translation examples. The Tatoeba training data was reformatted into a minimalistic instruction-following format by recasting each English-Finnish translation pair into a document with the following format:

```
<|user|>Translate into Finnish: {{en}}
<|assistant|>{{fi}}
```

Where {{en}} and {{fi}} are the English and Finnish texts (resp.) of the translation pair. We additionally reverse the translation order (i.e., Finnish to English instead of English to Finnish) for a total of two documents for each sentence pair. No weighting is applied to the approximately 8B tokens of cross-lingual data, which thus represents slightly under 1% of the pretraining tokens.

3 Methods

In this section, we describe the method used to create the Poro 34B tokenizer, the pretraining setup, and provide an estimate of the compute cost of pretraining the model.

3.1 Tokenization

The choice of tokenizer has a broad range of impacts, not only on the efficiency of training and inference but also the capabilities of trained models (Rust et al., 2021; Petrov et al., 2023; Ali et al., 2023). As we were not aware of any existing tokenizer that would be a good fit for our combination of languages and code, we created a new tokenizer for our model. Specifically, we trained a custom byte-level BPE tokenizer using the same pre-normalization as the FinGPT tokenizer. We selected a vocabulary size of 128K tokens, aiming to achieve low fertility on the targeted languages while keeping the vocabulary reasonably small. The tokenizer was trained on a uniform distribution of samples of the Finnish, English and code datasets.

We assess the fertility of the tokenizer on the English and Finnish sentences from the devtest portion of the widely used Flores-101 benchmark for machine translation (Goyal et al., 2022), which allows for a degree of cross-lingual comparability. For code, we use an approximately 1M character sample of lines from the Starcoder held-out test

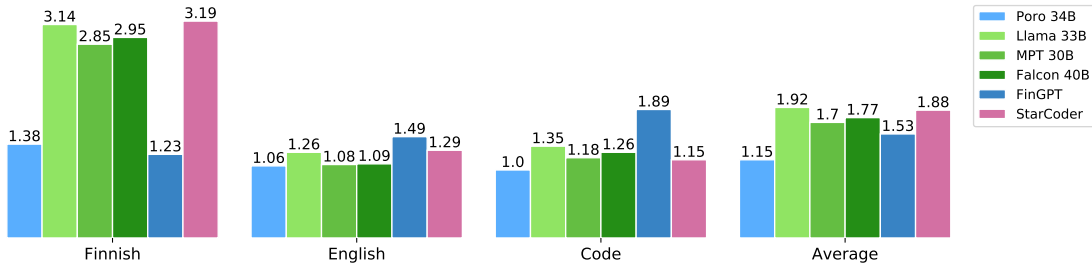


Figure 2: Tokenizer fertility comparison (lower is better).

data.⁴ Figure 2 provides a comparison of the fertility of the tokenizer compared to selected reference tokenizers (see Section 4). We find that on this data the new Poro 34B tokenizer has at least broadly comparable fertility to the lowest-scoring tokenizer on each of Finnish, English, and code, as well as the lowest average fertility of the compared tokenizers.

3.2 Pretraining

We next briefly present the key model and training parameters (detailed in Table 5 in the Appendix A.1) and the pretraining software and configuration.

Architecture Poro 34B is a decoder-only model with a parameter count of 34 billion, sharing its architecture with FinGPT (Luukkonen et al., 2023) and BLOOM (Le Scao et al., 2022). It incorporates layer normalization immediately following the input embedding layer for better training stability and uses ALiBi (Press et al., 2021) as its positional encoding method. The model consists of 54 layers with a hidden dimension of 7168 and a total of 56 attention heads.

Training We train to 1T tokens, intentionally exceeding the Chinchilla compute-optimality estimate (Hoffmann et al., 2022a) of approximately 700B tokens for a model of this size, thus gaining inference-time efficiency for the cost of additional compute investment in pretraining (Sardana and Frankle, 2023). We train with a sequence length of 2048 tokens⁵ using a cosine learning rate scheduler with a maximum learning rate of $1.5e-4$, decaying to a minimum of $2e-5$ over 990B tokens, and a linear warmup of 10B tokens. Our global batch

size is 2048 samples totaling to 4194304 tokens per optimization step.

Software Poro 34B was trained on the LUMI supercomputer GPU partition, which is powered by AMD MI250X GPUs. The majority of open source frameworks for large language model pretraining are made to be primarily NVIDIA-compatible, and we required scalable AMD-compatible training software. Thus, we adopted the Megatron-DeepSpeed fork⁶ introduced by (Luukkonen et al., 2023), which has optimized kernels converted from CUDA to be compatible with AMD ROCm, and has been demonstrated to be a viable solution for large model pretraining on LUMI. The hardware used to train the model is described in detail in Appendix A.3.

Configuration Considering the hardware available and the selected hyperparameters such as batch size, a configuration of 128 nodes was chosen for the training of the model, resulting in a world size of 1024. The training was done using activation checkpointing, a micro batch size of 1, gradient accumulation of 16, and a 3D parallelism strategy of tensor parallel degree 2, pipeline parallel degree 4, resulting in a data parallel degree of 128. This allowed total training cycle throughput of 49618 TFLOPs and 174378 tokens/second.

3.3 Compute cost

Following (Groeneveld et al., 2024), we estimate the carbon footprint of our pretraining by multiplying the theoretical upper bound of the total power used by the GPUs when they are utilized at 100% with the carbon intensity factor of LUMI. Taking into account the systems’s power usage effectiveness (PUE) value of 1.04,⁷ we approximate the total power consumption to be 448MWh. As LUMI is

⁴We only sample lines with at least 10 alphabetic characters to avoid very short lines.

⁵We acknowledge that this can be considered limiting by today’s standards, but this limitation can be relieved by methods for extending the context length, for example via linear extrapolation (Press et al., 2021) or interpolation (Al-Khateeb et al., 2023).

⁶<https://github.com/TurkuNLP/Megatron-DeepSpeed>

⁷<https://www.lumi-supercomputer.eu/sustainable-future/>

powered by fully renewable electricity, we assume the carbon intensity factor to be 0.⁸ This brings our emissions to a total of 0 tCO₂eq. It is important to note that we only take into account power consumption of the GPUs used, as the consumption of the entire node was not logged during training.

4 Evaluation

We thoroughly analyze the capabilities of the model for Finnish, English and code, first briefly reporting perplexity results and then focusing on community-standard benchmarks for evaluating generative models. We then assess the quality of Finnish text generated by the model and finally evaluate the model’s translation capability from English to Finnish (and vice versa). For comparison, we include results for the state-of-the-art Finnish language models, FinGPT 8B and FinGPT 13B (Luukkonen et al., 2023), and a selection of similarly-sized general-purpose open source base language models trained on broadly comparable numbers of tokens for English⁹: Llama 33B (Touvron et al., 2023), MPT 30B (MosaicML, 2023), and Falcon 40B (Almazrouei et al., 2023). We also provide results for StarCoder base (Li et al., 2023) as a reference for performance on code tasks.

4.1 Data and experimental setup

We assess the perplexity of the model on the same data used to evaluate tokenizer fertility (Section 3.1), namely Flores-101 devtest English and Finnish and a sample of the StarCoder test data. As token-level perplexity is dependent on tokenization, it cannot be used to directly compare models with different tokenizers. We therefore report character-level perplexity PPL_c following Ekgren et al. (2022), normalizing by character rather than token count when calculating perplexity.

We benchmark the capabilities of the model in Finnish using the FIN-bench¹⁰ dataset (Luukkonen et al., 2023), which covers a variety of tasks to assess various aspects of model capabilities in

Finnish, combining selected tasks translated and manually corrected from English BIG-bench (Srivastava et al., 2022) with additional Finnish tasks. We evaluate all FIN-bench results in a 3-shot setting using the standard metrics defined for the benchmark. For English evaluations, we use LM Eval Harness (Gao et al., 2023) to evaluate with the following datasets: ARC Challenge (Clark et al., 2018), GSM8K (Cobbe et al., 2021), HellaSwag (Zellers et al., 2019), MMLU (Hendrycks et al., 2021), TruthfulQA (Lin et al., 2022a), and Winogrande (Sakaguchi et al., 2019). We selected these evaluations based on their use as English language benchmarks by Beeching et al. (2023) and use an identical testing configuration here. Programming language proficiency is assessed via the Bigcode Evaluation Harness (Ben Allal et al., 2022) with the HumanEval (Chen et al., 2021), and MBPP (Austin et al., 2021) benchmarks, employing the pass@10 metric for evaluation.

To evaluate the quality of Finnish text generation, we generate responses to the translated MT-Bench questions with few-shot prompting (Zheng et al., 2023). We use a few-shot prompt because this benchmark is designed for chat models and we are evaluating base models. Moreover, we want to unlock the Finnish generation capabilities of the English-focused models by providing in-context examples in Finnish. We use GPT-4 Turbo and human judges to assess the quality of the responses. Finally, to evaluate translation performance, we use both the Flores-101 devtest (Goyal et al., 2022) as well as the Tatoeba test sets (Tiedemann, 2020) in an 8-shot setting, following Zhu et al. (2023).

4.2 Perplexity

Table 1 summarizes the results of the perplexity evaluation as mean character-level perplexity PPL_c for various models over the sentences/code lines. We find that Poro 34B has comparatively low (good) PPL_c on all three datasets, including the best result for Finnish. Poro 34B is to the best of our knowledge the only open model specifically trained for this combination of languages, and it is thus not surprising that it has the best overall average in this evaluation. While perplexity is not necessarily predictive of downstream performance and these datasets only represent a part of the relevant distribution, the result suggests that the model has learned all of its target languages well.

⁸We acknowledge that this assumption can be contested. As (Groeneveld et al., 2024) note: "LUMI is powered entirely by hydroelectric power and some sources (Ubierna et al., 2022) measure the carbon intensity factor of hydroelectric power to be 0.024."

⁹We chose English models of similar size and training token budget rather than state-of-the-art models to more directly assess the effects of our multilingual training setup on performance in English.

¹⁰<https://github.com/TurkuNLP/FIN-bench>

	Poro 34B	Llama 33B	MPT 30B	Falcon 40B	FinGPT 8B	FinGPT 13B	StarCoder
Finnish	1.89	2.98	2.89	3.57	1.94	1.92	3.83
English	1.87	1.81	1.89	1.85	2.55	2.46	2.38
Code	3.21	4.27	3.58	3.65	25.1	27.3	3.15
Average	2.32	3.02	2.79	3.02	9.86	10.6	3.12

Table 1: Character-level perplexity for Poro 34B and selected reference models (lower is better).

	Poro 34B	Llama 33B	MPT 30b	Falcon 40B	FinGPT 8B	FinGPT 13B	StarCoder
Finnish	66.28	53.36	53.22	42.58	49.69	48.92	45.55
English	50.57	59.96	52.62	49.87	31.47	32.85	35.44
Code	41.80	37.67	39.18	38.57	-	-	49.06

Table 2: Average benchmark results for Finnish, English and code for Poro 34B and selected reference models.

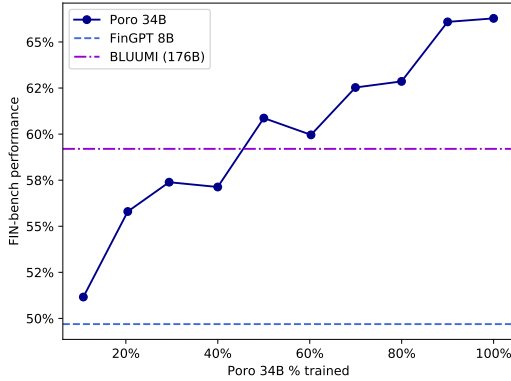


Figure 3: Poro 34B performance progression on FIN-bench. For reference, dotted lines show results for the best-performing monolingual FinGPT model and the massively multilingual BLUUMI model (Luukkonen et al., 2023), an extension of BLOOM (Le Scao et al., 2022) with Finnish.

4.3 Benchmark results

The overall results of the benchmark evaluations are summarized in Table 2 and detailed in Appendix A.2. We find that Poro 34B is the best-performing model for Finnish in this comparison, substantially outperforming the best previously introduced monolingual Finnish model. We further analyzed the progression of the Finnish capabilities by evaluating Poro 34B checkpoints at 10% intervals on FIN-bench. These results are summarized in Figure 3. Interestingly, the model outperforms the best FinGPT model already after 100B tokens of training (10%) despite the relatively small proportion of Finnish in the Poro 34B data and the fact that the FinGPT models were trained on

300B tokens in total. These results indicate that our limited multilingual approach is effective for creating stronger models for Finnish than possible through monolingual training and demonstrate that the model is benefiting substantially from its training data in other languages even when tested on Finnish tasks.

For English, we find that the model achieves broadly comparable results to the MPT 30B and Falcon 40B models, both of which were trained for 1T tokens of predominantly English data. This indicates that the limited multilingual training approach has not notably detracted from the English capabilities of the model. The best-performing open model in this comparison is Llama 33B, which was trained for longer (1.4T tokens), also predominantly on English data. We find that Poro 34B is nevertheless a capable model in its class also for English, despite not optimizing specifically for English performance. The programming language benchmarks indicate that Poro 34B is more capable on code than the other natural language-focused models, while the code-focused StarCoder model clearly outperforms all of the other models. We attribute the relatively high performance of Poro 34B on code to the comparatively large proportion of the training data dedicated to code. As with English, we consider the performance of the model on code a positive addition even though code generation was not a primary goal in creating the model.

Finally, we note a surprising finding arising from the Finnish evaluation: two of the larger English-focused models (Llama 33B and MPT 30B) score higher than the previously introduced smaller monolingual Finnish models on the FIN-

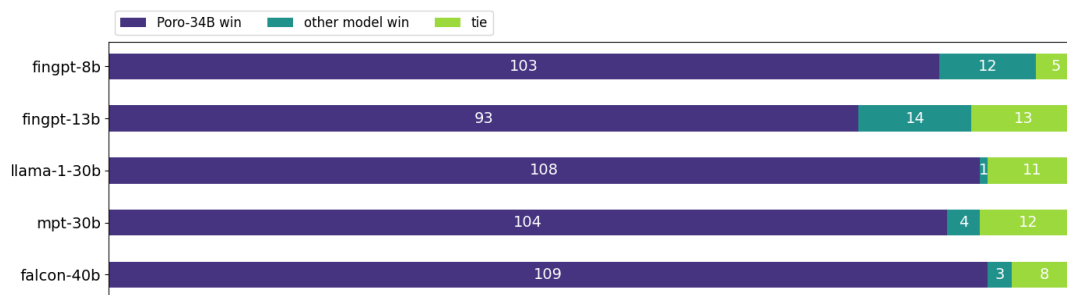


Figure 4: Win counts of reference models against Poro 34B on Finnish MT-Bench as judged by GPT-4 Turbo.

bench benchmark. While FIN-bench tasks are in Finnish, the benchmark consists of multiple-choice rather than generation tasks, has been produced in substantial part through translation from English, and includes tasks with little emphasis on natural language (esp. arithmetic). We hypothesize that the comparatively high performance of the English-focused models on this benchmark might not indicate that they can generate fluent Finnish, which also calls the Finnish proficiency of Poro 34B into question. We study this question specifically in the following section.

4.4 Open-ended generation

To assess the ability of the models to generate coherent and grammatically correct Finnish, we create a Finnish version of MT-Bench (Zheng et al., 2023), a benchmark for open-ended conversations that uses LLM-as-a-judge evaluation. We excluded math and coding questions to focus specifically on the natural language generation capabilities of the models. To create the benchmark, we initially translated the questions into Finnish using DeepL,¹¹ and the translations were then manually corrected by native Finnish speakers to create the final evaluation dataset. To evaluate base models using the data, we similarly translated and corrected the few-shot URIAL prompt (Lin et al., 2024).¹² We use pairwise judging to compare between Poro 34B and the competing models’ responses and use GPT-4 Turbo as the judge model.

To assess the reliability of the model as a judge and provide further insight into the quality of the generations, we additionally set up an annotation platform where two native Finnish speakers were

asked to pick a preference between a response generated by Poro 34B and a competing model.¹³ The judges are given the same judging prompt as GPT. The model names are hidden from the judges, and we randomly select the position of each response in every response pair to account for positional bias.

We found that the two human judges highly agree with each other, picking the same winner 88.8% of the time, and found an even higher agreement between GPT and each human judge: 91.6% between annotator 1 and GPT and 89.5% between annotator 2 and GPT. Figure 4 shows the win counts of the reference models against Poro 34B as judged by GPT-4 Turbo.

In manual analysis after the initial annotation, we found that the FinGPT models often struggled with the few-shot format, failing to follow questions or only giving short, minimal answers, while Poro 34B was better able to comply with questions and given requirements, such as listing a specified number of items. However, we found that Poro 34B also often hallucinated and did not follow all instructions, and we would not consider its responses to be at a level of consistency and quality required for user-facing applications, which is not an unexpected result given that it is a base model not specifically fine-tuned or otherwise aligned for such use. Despite outperforming FinGPT models on the FIN-bench benchmark, The English-focused models appeared to be unfit for Finnish generation: their generations had the surface appearance of Finnish text but were largely nonsensical and incoherent. This result underlines the need to include multiple perspectives when evaluating models: a high score on a multiple-choice benchmark may not indicate practical capability to generate coherent text in a language.

¹¹<https://www.deepl.com>

¹²We did not modify the judge prompts as previous work has found that keeping the prompt in English produces better results (Ahuja et al., 2023).

¹³We did not separately compensate the human judges as they are co-authors of this paper.

We make the Finnish MT-Bench available under an open license and provide the model generations at https://github.com/LumiOpen/FastChat/tree/main/fastchat/llm_judge.

4.5 Translation

General-purpose language models have shown promising results on translation benchmarks on multiple languages (Vilar et al., 2023; Garcia et al., 2023; Alves et al., 2024). Following Zhu et al. (2023), we evaluated Poro 34B for English to Finnish translation and vice versa on the first 100 sentences of the Flores-101 test data by prompting the model with eight translation examples sampled randomly from the development set, formatting the examples simply as `<src>=<trg>`. We further evaluated Poro 34B and three strong open-source translation models on the Tatoeba test set with more than 11,000 sentences: OPUS-MT (Tiedemann and Thottingal, 2020), NLLB-1.3B (Costa-jussà et al., 2022), and M2M-100-12B (Fan et al., 2021)¹⁴. We used the standard SentencePiece BLEU (spBLEU) as our metric. The results of both evaluations are shown in Table 3.¹⁵ These results demonstrate that Poro 34B is a remarkably strong translator, outperforming not only dedicated open-source translation models but even Google Translate, and scoring roughly on par with GPT-4 in this evaluation. We attribute this result to the combination of strong Finnish and English capabilities and the inclusion of a comparatively large number of translation examples in the pretraining data.

It should be noted, however, that the Tatoeba and Flores sentences are relatively short and simple, and this evaluation does thus not capture the full picture of the translation capabilities of the evaluated models. We aim to assess the translation capability of Poro 34B more comprehensively on longer texts, especially texts that might include different modalities such as tables and code, in future work.

5 Discussion and conclusions

In this study, we have considered the challenges that the availability of data poses for pretraining

¹⁴We did not evaluate the GPT models and Google Translate on Tatoeba because of the associated API costs.

¹⁵We attempted to reproduce some of the Flores-101 results reported by (Zhu et al., 2023) and obtained a slightly higher result for GPT-4 in Eng-Fin translation (37.5 instead of 35.33) and slightly lower results for M2M-12B and NLLB-1.3B (31.4 and 26.6, respectively). For the sake of consistency, we present the results from that study without modification.

Model	Flores-101		Tatoeba	
	En-Fi	Fi-En	En-Fi	Fi-En
ChatGPT	33.4	35.9	-	-
GPT-4	35.3	40.2	-	-
Google	37.3	39.0	-	-
M2M-12B	33.3	33.8	36.7	41.3
NLLB-1.3B	30.0	35.4	40.2	55.7
OPUS-MT	37.2	35.6	46.7	58.4
Poro 34B	37.6	39.8	47.3	60.5

Table 3: spBLEU on the Flores-101 devtest and Tatoeba test sets. Flores-101 results except for OPUS-MT and Poro 34B are from Zhu et al. (2023).

large generative models for smaller languages and explored a limited multilingual approach to create Poro 34B, a 34B-parameter model trained on 1T tokens of Finnish, English, and code, including 8B tokens of Finnish-English translation pairs. We thoroughly evaluated the model and found it to substantially advance over the performance of existing models for Finnish while also performing competitively in its class of open models for English and code generation, as well as achieving remarkably good results in translation tasks. Two human judges and GPT-4 Turbo found the texts generated by Poro 34B to be superior to the competing models.

Our model architecture and the Finnish datasets included follow those of the FinGPT family of monolingual Finnish models, which were constrained by the available Finnish training data. The superior performance of our model in Finnish evaluations demonstrates that multilingual training can lift such limitations, allowing further scaling of models focused on smaller languages. In future work, we hope to explore this effect more systematically to answer some of the many questions that remain open regarding the training of large generative models for smaller languages, including the impacts of covering multiple smaller languages and the effect of the size of data available in the target languages.

A number of the choices made in training Poro 34B were made with incomplete information regarding their specific impacts on the final model. For example, we opted to include a comparatively large amount of programming language data as well as instruction-formatted translation examples in the pretraining data, the latter on the assumption

that this would provide a cross-lingual signal that would strengthen the ability of the model to benefit from data in a more distantly related language (English). While this approach is intuitively appealing and the performance of our model suggests that it has at a minimum not notably detracted from the capabilities of the model, we did not as part of this work have the resources to conduct ablation studies nor to explore alternative ways to incorporate cross-lingual information in pretraining. We aim to study these questions further in future work.

We hope that our approach can serve as a template for the creation of larger models for other smaller languages and that the model introduced in this work can serve as both as a focus of research in its own right as well as a starting point for further pretraining, finetuning and alignment to create useful models, tools and methods not only for Finnish but also other languages. We release the model weights as well as all relevant documentation and software fully openly at <https://huggingface.co/LumiOpen/Poro-34B>.

Limitations

Our study applies a pretraining recipe that combines insights on effective multilingual and data-constrained model training from a variety of previous studies. While the findings of these studies are supported by a broad range of relevant experimental results, we did not have the resources to perform separate ablation experiments specifically assessing the impact that various parts of our combined pretraining recipe (e.g., four repetitions of target language data and the inclusion of a translation signal) have on the resulting model. Thus, while we believe that our results demonstrate the pretraining recipe to be effective for creating state-of-the-art models for data-constrained languages, our work is limited in leaving many questions open regarding specific choices that form part of that recipe.

Poro 34B is a base model and as such has not been aligned to follow instructions and engage in conversations. It has not been evaluated on safety and toxicity benchmarks. As we have noted in our language generation evaluation, Poro 34B does not adequately follow instructions and has the tendency to generate texts with hallucinations. Further research is needed to improve the model in terms of factuality, safety, and alignment in English and Finnish. We encourage developers using Poro 34B to be aware of the potential risks associated with

LLMs such as non-factual outputs, harmful language, and perpetuation of biases and stereotypes. We recommend that developers finetune Poro 34B to meet their specific needs and codes of conduct.

Ethical considerations

We are committed to open science, transparency and accessibility in our work. While we acknowledge the concerns and the potential for negative impacts associated with making powerful generative models and the technology to create them more widely available, we believe that in the case of Poro 34B the positives clearly outweigh the negatives. We discuss some specific concerns and their mitigations in the following.

Poro 34B is a base model trained in substantial part on texts sourced from web crawls, which are known to include biases, toxicity and factual errors. While we have selected curated text sources that have been extensively filtered to remove problematic material, no such filtering is perfect. Like all language models, Poro 34B is a product of its inputs, and its output may reflect issues in its training material. Furthermore, as Poro 34B is a base model that has not been finetuned for any specific purpose, extra care should be taken when interpreting its output, and the model should not be used as is in any application with potential for significant impact on people's rights or well-being. We emphasize these limitations in the model card published with the model.

Pretraining large language models is computationally intensive, and the creation of large models can have substantial environmental impacts. Poro 34B was trained on the LUMI supercomputer, which is powered entirely by renewable energy resources. According to the official specifications, the carbon intensity factor of LUMI's operation is considered to be zero. This approach effectively minimizes the carbon footprint associated with the computational aspects of training our model.

Though concerns about the capabilities of frontier models to cause catastrophic harm have been discussed in the literature, a model of Poro 34B's size and training duration does not represent new frontier capability and releasing the model does not introduce any new classes of risk.

Acknowledgements

The authors wish to acknowledge CSC – IT Center for Science, Finland, for generous computational resources on the LUMI supercomputer. This

project has received funding from the European Union’s Horizon Europe research and innovation programme under Grant agreement No 101070350. The contents of this publication are the sole responsibility of its authors and do not necessarily reflect the opinion of the European Union.

References

- Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, et al. 2023. Mega: Multilingual evaluation of generative ai. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4232–4267.
- Faisal Al-Khateeb, Nolan Dey, Daria Soboleva, and Joel Hestness. 2023. Position Interpolation Improves ALiBi Extrapolation. *arXiv preprint arXiv:2310.13017*.
- Mehdi Ali, Michael Fromm, Klaudia Thellmann, Richard Rutmann, Max Lübbering, Johannes Leveling, Katrin Klug, Jan Ebert, Niclas Doll, Jasper Schulze Buschhoff, et al. 2023. Tokenizer Choice For LLM Training: Negligible or Crucial? *arXiv preprint arXiv:2310.08754*.
- Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hestlow, Julien Launay, Quentin Malartic, et al. 2023. The Falcon series of open language models. *arXiv preprint arXiv:2311.16867*.
- Duarte M. Alves, José Pombal, Nuno M. Guerreiro, Pedro H. Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G. C. de Souza, and André F. T. Martins. 2024. Tower: An open multilingual large language model for translation-related tasks.
- 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*.
- Viraat Aryabumi, Yixuan Su, Raymond Ma, Adrien Morisot, Ivan Zhang, Acyr Locatelli, Marzieh Fadaee, Ahmet Üstün, and Sara Hooker. 2024. To code, or not to code? exploring impact of code in pre-training.
- Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.
- Edward Beeching, Clémentine Fourrier, Nathan Habib, Sheon Han, Nathan Lambert, Nazneen Rajani, Omar Sanseviero, Lewis Tunstall, and Thomas Wolf. 2023. Open llm leaderboard. https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard.
- Loubna Ben Allal, Niklas Muennighoff, Logesh Kumar Umapathi, Ben Lipkin, and Leandro von Werra. 2022. A framework for the evaluation of code generation models. <https://github.com/bigcode-project/bigcode-evaluation-harness>.
- 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. Curran Associates, Inc.
- Tyler A. Chang, Catherine Arnett, Zhuowen Tu, and Benjamin K. Bergen. 2023. When is multilinguality a curse? language modeling for 250 high- and low-resource languages.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgens Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating large language models trained on code.
- Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. *ArXiv*, abs/1803.05457.
- 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.
- 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. Association for Computational Linguistics.
- Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.
- 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.
- Ariel Ekgren, Amaru Cuba Gyllensten, Evangelia Gogoulou, Alice Heiman, Severine Verlinden, Joey Öhman, Fredrik Carlsson, and Magnus Sahlgren. 2022. Lessons learned from GPT-SW3: Building the first large-scale generative language model for Swedish. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3509–3518.
- Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. 2021. Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48.
- Yoshinari Fujinuma, Jordan Boyd-Graber, and Katharina Kann. 2022. Match the script, adapt if multilingual: Analyzing the effect of multilingual pretraining on cross-lingual transferability. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1500–1512. Association for Computational Linguistics.
- Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac’h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2023. A framework for few-shot language model evaluation.
- Xavier Garcia, Yamini Bansal, Colin Cherry, George Foster, Maxim Krikun, Melvin Johnson, and Orhan Firat. 2023. The unreasonable effectiveness of few-shot learning for machine translation. In *International Conference on Machine Learning*, pages 10867–10878. PMLR.
- Evangelia Gogoulou, Timothée Lesort, Magnus Boman, and Joakim Nivre. 2023. Continual learning under language shift.
- Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc’Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.
- 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.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Danny Hernandez, Tom Brown, Tom Conerly, Nova DasSarma, Dawn Drain, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Tom Henighan, Tristan Hume, et al. 2022. Scaling laws and interpretability of learning from repeated data. *arXiv preprint arXiv:2205.10487*.
- Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022a. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.
- Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Thomas Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karén Simonyan, Erich Elsen, Oriol Vinyals, Jack Rae, and Laurent Sifre. 2022b. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*, volume 35, pages 30016–30030. Curran Associates, Inc.
- Adam Ibrahim, Benjamin Thérien, Kshitij Gupta, Mats L. Richter, Quentin Anthony, Timothée Lesort, Eugene Belilovsky, and Irina Rish. 2024. Simple and scalable strategies to continually pre-train large language models.

- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293.
- Tannon Kew, Florian Schottnann, and Rico Sennrich. 2023. Turning english-centric LLMs into polyglots: How much multilinguality is needed?
- Denis Kocetkov, Raymond Li, Loubna Ben allal, Jia LI, Chenghao Mou, Yacine Jernite, Margaret Mitchell, Carlos Muñoz Ferrandis, Sean Hughes, Thomas Wolf, Dzmitry Bahdanau, Leandro Von Werra, and Harm de Vries. 2023. The stack: 3 TB of permissively licensed source code. *Transactions on Machine Learning Research*.
- Teven Le Scao, Angela Fan, Christopher Akiki, Elie 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.
- Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, Jenny Chim, Qian Liu, Evgenii Zheltonozhskii, Terry Yue Zhuo, Thomas Wang, Olivier Dehaene, Mishig Davaadorj, Joel Lamy-Poirier, João Monteiro, Oleh Shliazhko, Nicolas Gontier, Nicholas Meade, Armel Zebaze, Ming-Ho Yee, Logesh Kumar Umapathi, Jian Zhu, Benjamin Lipkin, Muhtasham Oblokulov, Zhiruo Wang, Rudra Murthy, Jason Stillerman, Siva Sankalp Patel, Dmitry Abulkhanov, Marco Zocca, Manan Dey, Zhihan Zhang, Nour Fahmy, Urvashi Bhattacharyya, Wenhao Yu, Swayam Singh, Sasha Luccioni, Paulo Villegas, Maxim Kunakov, Fedor Zhdanov, Manuel Romero, Tony Lee, Nadav Timor, Jennifer Ding, Claire Schlesinger, Hailey Schoelkopf, Jan Ebert, Tri Dao, Mayank Mishra, Alex Gu, Jennifer Robinson, Carolyn Jane Anderson, Brendan Dolan-Gavitt, Danish Contractor, Siva Reddy, Daniel Fried, Dzmitry Bahdanau, Yacine Jernite, Carlos Muñoz Ferrandis, Sean Hughes, Thomas Wolf, Arjun Guha, Leandro von Werra, and Harm de Vries. 2023. Starcoder: may the source be with you!
- Bill Yuchen Lin, Abhilasha Ravichander, Ximing Lu, Nouha Dziri, Melanie Sclar, Khyathi Chandu, Chandrabhagavatula, and Yejin Choi. 2024. The unlocking spell on base llms: Rethinking alignment via in-context learning. In *International Conference on Learning Representations*.
- Stephanie Lin, Jacob Hilton, and Owain Evans. 2022a. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252.
- Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O’Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. 2022b. Few-shot learning with multilingual language models.
- Anton Lozhkov, Raymond Li, Loubna Ben Allal, Federico Cassano, Joel Lamy-Poirier, Nouamane Tazi, Ao Tang, Dmytro Pykhtar, Jiawei Liu, Yuxiang Wei, Tianyang Liu, Max Tian, Denis Kocetkov, Arthur Zucker, Younes Belkada, Zijian Wang, Qian Liu, Dmitry Abulkhanov, Indraneil Paul, Zhuang Li, Wen-Ding Li, Megan Risdal, Jia Li, Jian Zhu, Terry Yue Zhuo, Evgenii Zheltonozhskii, Nii Osa Osa Dade, Wenhao Yu, Lucas Krauß, Naman Jain, Yixuan Su, Xuanli He, Manan Dey, Edoardo Abati, Yekun Chai, Niklas Muennighoff, Xiangru Tang, Muhtasham Oblokulov, Christopher Akiki, Marc Marone, Chenghao Mou, Mayank Mishra, Alex Gu, Binyuan Hui, Tri Dao, Armel Zebaze, Olivier Dehaene, Nicolas Patry, Canwen Xu, Julian McAuley, Han Hu, Torsten Scholak, Sebastien Paquet, Jennifer Robinson, Carolyn Jane Anderson, Nicolas Chapados, Mostofa Patwary, Nima Tajbakhsh, Yacine Jernite, Carlos Muñoz Ferrandis, Lingming Zhang, Sean Hughes, Thomas Wolf, Arjun Guha, Leandro von Werra, and Harm de Vries. 2024. Starcoder 2 and the stack v2: The next generation.
- Risto Luukkainen, Ville Komulainen, Jouni Luoma, Anni Eskelinen, Jenna Kanerva, Hanna-Mari Kupari, Filip Ginter, Veronika Laippala, Niklas Muennighoff, Aleksandra Piktus, Thomas Wang, Nouamane Tazi, Teven Scao, Thomas Wolf, Osmo Suominen, Samuli Sairanen, Mikko Merioksa, Jyrki Heinonen, Aija Vah-tola, Samuel Antao, and Sampo Pyysalo. 2023. FinGPT: Large generative models for a small language. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2710–2726. Association for Computational Linguistics.
- Aman Madaan, Shuyan Zhou, Uri Alon, Yiming Yang, and Graham Neubig. 2022. Language models of code are few-shot commonsense learners. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1384–1403.
- MosaicML. 2023. Introducing MPT-30B: Raising the bar for open-source foundation models.
- Niklas Muennighoff, Alexander Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin A Raffel. 2024. Scaling data-constrained language models. *Advances in Neural Information Processing Systems*, 36.
- Aleksandar Petrov, Emanuele La Malfa, Philip HS Torr, and Adel Bibi. 2023. Language model tokenizers introduce unfairness between languages. *arXiv preprint arXiv:2305.15425*.
- Jonas Pfeiffer, Naman Goyal, Xi Lin, Xian Li, James Cross, Sebastian Riedel, and Mikel Artetxe. 2022.

- Lifting the curse of multilinguality by pre-training modular transformers. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3479–3495, Seattle, United States. Association for Computational Linguistics.
- Ofir Press, Noah A Smith, and Mike Lewis. 2021. Train short, test long: Attention with linear biases enables input length extrapolation. *arXiv preprint arXiv:2108.12409*.
- Sampo Pyysalo, Jenna Kanerva, Antti Virtanen, and Filip Ginter. 2021. WikiBERT models: Deep transfer learning for many languages. *NoDaLiDa 2021*, page 1.
- Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.
- Phillip Rust, Ivan Pfeiffer, Jonas an Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. How good is your tokenizer? on the monolingual performance of multilingual 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 (Volume 1: Long Papers)*, pages 3118–3135.
- Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Winogrande: An adversarial winograd schema challenge at scale. *arXiv preprint arXiv:1907.10641*.
- Nikhil Sardana and Jonathan Frankle. 2023. Beyond chinchilla-optimal: Accounting for inference in language model scaling laws. *arXiv preprint arXiv:2401.00448*.
- Daria Soboleva, Faisal Al-Khateeb, Robert Myers, Jacob R Steeves, Joel Hestness, and Nolan Dey. 2023. SlimPajama: A 627B token cleaned and deduplicated version of RedPajama.
- Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Harsh Jha, Sachin Kumar, Li Lucy, Xinxu Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Abhilasha Ravichander, Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Pete Walsh, Luke Zettlemoyer, Noah A. Smith, Hannaneh Hajishirzi, Iz Beltagy, Dirk Groeneveld, Jesse Dodge, and Kyle Lo. 2024. Dolma: an Open Corpus of Three Trillion Tokens for Language Model Pretraining Research. *arXiv preprint*.
- Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*.
- Jörg Tiedemann. 2009. News from OPUS-a collection of multilingual parallel corpora with tools and interfaces. In *Recent advances in natural language processing*, volume 5, pages 237–248.
- Jörg Tiedemann. 2020. The Tatoeba Translation Challenge – Realistic data sets for low resource and multilingual MT. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1174–1182, Online. Association for Computational Linguistics.
- Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT — Building open translation services for the World. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation (EAMT)*, Lisbon, Portugal.
- Together Computer. 2023. Redpajama: An open source recipe to reproduce llama training dataset.
- 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.
- María Ubierna, Cristina Díez Santos, and Sara Mercier-Blais. 2022. Water security and climate change: hydropower reservoir greenhouse gas emissions. *Water Security Under Climate Change*, pages 69–94.
- 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.
- David Vilar, Markus Freitag, Colin Cherry, Jiaming Luo, Viresh Ratnakar, and George Foster. 2023. Prompting palm for translation: Assessing strategies and performance. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15406–15427.
- Pablo Villalobos, Jaime Sevilla, Lennart Heim, Tamay Besiroglu, Marius Hobbhahn, and Anson Ho. 2022. Will we run out of data? an analysis of the limits of scaling datasets in machine learning. *arXiv preprint arXiv:2211.04325*.
- Xiangpeng Wei, Haoran Wei, Huan Lin, Tianhao Li, Pei Zhang, Xingzhang Ren, Mei Li, Yu Wan, Zhiwei Cao, Binbin Xie, et al. 2023. PolyLM: An open source polyglot large language model. *arXiv preprint arXiv:2307.06018*.
- Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800. Association for Computational Linguistics.

Jun Zhao, Zhihao Zhang, Luhui Gao, Qi Zhang, Tao Gui, and Xuanjing Huang. 2024. LLaMA beyond english: An empirical study on language capability transfer.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2023. Multilingual machine translation with large language models: Empirical results and analysis.

A Appendix

A.1 Training details

It has been our aim throughout this work to release Poro 34B fully openly, including model weights, pretraining configuration, the pretraining and evaluation data, and all associated scripts and tools. We provide here additional details of these to facilitate accurate reproduction of our work. The pretraining data sources are detailed in Table 4, and the model and pretraining hyperparameters in Table 5.

Dataset	Language	Reference
SlimPajama	English	https://huggingface.co/datasets/cerebras/SlimPajama-627B
Starcode	Code	https://huggingface.co/datasets/bigcode/starcode
Tatoeba challenge	Eng-Fin	https://huggingface.co/datasets/tatoeba
Project Gutenberg	English	https://huggingface.co/datasets/allenai/dolma
Parsebank	Finnish	https://turkunlp.org/finnish_nlp.html
mC4	— —	https://huggingface.co/datasets/mc4
CC-Fi	— —	https://github.com/TurkuNLP/CC-Fi
Fiwiki	— —	https://fi.wikipedia.org/wiki
Lönnrot	— —	http://www.lonnrot.net
Suomi24	— —	http://urn.fi/urn:nbn:fi:lb-2021101527
Reddit-Fi	— —	https://www.reddit.com/r/Suomi
STT	— —	http://urn.fi/urn:nbn:fi:lb-2019041501
Yle	— —	http://urn.fi/urn:nbn:fi:lb-2017070501
Yle	— —	http://urn.fi/urn:nbn:fi:lb-2021050401
Yle	— —	http://urn.fi/urn:nbn:fi:lb-2019050901
Yle	— —	http://urn.fi/urn:nbn:fi:lb-2021050701

Table 4: Data sources

<i>Architecture hyperparameters</i>		<i>Pretraining hyperparameters</i>	
Parameters	34B	Global Batch Size	2048
Precision	bfloat16	Learning rate	1.5e-4
Layers	54	Total tokens	1000B
Hidden dim	7168	Warmup tokens	10B
Attention heads	56	Decay tokens	1000B
Vocab size	131072	Decay style	cosine
Sequence length	2048	Min. learning rate	2e-5
Activation	GELU	Adam (β_1, β_2)	(0.9, 0.95)
Position embedding	ALiBi	Weight decay	2e-5
Tied embeddings	True	Gradient clipping	1.0

Table 5: Model and training hyperparameters

A.2 Detailed benchmark results

Tables 6, 7, and 8 show the detailed benchmark results for Finnish, English, and code.

Benchmark	Poro 34B	Llama 33B	MPT-30b	Falcon-40b	FinGPT 8B	FinGPT 13B	Starcoder
Analogies	77.69	61.54	57.69	43.85	40.0	36.15	46.15
Arithmetic	54.28	47.74	57.25	51.06	41.96	45.23	48.41
Cause and Effect	67.97	60.78	58.82	46.41	66.01	69.28	54.90
Emotions	55.00	45.00	39.37	16.88	45.62	38.75	23.13
Empirical Judg.	62.63	43.43	43.43	34.34	32.32	36.36	44.44
General Knowl.	75.71	48.57	37.14	22.86	51.43	40.00	22.86
Intent Recogn.	83.24	77.75	77.31	46.24	51.43	58.24	65.03
Misconceptions	53.73	51.49	50.00	50.00	51.45	45.52	47.01
Paraphrase	58.50	53.00	52.50	54.50	49.50	45.50	47.50
Sentence Ambig.	66.67	45.00	56.67	48.33	48.33	53.33	51.67
Similarities Abst.	73.68	52.63	55.26	53.95	68.42	69.74	50.00
Average	66.28	53.36	53.22	42.58	49.69	48.92	45.55

Table 6: FIN-Bench Finnish benchmark results

Benchmark	Poro 34B	Llama 33B	MPT-30b	Falcon-40b	FinGPT 8B	FinGPT 13B	Starcoder
ARC-Challenge	53.16	61.61	55.80	50.51	25.34	24.31	30.29
Hellaswag	77.77	84.64	82.23	77.01	42.91	46.77	47.22
MMLU	46.29	58.13	47.27	46.13	23.34	23.64	32.11
TruthfulQA	41.66	42.84	38.44	41.64	43.80	44.58	40.06
Winogrande	72.77	80.27	74.82	81.53	53.19	57.53	54.85
GSM8K	11.75	32.27	17.13	2.43	0.22	0.22	8.11
Average	50.57	59.96	52.62	49.87	31.47	32.85	35.44

Table 7: English benchmark results

Benchmark	Category	Poro 34B	Llama 33B	MPT-30b	Falcon-40b	Starcoder
HumanEval	Python	37.20	34.15	35.37	34.15	45.12
MBPP	Python	47.40	41.20	43.00	43.00	53.00
Average		41.80	37.67	39.18	38.57	49.06

Table 8: Code benchmark results

A.3 Hardware

Poro 34B was trained on the LUMI-G GPU partition of the LUMI supercomputer, located in Finland. LUMI is, at the time of this writing, the third fastest supercomputer in Europe, and the 8th fastest in the world (<https://www.top500.org/>). LUMI is also ranked 7th greenest by the Green500 list (<https://www.top500.org/lists/green500/>).

The LUMI-G partition has 2978 nodes, with each node having four AMD MI250x GPUs with 128GB of memory each, and a single 64-core CPU. The MI250x is a multi-chip module (MCM), with dual-GCD (graphics compute die) design, which in practice means a node has eight logical devices, each logical device with access to 64GB of high bandwidth memory.

Each node has four 200Gbps Slingshot-11 network interconnects. The nodes are connected together in a dragonfly topology. During benchmarking and scale testing we did not observe the network topology as a limiting factor for the required collective operation sizes. The total of 800 Gbps per-node bandwidth proved to be more than sufficient, and the communication overhead was minimal during training.

Can summarization approximate simplification? A gold standard comparison

Giacomo Magnifico Eduard Barbu

Institute of Computer Science

University of Tartu

Estonia

{giacomo.magnifico, eduard.barbu}@ut.ee

Abstract

This study explores the overlap between text summarization and simplification outputs. While summarization evaluation methods are streamlined, simplification lacks cohesion, prompting the question: how closely can abstractive summarization resemble gold-standard simplification? We address this by applying two BART-based BRIO summarization methods to the Newsela corpus, comparing outputs with manually annotated simplifications and achieving a top ROUGE-L score of 0.654. This provides insight into where summarization and simplification outputs converge and differ.

1 Introduction

Text simplification can operate at various linguistic levels—semantic, syntactic, or lexical—using diverse strategies to achieve specific goals (Pellow and Eskenazi, 2014; Paetzold and Specia, 2016; Chen et al., 2017; Van et al., 2021). In practice, Automatic Text Simplification (ATS) transforms complex text into simpler versions by splitting sentences, shortening length, and simplifying vocabulary and grammar. The best English-language ATS models rely on parallel corpora like WikiSmall (Zhu et al., 2010; Zhang and Lapata, 2017), aligning complex and simple sentences from standard and Simple English Wikipedias (originally 108,000 instances from 65,133 articles, currently 89,042). The most valuable resource for text simplification is the Newsela corpus Xu et al. (2015), which includes 9,565 news articles professionally rewritten at multiple reading levels, with 1,913 original articles and four levels of simplification. However, it lacks the volume needed to train advanced deep-learning models effectively.

Simplification lacks standardized procedures and a common algorithm, partly due to the absence

of a “native speaker of simplified language” (Siddharthan, 2014). The subjective nature of simplification also makes consistent methodology difficult (Grabar and Saggion, 2022). The evaluation metrics for simplification are similarly inconsistent. Some, like BLEU or Levenshtein distance (Levenshtein, 1965; Papineni et al., 2002), focus on intrinsic grammatical features and struggle with semantic changes, while others, such as cosine distance, emphasize semantic similarity. By contrast, summarization metrics are well-established, even when imperfectly applied (Grusky, 2023). Furthermore, while the two tasks present some divergences in their focus (e.g. the relevance of information ordering, the choice of domain-agnostic lexicon, and the preference for short active forms instead of long passive forms), they remain convergent in producing shorter and poignant text. Given the state of things, we believe that comparing simplification with summarization could provide insights into their convergence.

This study investigates whether a state-of-the-art (SotA) summarization system can approximate manual simplification by comparing annotated simplifications with automated summarization. Starting with Newsela’s English documents, we process original articles with BRIO (Liu et al., 2022), a SotA abstractive summarizer, applying document-wide and paragraph-by-paragraph summarization methods. We then evaluate each output set against the four simplification levels using ROUGE-L scores to measure similarity. Results indicate an average performance difference of 0.444, with paragraph-by-paragraph summarization achieving the highest score (0.654) at level 1, gradually decreasing through levels 2 to 4. While paragraph-by-paragraph summarization does not equate to manual simplification, it may serve as an effective preparatory step for manual annotators.

Background and related research are discussed

in Section 2, with the experimental setup and findings detailed in Sections 3 and 4. A summary of the presented work, followed by the limits of the scope and suggestions for future research, are provided in Section 5.

2 Related work

The multifaceted nature of implementing text simplification has led to multiple works that share the goal of rewriting complex documents with simpler, more straightforward language. This is ultimately achieved by modifying the original text both lexically and syntactically as defined in Truică et al. (2023), either in an automated or manual way. Multiple works in the field have tackled different applications, from aiding people with disabilities (Rello et al., 2013; Chen et al., 2017), low-literacy adults (Watanabe et al., 2009; Paetzold and Specia, 2016), non-native learners (Allen, 2009; Pellow and Eskenazi, 2014) to auxiliary systems to improve the effectiveness of other NLP tasks (Stymne et al., 2013; Wei et al., 2014; Štajner and Popovic, 2016).

Due to the wide range of applications, a major subjectivity issue emerges when evaluating the different methods for simplification (Grabar and Saggion, 2022). Different scoring methods that have been utilized for simplification include: *BLEU* (Papineni et al., 2002); *TERp*, Translation Edit Rate plus, which computes the number of the three edit operations plus the inverse (Snover et al., 2009); *OOV*, Out Of Vocabulary, which measures the rate of oov words from a chosen simple vocabulary (e.g. Basic English list) (Vu et al., 2014); *changed*, measuring the percentage of the test examples where the system suggested some change (Horn et al., 2014); *potential*, computing the proportion of instances in which at least one of the candidates generated is in the gold-standard (Paetzold and Specia, 2016); *SARI*, the most recent, which performs a similar comparison to BLEU but is considered more reliable (Xu et al., 2016).

The general approach to text summarization is more streamlined, aiming to produce a shorter text than the input one while keeping all relevant information, defined as abstract or summary (Moiyadi et al., 2016). The most common approaches are naïve Bayes (Kupiec et al., 1995; Gambhir and Gupta, 2017), swarm algorithms (Jarraya and Bouri, 2012; Izakian and Mesgari, 2015), and sequence-to-sequence models (Sutskever et al.,

2014; Zhang et al., 2020).

A further distinction can be made between abstractive and extractive summarization methods (Nazari and Mahdavi, 2019). Where extractive methods produce text by concatenating selected parts of the original document, abstractive methods apply language generation techniques to produce a shorter document (Jeæek and Steinberger, 2008; Gupta and Lehal, 2010). Standard scoring methods for text summarisation are precision/recall measures and various instances of *ROUGE* (Lin and Och, 2004a; Grusky, 2023), some examples being *ROUGE-n*, *ROUGE-L*, and the most recent *ROUGE-SEM* (Zhang et al., 2024).

The Newsela corpus is a collection of 1,130 articles rewritten and simplified by professional editors, aimed at children of different grade levels (Xu et al., 2015). From each individual article, four different versions have been derived through manual simplification process and labelled with a number from 1 to 4, representative of the level of simplification. Label 4 represents the most simplified output, suitable for a 3rd grader; label 3 represents an output suitable for a 4th grader; labels 2 and 1 identify outputs suitable for 6th and 7th graders. The original articles are suitable for 12th graders.

Considering possible modifications to the dataset past the authors' presentation of their work, the corpus currently consists of 9,565 documents, of which 1,913 original articles.

3 Experimental setup

For the purpose of this work, the architecture chosen to perform the summarization procedure was BRIO, a system presented in Liu et al. (2022) and based both on the BART architecture (Lewis et al., 2020) and the PEGASUS architecture (Zhang et al., 2020). The choice was motivated by its state-of-the-art performance in summarization tasks, its ease of availability and implementation, and the double-model-based system that it employs. The dual nature of BRIO is the result of fine-tuning two different architectures on two different datasets with a specific training paradigm. Since the two datasets were characterized by longer texts (Hermann et al., 2015) and shorter texts (Narayan et al., 2018), the two backbones for the architecture keep these properties. Therefore, the BART-based BRIO was chosen as a

summarizer for its performance with longer texts, as suggested by the original authors.

The original articles from the Newsela corpus were then processed through the summarization model. For each article, two procedures were followed to produce different output documents: document-wide summarization and paragraph-by-paragraph summarization, as explained below. A graphic representation of the general procedure is provided in Figure 1

Document-wide. The more intuitive application of text summarization, this method involved the generation of a single string containing the whole text by joining the various paragraphs and subsequently processing it with the summarizer model. Once the architecture produced an output string, it was written in a separate `*.txt` file.

Paragraph-by-paragraph. This summarization approach stems from the visual structure of academic texts, which usually separate topics and changes in content by dividing the document into paragraphs. Thus, the intuition was to make the architecture follow a similar pattern to preserve the content and produce a more effective summarization. This method implemented splitting the original text into paragraphs and processing each paragraph separately with the summarizer model. The resulting outputs were subsequently rejoined and written as a single document in a separate `*.txt` file.

Both procedures were applied to each of the original 1,913 English articles in the Newsela corpus, and the resulting two sets of summarized documents were compared to the simplified version produced by the editors. This was done by iterating through the different levels of simplification (1, 2, 3, and 4) and calculating the *precision*, *recall* and *ROUGE F1* score between each simplified version of the document and the summarized version of it. The resulting evaluation was stored, and the average was calculated level-wise for each metric with the scores from the whole set. Then, the scoring procedure was repeated for the remaining summarized set. The chosen evaluation score was *ROUGE-L* as it was both a part of the original BRIO publication (Liu et al., 2022) and a statistic based on Long Common Sequence (LCS) (Lin and Och, 2004b), which made it well suited to measure the grammatical integrity, keyword conservation and coherence in the summarized texts.

4 Results

The average scores for the three evaluation metrics used in comparing the human-produced simplification and the automated summarization are available in Table 1. To provide an easier analysis, the scores have been divided by the level of simplification taken under scrutiny and the type of summarization procedure performed on the original articles. The upper section of Table 1 provides the average evaluation score between all the documents summarized with the first method mentioned in Section 3 and their simplified equivalent for each level. The second summarization method, paragraph-by-paragraph, is evaluated in the lower part of the Table.

Level	Precision	Recall	ROUGE-L
DOCUMENT			
label 1	0.058	0.918	0.109
label 2	0.061	0.884	0.113
label 3	0.066	0.811	0.122
label 4	0.078	0.731	0.141
PARAGRAPH			
label 1	0.731	0.615	0.654
label 2	0.721	0.561	0.616
label 3	0.703	0.461	0.541
label 4	0.699	0.354	0.451

Table 1: Average precision, recall and ROUGE-L scores when comparing the summarization output against the different levels of manually simplified articles. The table is divided according to the two types of summarization techniques presented, *document-level* and *paragraph-level*.

To make the gap in scores and the variability in summarization performance through the different processes more apparent, two graphic representations of the average scores are provided in Figure 2. The data corresponds to the document-wide summarization method on the left side and the paragraph-by-paragraph method on the right.

When comparing the results from the two processes, the overall difference in balance between precision/recall for the document-wide summarization method is immediately noticeable. Even considering the progressive improvement of the precision rate and the lowering of the recall score, the minimum gap between the two is 0.653. The first hypothesis was that it was due to the summarizer generating lengthy and repetitive summaries;

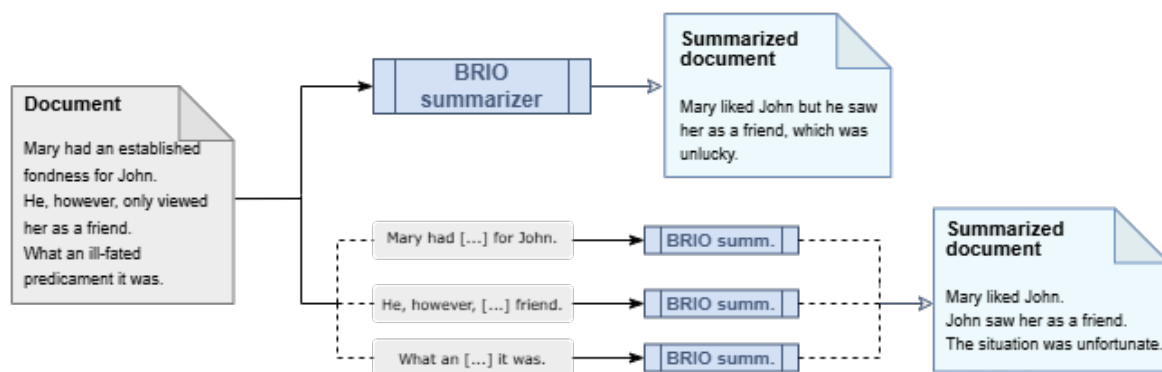


Figure 1: Representation of the processing pipeline for each article, showing the document-wide method (upper side) and paragraph-by-paragraph (lower side).

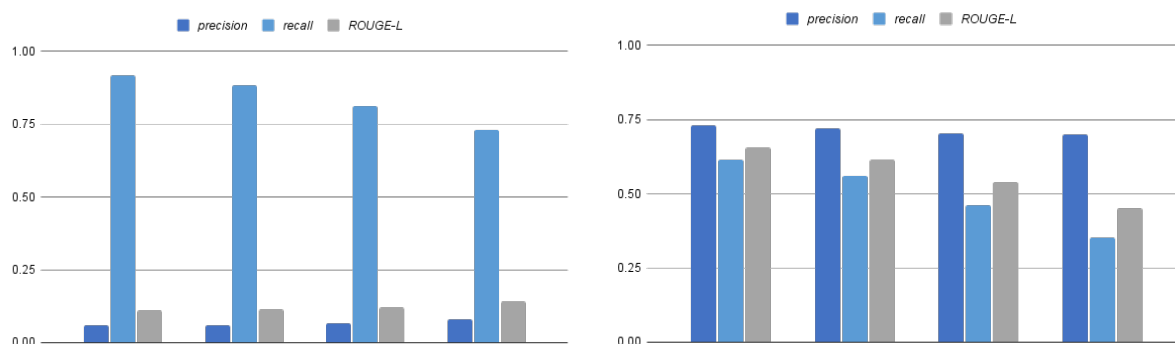


Figure 2: Comparison between the different levels of simplified text (1 to 4, left to right) and the two automated types of summarization. On the left is the performance of the document-wide summarization, on the right the performance of the paragraph-by-paragraph method.

however, a quick analysis of the outputs confirmed the variety in length and the production of shorter documents than their input. Therefore, the more plausible hypothesis is that while the longest common sentences between manual simplification and automated summarization are recalled in the text (most likely the keywords), the structural lexicon and syntactical choices of the simplified version would not appear through document-wide summarization. Consequently, this can lead to the poor similarity between the two document types and the convolution of information through summarization, a hypothesis corroborated by the low *ROUGE-L* score.

On the other side of Figure 2, the scores provide a better-looking picture of the paragraph-by-paragraph performance. With a *ROUGE-L* score of 0.566 averaged between all levels of simplification, the similarity between the simplified and summarized versions is noticeable. Although they perform better when compared to lower levels

of simplification than to more simplified documents, the summarized outputs obtained through paragraph-by-paragraph processing perform well enough to justify further investigation and analysis. Our hypothesis for the better performance of the paragraph-by-paragraph, when compared to the document-wide processing, lies in the nature of the process: a block-by-block iteration might be more similar to the manually performed annotation than a text-wide transformation is.

Worth of notice for the production of these results was the difference in time requirements between the first summarization method and the second when operating on an average machine (16 GB RAM, 8 cores, 2,90 GHz CPU). The time elapsed for the paragraph-by-paragraph processing method was greatly increased, ranging between 10x and 50x more for each iteration and thus requiring several minutes instead of seconds. While the reason behind this issue requires more investigation, with the current implementation,

performing such a method on a large-scale dataset without some optimization or access to a powerful machine is not recommended.

5 Conclusions, limitations and future work

In this work, the similarities between simplified and summarized text have been analysed through the automated summarization of articles from the Newsela corpus, performed with two different methods and compared to four levels of professional manual simplification representative of diverse school grade levels. By examining the results obtained by a ROUGE-L scoring comparison between our output and the manual standard, it is shown that the proposed paragraph-by-paragraph method is superior to a document-wide approach, with the highest score being 0.654. Hence, it is possible to claim that while automated summarization does not produce text similar enough to simplified documents to justify its substitution, it still produces text similar enough to be used as a baseline to perform simplification on - instead of starting from the original text.

However, there are important limitations to the currently chosen metric. As ROUGE-L cannot measure semantic similarity between instances, all sequences that are semantically correct but lexically different would not compute as "similar". Since abstractive summarization could generate text that is lexically different from the simplification golden standard but still effectively simplified, further analysis with semantically relevant metrics should be conducted. In addition, future work in this direction should implement ulterior thorough analyses with more refined metrics, such as ROUGE-SEM or SARI, along with a comparison between manual simplification, automated summarization and automated simplification algorithms. In particular, the latter could shed some light on the intrinsic similarities between simplification and summarization and help further investigate the potential interdisciplinary approaches to the text simplification field of research.

Further investigation into optimization procedures to make the most-performing methods available for lower-end machines should also be conducted to allow for wider access to the tools and improved effectiveness of summarizers as a simplification helping tool.

6 Acknowledgments

This research has been supported by the EKT B55 project "Teksti lihtsustamine eesti keeles".

References

- David Allen. 2009. A study of the role of relative clauses in the simplification of news texts for learners of english. *System*, 37:585–599.
- Ping Chen, John Rochford, David Kennedy, Soussan Djamasbi, Peter Fay, and Will Scott. 2017. Automatic text simplification for people with intellectual disabilities. In *Artificial Intelligence Science and Technology*, pages 725–731.
- Mahak Gambhir and Vishal Gupta. 2017. Recent automatic text summarization techniques: a survey. *Artificial Intelligence Review*, 47.
- Natalia Grabar and Horacio Saggion. 2022. Evaluation of Automatic Text Simplification: Where are we now, where should we go from here. In *Traitement Automatique des Langues Naturelles*, pages 453–463, Avignon, France. ATALA.
- Max Grusky. 2023. Rogue scores. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1914–1934, Toronto, Canada. Association for Computational Linguistics.
- Vishal Gupta and Gurpreet Lehal. 2010. A survey of text summarization extractive techniques. *Journal of Emerging Technologies in Web Intelligence*, 2.
- Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *ArXiv*, abs/1506.03340.
- Colby Horn, Cathryn Manduca, and David Kauchak. 2014. Learning a lexical simplifier using Wikipedia. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 458–463, Baltimore, Maryland. Association for Computational Linguistics.
- Z. Izakian and M. Mesgari. 2015. Fuzzy clustering of time series data: A particle swarm optimization approach. *Journal of AI and Data Mining*, 3(1):39–46.
- Bilel Jarraya and Abdelfatteh Bouri. 2012. Meta-heuristic optimization backgrounds: A literature review. *International Journal of Contemporary Business Studies*, 3:31–44.
- Karel Jeek and Josef Steinberger. 2008. Automatic text summarization (the state of the art 2007 and new challenges).

- 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*, SIGIR '95, page 68–73, New York, NY, USA. Association for Computing Machinery.
- Vladimir I. Levenshtein. 1965. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet physics. Doklady*, 10:707–710.
- 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 and FJ Och. 2004a. Looking for a few good metrics: Rouge and its evaluation. In *Ntcir workshop*.
- Chin-Yew Lin and Franz Josef Och. 2004b. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 605–612, Barcelona, Spain.
- Yixin Liu, Pengfei Liu, Dragomir Radev, and Graham Neubig. 2022. Brio: Bringing order to abstractive summarization.
- Hamza Shabbir Moiyadi, Harsh Desai, Dhairya Pawar, Geet Agrawal, and Nilesh M. Patil. 2016. Nlp based text summarization using semantic analysis. *International Journal of Advanced Engineering, Management and Science*, 2(10).
- Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.
- N. Nazari and M. A. Mahdavi. 2019. A survey on automatic text summarization. *Journal of AI and Data Mining*, 7(1):121–135.
- Gustavo Paetzold and Lucia Specia. 2016. Benchmarking lexical simplification systems. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3074–3080, Portorož, Slovenia. European Language Resources Association (ELRA).
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Annual Meeting of the Association for Computational Linguistics*.
- David Pellow and Maxine Eskenazi. 2014. An open corpus of everyday documents for simplification tasks. In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR)*, pages 84–93, Gothenburg, Sweden. Association for Computational Linguistics.
- Luz Rello, Ricardo Baeza-Yates, and Horacio Saggion. 2013. The impact of lexical simplification by verbal paraphrases for people with and without dyslexia. In *Proceedings of the 14th International Conference on Computational Linguistics and Intelligent Text Processing - Volume 2, CICLing'13*, page 501–512, Berlin, Heidelberg. Springer-Verlag.
- Advait Siddharthan. 2014. A survey of research on text simplification. *ITL - International Journal of Applied Linguistics*, 165:259–298.
- Matthew Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. 2009. Ter-plus: paraphrase, semantic, and alignment enhancements to translation edit rate. *Machine Translation*, 23:117–127.
- Sanja Štajner and Maja Popovic. 2016. Can text simplification help machine translation? In *Proceedings of the 19th Annual Conference of the European Association for Machine Translation*, pages 230–242.
- Sara Stymne, Jörg Tiedemann, Christian Hardmeier, and Joakim Nivre. 2013. Statistical machine translation with readability constraints. In *Proceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA 2013)*, pages 375–386. Linköping University Electronic Press, Sweden. 19th Nordic Conference of Computational Linguistics, NODALIDA 2013 ; Conference date: 22-05-2013 Through 24-05-2013.
- Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks.
- Ciprian-Octavian Truică, Andrei-Ionuț Stan, and Elena-Simona Apostol. 2023. Simplex: a lexical text simplification architecture. *Neural Computing and Applications*, 35(8):6265–6280.
- Hoang Van, Zheng Tang, and Mihai Surdeanu. 2021. How may I help you? using neural text simplification to improve downstream NLP tasks. *CoRR*, abs/2109.04604.
- Tu Thanh Vu, Giang Binh Tran, and Son Bao Pham. 2014. Learning to simplify children stories with limited data. In *Intelligent Information and Database Systems*, pages 31–41, Cham. Springer International Publishing.
- William Watanabe, Arnaldo Junior, Vinícius Uzêda, Renata Fortes, Thiago Pardo, and Sandra Aluisio. 2009. Facilita: Reading assistance for low-literacy readers. pages 29–36.

- Chih-Hsuan Wei, Robert Leaman, and Zhiyong lu. 2014. Simconcept: A hybrid approach for simplifying composite named entities in biomedical text. volume 19.
- Wei Xu, Chris Callison-Burch, and Courtney Napoles. 2015. Problems in current text simplification research: New data can help. *Transactions of the Association for Computational Linguistics*, 3:283–297.
- Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.
- Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 11328–11339. PMLR.
- Ming Zhang, Chengzhang Li, Meilin Wan, Xuejun Zhang, and Qingwei Zhao. 2024. Rouge-sem: Better evaluation of summarization using rouge combined with semantics. *Expert Systems with Applications*, 237:121364.
- Xingxing Zhang and Mirella Lapata. 2017. Sentence simplification with deep reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 584–594, Copenhagen, Denmark. Association for Computational Linguistics.
- Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. 2010. A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1353–1361, Beijing, China. Coling 2010 Organizing Committee.

A Comparative Study of PEFT Methods for Python Code Generation

Johanna Männistö Joseph Attieh Jörg Tiedemann
Department of Digital Humanities, University of Helsinki
{first.last}@helsinki.fi

Abstract

Fine-tuning language models incurs high costs in training, inference and storage. Parameter-efficient fine-tuning (PEFT) methods have emerged as a more cost-effective alternative to full fine-tuning. However, limited work has compared different PEFT approaches for tasks like code generation. In this study, we examine the effect of various PEFT training methods on model performance in the task of Python code generation. We fine-tune four model families, ranging from 124M to 7B parameters, using three PEFT approaches alongside standard full fine-tuning. Our findings reveal that the effectiveness of each PEFT method varies with the model size and the corpus used.

1 Introduction

Language models (LMs) have shown great capabilities across a variety of natural language processing (NLP) downstream tasks, including code generation tasks (Chen et al., 2021; Li et al., 2023; Nijkamp et al., 2023; Rozière et al., 2023a; Xu et al., 2022). Generally, larger LMs tend to perform better on downstream tasks (Kaplan et al., 2020), as evidenced by CodeLlama, which exhibits improved code completion and generation abilities as its size increases from 7 billion to 70 billion parameters (Rozière et al., 2023a). However, the training of these larger models is resource-intensive, requiring substantial computational power and high storage costs.

To address these challenges, Parameter-Efficient Fine-Tuning (PEFT) methods have emerged (Dettmers et al., 2023; Houlisby et al., 2019; Hu et al., 2022; Lester et al., 2021; Lialin et al., 2023; Liu et al., 2022). These approaches update a small subset of the model parameters

during fine-tuning, while the rest remain frozen, significantly reducing both computational and storage costs for each downstream task.

While significant research has been conducted on both PEFT methods and code LMs individually, at the time of this study, there is only limited research evaluating PEFT approaches applied to code LMs for code generation tasks (Purnawansyah et al., 2024; Weyssow et al., 2023; Zhuo et al., 2024). Existing studies on this topic have notable shortcomings: many focus only on smaller models, ignoring those with 1B parameters or more (Ayupov and Chirkova, 2022; Zou et al., 2023), while others concentrate solely on tasks like code understanding or clone detection, which often outperform code generation tasks under similar PEFT training conditions (Liu et al., 2023; Wang et al., 2023; Zou et al., 2023). These limitations highlight a significant research gap, particularly as state-of-the-art models increasingly feature billions of parameters and are predominantly generative.

We aim to fill existing research gaps through two key questions: 1) Which PEFT method delivers the best performance across various model sizes for Python generation tasks? 2) How do these methods compare to full fine-tuning?

2 Parameter Efficient Fine-Tuning

Parameter efficient fine-tuning (PEFT) methods provide a more efficient alternative to full fine-tuning of large LMs (LLMs), significantly reducing both computational and storage costs (Lialin et al., 2023). Various PEFT methods achieve remarkable performance compared to full fine-tuning for models of different sizes (Ding et al., 2023; Lester et al., 2021; Wang et al., 2023), all while offering substantial computational savings. We present an overview of the three methods that are relevant to our work. These methods are illustrated in Figure 1.

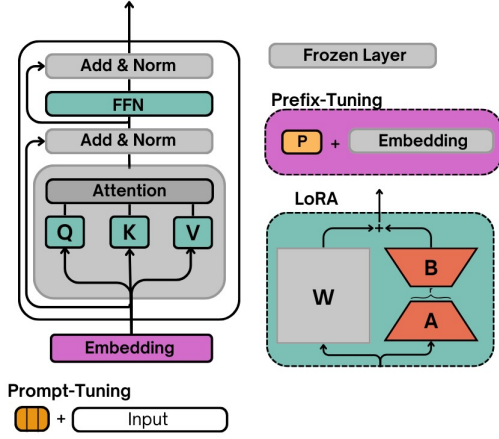


Figure 1: Diagram showing a decoder layer, as well the PEFT Techniques employed in the study.

Low Rank Adaptation (LoRA) LoRA (Hu et al., 2022) approximates model weight matrices through low-rank decomposition into a smaller set of parameters. The pretrained weights are frozen, and the approximation is fine-tuned during training. LoRA can be applied to any weight matrix, and Dettmers et al. (2023) shows that applying it to all linear layers enhances performance compared to limiting it to query and value matrices as done in Hu et al. (2022). The efficiency of LoRA is determined by the rank of the decomposed matrices and the scaling factor, alpha. Alpha is often set to be twice the size of the rank (Zhuo et al., 2024; Weyssow et al., 2023) or equivalent to the rank (Lee et al., 2023).

Prefix-tuning Inspired by in-context learning, this method (Li and Liang, 2021) prepends trainable tensors called "soft prompts" to the input of each transformer block. These task-specific prefixes are updated during training while the original model parameters are frozen.

Prompt-tuning Similar to prefix-tuning, prompt-tuning (Lester et al., 2021) adds trainable parameters to the input layer only, leading to a further reduction in the number of parameters that need updating compared to prefix-tuning.

3 Experimental Approach

In this section, we describe the methodology used to investigate how models of different sizes adapt to the Python code generation task using PEFT.

The experimental approach is illustrated in Figure 2, which outlines the models, dataset, data processing methods, training setup, and evaluation strategy. These elements will be described in detail in the following section.

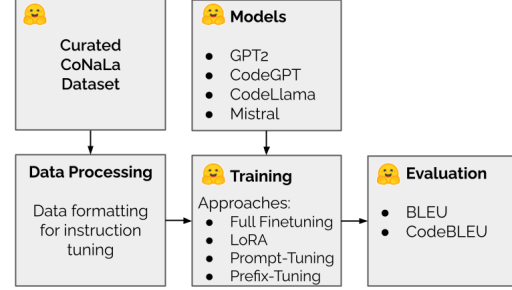


Figure 2: Diagram describing the experimental approach adopted in this study.

3.1 Models

In this study, we strategically select four distinct model families, mainly GPT-2, CodeGPT, CodeLlama, and Mistral v0.1¹.

We selected the models with sizes ranging from 124M to 7B parameters and trained them on either text, code, or both. This enable us to explore model sizes that have been overlooked in similar studies.

GPT-2 (Radford et al., 2019) Autoregressive models ranging from 124M to 1.5B. The study employs GPT-2, GPT-2 M, L, and XL.

CodeGPT (Lu et al., 2021) is initialized from GPT-2 and fine-tuned on code corpora. The study focuses on the Python variants of the models, using both adapted and small versions².

CodeLlama (Rozière et al., 2023b) Available in three sizes (7B, 13B, and 34B) and three variants. Only the 7 billion parameter base model was fine-tuned for this study.

Mistral Mistral v.01 (Jiang et al., 2023) A 7B autoregressive model trained on open-source text and code data, with no training datasets listed. At the time of this study, Mistral did not support prefix-tuning.

¹This model was the latest release at the time of the study. It was selected as it is trained on both text and code.

²The adapted is trained using the same tokenizer as GPT-2 and the small uses another newly trained BPE tokenizer.

3.2 Datasets

The study utilizes the CoNaLa dataset (Yin et al., 2018), consisting of 2,379 natural language-code pairs for training and 500 pairs for testing. This dataset is derived from the larger CoNaLa-mined dataset, initially sourced from Stack Overflow. For training, we use the `rewritten_intent` field, which contains the natural language instruction (i.e., Python problem), and the `snippet` field, which provides the corresponding Python code solution. As the dataset was already curated for quality by annotators, no additional filtering was conducted prior to training.

We formatted the data for model input by adding indicator prompts `### Instruction:` before the `rewritten_intent` and `### Response:` before the `snippet`, followed by a newline separator³. An example from the processed dataset can be seen in Table 1.

Rewritten Intent	<code>### Instruction:</code> How can I send a signal from a Python program?
Snippet	<code>### Response:</code> <code>os.kill(os.getpid(), signal.SIGUSR1)</code>

Table 1: Example from the CoNaLa dataset showing the structure of processed training data.

3.3 Training Setup

The implementation relies on the following libraries: HuggingFace transformers (Wolf et al., 2020), TRL (Werra et al., 2020) and PEFT (Manrulkar et al., 2022). We perform the training using HuggingFace’s SFTTrainer. The training arguments were selected to be the same as the reported hyperparameters for each model whenever feasible; otherwise, we pick hyperparameters and empirically validate them to ensure a reliable baseline for our experiments.

The models were given packed⁴ input sequences of length 1024, which included any additional prefix or prompt tokens when needed, and were separated by an EOS (end-of-sequence) token. This value was selected due to GPU memory limitations. As done by Shi et al. (2024), we include the entire instruction-response set in the loss calculation rather than masking the instruc-

³This structure follows the Stanford Alpaca.

⁴Packed input sentences combine multiple sequences into a single one separated by end-of-sequence token, to maximize training efficiency.

tions, as this approach can enhance performance with smaller datasets.

We apply LoRA to all linear layers of the model following Dettmers et al. (2023), and we set the rank to 16 and alpha to 32. Experiments by Lester et al. (2021) on prompt length demonstrated that only marginal gains were achieved when prompts exceeded 20 tokens, motivating the use of just 20 tokens for prompt-tuning and prefix-tuning.

3.4 Evaluation

We evaluate the models on the CoNaLa dataset using BLEU-4 (Papineni et al., 2002) and CodeBLEU (Ren et al., 2020). For both metrics, 1.0 is the highest score. To generate the predictions, we use a temperature of 0.2 and nucleus sampling (Holtzman et al., 2020) with top p = 0.95. All models are loaded using BF16 for inference.

4 Discussion

Table 2 summarizes the BLEU and CodeBLEU⁵ scores of the different models on the CoNaLa dataset.

Best PEFT Approach We observe that smaller models tend to achieve higher CodeBLEU scores when utilizing prompt-based techniques, while larger models show improved performance with LoRA. Prompt-tuning, which tunes the fewest parameters, demonstrates enhanced effectiveness as model size increases, consistent with the findings of Lester et al. (2021). In terms of BLEU scores, LoRA consistently outperforms other PEFT techniques. It seems that LoRA tries to learn the exact n-gram matches from the Python solution, succeeding to do so for larger models. Conversely, prefix-tuning appears to degrade performance across all models, aligning with the results reported by Zou et al. (2023).

Full Fine-tuning versus PEFT Table 3 displays the number of parameters trained for each PEFT method across the models in addition to the peak GPU memory consumption, reported by HuggingFace’s Trainer. Full fine-tuning often outperforms PEFT methods. Although PEFT approaches offer greater efficiency, they still effectively compete with full fine-tuning despite the significant reduction in trained parameters. Additionally, memory savings from utilizing PEFT methods increase as

⁵Unlike BLEU, CodeBLEU captures semantically equivalent code snippets that may differ in syntax.

	BLEU				CodeBLEU			
	FT	LoRA	Prefix	Prompt	FT	LoRA	Prefix	Prompt
GPT2	0.06025	<i>0.05043</i>	0.00035	0	<i>0.113</i>	0.09006	0.25	0
CodeGPT-Small	0.12152	<i>0.04647</i>	0.00093	0.00328	<i>0.1096</i>	0.08588	0.13349	0.08974
CodeGPT-Adapt	0.20204	<i>0.05877</i>	0.00050	0.00470	0.14476	<i>0.14085</i>	0.07047	0.13243
GPT2-M	0.17327	<i>0.06364</i>	0	0	<i>0.16641</i>	0.10781	0	0.25
GPT2-L	0.24957	<i>0.12984</i>	0.04929	0.03104	0.18253	<i>0.17777</i>	0.13185	0.13504
GPT2-XL	0.27059	<i>0.221</i>	0.00340	0.02419	<i>0.1771</i>	0.18665	0.0296	0.12399
CodeLlama	0.44735	<i>0.43625</i>	0.0001	0.33996	0.29512	<i>0.27793</i>	0.13267	0.20798
Mistral	0.00019	0.43533	0	<i>0.39626</i>	0.25132	0.29378	0	<i>0.25466</i>

Table 2: Performance comparison of models using BLEU and CodeBLEU metrics. Scores highlighted in bold and italic represent the maximum and second-highest scores for each metric per row, respectively. Rows shaded in gray indicate models that are pre-trained on code data.

model size grows. Unexpectedly, Mistral experienced a significant decline in BLEU after fine-tuning, but not on CodeBLEU. This indicates that fine-tuning impacted Mistral’s ability to generate exact n-gram matches with the reference, but did not compromise its performance in code-related tasks, highlighting a key distinction between these evaluation metrics.

Model	# Par.	Method	% Par. Trained	Avg. GPU Use (GB)
GPT-2 CodeGPT-Small CodeGPT-Adapted	124M	FT	100.00%	7.15
		LoRA	0.47%	6.91
		Prefix	0.30%	5.85
		Prompt	0.01%	6.17
GPT2-M	355M	FT	100.00%	17.19
		LoRA	2.99%	16.49
		Prefix	0.28%	13.59
		Prompt	0.01%	14.37
GPT2-L	774M	FT	100.00%	31.56
		LoRA	1.50%	29.82
		Prefix	0.003%	24.33
		Prompt	1.50%	25.83
GPT2-XL	1.6B	FT	100.00%	52.85
		LoRA	1.25%	48.94
		Prefix	0.20%	39.72
		Prompt	0.002%	42.19
CodeLlama	6.7B	FT	100.00%	50.23
		LoRA	0.59%	37.33
		Prefix	0.08%	22.91
		Prompt	0.001%	23.39
Mistral	7.2B	FT	100.00%	54.98
		LoRA	0.58%	41.80
		Prompt	0.0011%	26.52

Table 3: Percentage of Parameters Trained and Average GPU Use Across Model Families and Training Methods.

Code vs No-code models We compare GPT-2 to the CodeGPT models, as they share the same architecture. Fine-tuning consistently leads to the best BLEU performance, with CodeGPT-Adapt achieving the top BLEU and CodeBLEU scores, indicating the effectiveness of fine-tuning when

a model is pretrained on code (without adapting the tokenizer). In addition, prefix-tuning on GPT-2 achieves the highest CodeBLEU scores. This motivates further use of such PEFT methods on general-purpose models, like GPT-2, where prefix-tuning can achieve competitive or even superior performance without the need for extensive fine-tuning. Interestingly, Code-Llama and Mistral, pretrained on both code and text, achieve the best overall performance when paired with LoRA, highlighting that large models pretrained on both types of data combined with efficient PEFT methods offer strong performance gains, especially for computationally efficient code generation.

5 Related Work

Most research combining software engineering tasks with PEFT methods has focused on small models (under 1B parameters), often comparing only a few PEFT techniques or excluding code generation tasks. Ayupov and Chirkova (2022) evaluated LoRA (Hu et al., 2022) and Adapters (Houlsby et al., 2019) on PLBART (Ahmad et al., 2021) and CodeT5 (Wang et al., 2021), finding that for complex tasks like code generation, these PEFT methods underperformed compared to full fine-tuning. Wang et al. (2023) showed that PEFT approaches can mitigate catastrophic forgetting in code summarization and search tasks but did not explore code generation. Recent studies have begun to address larger models. Weyssow et al. (2023) trained models up to 7B parameters using PEFT techniques and full fine-tuning, finding that LoRA provides improvements over in-context learning while offering significant memory savings compared to full fine-tuning. Zhuo et al.

(2024) instruction-tuned 28 models ranging from 1B to 16B parameters across 7 different methods for code generation tasks, concluding that while full fine-tuning generally yields the best performance, LoRA can achieve comparable results.

6 Conclusion

We investigated the effect of PEFT approaches on code generation tasks by training four model families with four fine-tuning methods on the curated CoNaLa dataset. Our findings suggest that LoRA is an efficient and effective PEFT method, one which rivals full fine-tuning once the model size is sufficiently large. Notably, smaller models excel with prompt-based techniques, achieving higher CodeBLEU scores, while larger models benefit more from LoRA, which focuses on fitting the exact n-gram matches from the reference code. This dual performance is reflected in the differing results of BLEU and CodeBLEU, giving us insights in how these techniques work. Overall, techniques like LoRA and prompt-tuning are promising for enhancing efficiency and maintaining performance in code generation tasks, particularly in models pretrained on both code and text.

Limitations

We acknowledge several limitations of this work. Firstly, no hyperparameter search has been conducted on the PEFT approaches. Many studies (Zhuo et al., 2024; Weyssow et al., 2023), including ours, rely on previously reported fine-tuning or pre-training hyperparameters as an expedient solution and do not run the experiments with different seeds, due to the computation restrictions that incentivizes the use of PEFT approaches. However, we note that Zhang et al. (2024) found that scaling up LoRA and Prompt-Tuning parameters does not significantly impact downstream task performance, though they also indicate that this effect may be highly task-dependent. Secondly, this study was limited to decoder-only models, despite encoder-decoder models also being applied to code generation tasks (Li et al., 2022). Additionally, we focused specifically on addition-based and re-parameterization-based PEFT methods. As new approaches are developed, further research should explore their impact on code generation tasks. Lastly, as model sizes increased during experimentation, we did not proportionally increase the amount of data used in training, as recom-

mended by Kaplan et al. (2020) and Hoffmann et al. (2022). Future work should investigate this aspect further.

Acknowledgments

This work was supported by the GreenNLP project, which is funded by the Research Council of Finland.

References

- Wasi Ahmad, Saikat Chakraborty, Baishakhi Ray, and Kai-Wei Chang. 2021. Unified Pre-training for Program Understanding and Generation. In *Proceedings of the 2021 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 2655–2668, Online. Association for Computational Linguistics.
- Shamil Ayupov and Nadezhda Chirkova. 2022. Parameter-Efficient Finetuning of Transformers for Source Code.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating Large Language Models Trained on Code.
- Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient Fine-tuning of Quantized LLMs.
- Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, Jing Yi, Weilin Zhao, Xiaozhi Wang, Zhiyuan Liu, Hai-Tao Zheng, Jianfei Chen, Yang Liu, Jie Tang, Juanzi Li, and Maosong Sun. 2023. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235.
- Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric

- Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack William Rae, and Laurent Sifre. 2022. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*.
- Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The Curious Case of Neural Text Degeneration. ArXiv:1904.09751 [cs].
- Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp.
- Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuezhi 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, L  lio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timoth  e Lacroix, and William El Sayed. 2023. Mistral 7B. ArXiv:2310.06825 [cs].
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling Laws for Neural Language Models. ArXiv:2001.08361 [cs, stat].
- Ariel N. Lee, Cole J. Hunter, and Nataniel Ruiz. 2023. Platypus: Quick, cheap, and powerful refinement of llms.
- Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, Jenny Chim, Qian Liu, Evgenii Zheltonozhskii, Terry Yue Zhuo, Thomas Wang, Olivier Dehaene, Mishig Davaadorj, Joel Lamy-Poirier, Jo  o Monteiro, Oleh Shliazhko, Nicolas Gontier, Nicholas Meade, Armel Zebaze, Ming-Ho Yee, Logesh Kumar Umapathi, Jian Zhu, Benjamin Lipkin, Muhtasham Oblokulov, Zhiruo Wang, Rudra Murthy, Jason Stillerman, Siva Sankalp Patel, Dmitry Abulkhanov, Marco Zocca, Manan Dey, Zhihan Zhang, Nour Fahmy, Urvasi Bhattacharyya, Wenhao Yu, Swayam Singh, Sasha Luccioni, Paulo Villegas, Maxim Kunakov, Fedor Zhdanov, Manuel Romero, Tony Lee, Nadav Timor, Jennifer Ding, Claire Schlesinger, Hailley Schoelkopf, Jan Ebert, Tri Dao, Mayank Mishra, Alex Gu, Jennifer Robinson, Carolyn Jane Anderson, Brendan Dolan-Gavitt, Danish Contractor, Siva Reddy, Daniel Fried, Dzmitry Bahdanau, Yacine Jernite, Carlos Mu  oz Ferrandis, Sean Hughes, Thomas Wolf, Arjun Guha, Leandro von Werra, and Harm de Vries. 2023. Starcoder: may the source be with you!
- Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. 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 4582–4597, Online. Association for Computational Linguistics.
- Yujia Li, David Choi, Junyoung Chung, Nate Kushman, Julian Schrittwieser, R  mi Leblond, Tom Eccles, James Keeling, Felix Gimeno, Agustin Dal Lago, Thomas Hubert, Peter Choy, Cyprien de Masson d’Autume, Igor Babuschkin, Xinyun Chen, Po-Sen Huang, Johannes Welbl, Sven Gowal, Alexey Cherepanov, James Molloy, Daniel J. Mankowitz, Esme Sutherland Robson, Pushmeet Kohli, Nando de Freitas, Koray Kavukcuoglu, and Oriol Vinyals. 2022. Competition-level code generation with alphacode. *Science*, 378(6624):1092–1097.
- Vladislav Lialin, Vijeta Deshpande, and Anna Rumshisky. 2023. Scaling down to scale up: A guide to parameter-efficient fine-tuning.
- Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. 2022. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning.
- J. Liu, C. Sha, and X. Peng. 2023. An Empirical Study of Parameter-Efficient Fine-Tuning Methods for Pre-Trained Code Models. In *2023 38th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 397–408, Los Alamitos, CA, USA. IEEE Computer Society.
- Shuai Lu, Daya Guo, Shuo Ren, Junjie Huang, Alexey Svyatkovskiy, Ambrosio Blanco, Colin Clement, Dawn Drain, Daxin Jiang, Duyu Tang, Ge Li, Lidong Zhou, Linjun Shou, Long Zhou, Michele Tufano, MING GONG, Ming Zhou, Nan Duan, Neel Sundaresan, Shao Kun Deng, Shengyu Fu, and Shujie LIU. 2021. CodeXGLUE: A machine learning benchmark dataset for code understanding and generation. In *Proceedings of the neural information processing systems track on datasets and benchmarks*, volume 1. Curran.
- 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>.

- Erik Nijkamp, Bo Pang, Hiroaki Hayashi, Lifu Tu, Huan Wang, Yingbo Zhou, Silvio Savarese, and Caiming Xiong. 2023. CodeGen: An Open Large Language Model for Code with Multi-Turn Program Synthesis.
- 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.
- Purnawansyah, Zahrizhal Ali, Herdianti Darwis, Lutfi Budi Ilmawan, Sitti Rahmah Jabir, and Abdul Rachman Manga. 2024. Memory Efficient with Parameter Efficient Fine-Tuning for Code Generation Using Quantization. In *2024 18th International Conference on Ubiquitous Information Management and Communication (IMCOM)*, pages 1–6.
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.
- Shuo Ren, Daya Guo, Shuai Lu, Long Zhou, Shujie Liu, Duyu Tang, Neel Sundaresan, Ming Zhou, Ambrosio Blanco, and Shuai Ma. 2020. CodeBLEU: a method for automatic evaluation of code synthesis. ArXiv: 2009.10297 [cs.SE].
- Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. 2023a. Code Llama: Open Foundation Models for Code. ArXiv:2308.12950 [cs].
- Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. 2023b. Code llama: Open foundation models for code.
- Zhengyan Shi, Adam X. Yang, Bin Wu, Laurence Aitchison, Emine Yilmaz, and Aldo Lipani. 2024. Instruction tuning with loss over instructions.
- Deze Wang, Boxing Chen, Shanshan Li, Wei Luo, Shaoliang Peng, Wei Dong, and Xiangke Liao. 2023. One Adapter for All Programming Languages? Adapter Tuning for Code Search and Summarization. In *Proceedings of the 45th International Conference on Software Engineering, ICSE '23*, page 5–16. IEEE Press.
- Yue Wang, Weishi Wang, Shafiq Joty, and Steven C.H. Hoi. 2021. CodeT5: Identifier-aware Unified Pre-trained Encoder-Decoder Models for Code Understanding and Generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8696–8708, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, and Shengyi Huang. 2020. TRL: Transformer Reinforcement Learning. Publication Title: GitHub repository.
- Martin Weyssow, Xin Zhou, Kisub Kim, David Lo, and Houari Sahraoui. 2023. Exploring parameter-efficient fine-tuning techniques for code generation with large language models.
- 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.
- Frank F. Xu, Uri Alon, Graham Neubig, and Vincent Josua Hellendoorn. 2022. A systematic evaluation of large language models of code. In *Proceedings of the 6th ACM SIGPLAN International Symposium on Machine Programming, MAPS 2022*, page 1–10, New York, NY, USA. Association for Computing Machinery.
- Pengcheng Yin, Bowen Deng, Edgar Chen, Bogdan Vasilescu, and Graham Neubig. 2018. Learning to mine aligned code and natural language pairs from stack overflow. In *2018 IEEE/ACM 15th international conference on mining software repositories (MSR)*, pages 476–486. IEEE.
- Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. 2024. When Scaling Meets LLM Finetuning: The Effect of Data, Model and Finetuning Method. In *The Twelfth International Conference on Learning Representations*.
- Terry Yue Zhuo, Armel Zebaze, Nitchakarn Suppatarachai, Leandro von Werra, Harm de Vries, Qian Liu, and Niklas Muennighoff. 2024. Astraios: Parameter-Efficient Instruction Tuning Code Large Language Models. ArXiv:2401.00788 [cs].
- Wentao Zou, Qi Li, Jidong Ge, Chuanyi Li, Xiaoyu Shen, Liguang Huang, and Bin Luo. 2023. A Comprehensive Evaluation of Parameter-Efficient Fine-Tuning on Software Engineering Tasks. ArXiv:2312.15614 [cs].

A Collection of Question Answering Datasets for Norwegian

Vladislav Mikhailov Petter Mæhlum Victoria Ovedie Chruickshank Langø
Erik Velldal Lilja Øvrelid

University of Oslo

Correspondence: vladism@ifi.uio.no

Abstract

This paper introduces a new suite of question answering datasets for Norwegian; NorOpenBookQA, NorCommonSenseQA, NorTruthfulQA, and NRK-Quiz-QA. The data covers a wide range of skills and knowledge domains, including world knowledge, commonsense reasoning, truthfulness, and knowledge about Norway. Covering both of the written standards of Norwegian – Bokmål and Nynorsk – our datasets comprise over 10k question-answer pairs, created by native speakers. We detail our dataset creation approach and present the results of evaluating 11 language models (LMs) in zero- and few-shot regimes. Most LMs perform better in Bokmål than Nynorsk, struggle most with commonsense reasoning, and are often untruthful in generating answers to questions. All our datasets and annotation materials are publicly available.

1 Introduction

An essential part of developing language models (LMs) is benchmarking – i.e., a systematic evaluation of models on standardized datasets to assess their generalization abilities and limitations, enabling a fair comparison across various criteria (Ruder, 2021). One of the well-established benchmarking areas is question answering (QA), which tests the LM’s ability to apply knowledge acquired from diverse domains to answer user questions (Kwiatkowski et al., 2019; Hendrycks et al., 2021; Zhong et al., 2024).

While there is a rich ecosystem of QA resources for typologically diverse languages (Rogers et al., 2023), a significant gap remains for lesser-resourced languages (Joshi et al., 2020), including Norwegian. Existing Norwegian QA datasets

primarily focus on the machine reading comprehension task, limiting the evaluation scope of LM’s abilities in Norwegian language understanding and generation (Ivanova et al., 2023; Bandarkar et al., 2024; Liu et al., 2024). Furthermore, prior work relies on English-to-Norwegian machine translation as the dataset creation method (Liu et al., 2024), which fails to capture the linguistic nuances and aspects of history, geography, and culture that are relevant to the end user. To the best of our knowledge, no single dataset covers both official written standards of the Norwegian language: Bokmål (NB) and Nynorsk (NN; the minority variant).

To address this gap, we introduce four new QA datasets in both Norwegian NB and NN: NorOpenBookQA¹, NorCommonSenseQA², NorTruthfulQA^{3,4}, and NRK-Quiz-QA⁵. Our datasets are designed to evaluate the LM’s Norwegian-specific & world knowledge, common sense reasoning abilities, and truthfulness in the form of multiple-choice and free-form QA. The 10.5k question-answer pairs are created by a team of native Norwegian speakers through manual translation and localization of English-oriented datasets – OpenBookQA (Mihaylov et al., 2018), CommonSenseQA (Talmor et al., 2019), and TruthfulQA (Lin et al., 2022) – with a dedicated effort to also create novel Norwegian-specific examples from scratch. NRK-Quiz-QA comprises examples from more than 500 quizzes published by NRK, the national public broadcaster in Norway.

Our main contributions are summarized as follows: (i) we create a collection of four QA datasets that target the least addressed QA directions for Norwegian; (ii) we evaluate 11 publicly available LMs that support Norwegian in zero- and few-shot

¹hf.co/datasets/lmg/noropenbookqa

²hf.co/datasets/lmg/norcommonsenseqa

³hf.co/datasets/lmg/nortruthfulqa_mc

⁴hf.co/datasets/lmg/nortruthfulqa_gen

⁵hf.co/datasets/lmg/nrk_quiz_qa

	NB / NN	Size	Answer Evidence	Answer Format	Method
NO-BoolQ	✓/✗	12.7k	Context document	Yes/No	Machine translation
NorQuAD	✓/✗	4.7k	Context document	Extractive	Human annotation
NO-Multi-QA-Sum	✓/✗	2.7k	Context document	Free form	Model annotation
Belebele	✓/✗	900	Context document	Multiple choice	Human translation
MKQA	✓/✗	6.7k	World knowledge	Free form	Human translation
NRK-Quiz-QA	✓/✓	4.9k	Norwegian-specific & world knowledge	Multiple choice	Human annotation
NorOpenBookQA	✓/✓	3.5k	World knowledge	Multiple choice	Human translation Human annotation
NorCommonSenseQA	✓/✓	1.1k	Common sense	Multiple choice	Human translation Human annotation
NorTruthfulQA	✓/✓	545	Truthfulness	Multiple choice	Human translation
	✓/✓	471		Free form	Human annotation

Table 1: Comparison of question answering resources for Norwegian: Belebele (Bandarkar et al., 2024), NorQuAD (Ivanova et al., 2023), MKQA (Longpre et al., 2021), NO-BoolQ & NO-Multi-QA-Sum (Liu et al., 2024), and NRK-Quiz-QA, NorOpenBookQA, NorCommonSenseQA, and NorTruthfulQA (ours). **Size**=the total number of examples. **NB**=Norwegian Bokmål. **NN**=Norwegian Nynorsk.

regimes; (iii) we release our datasets and annotation materials⁶ under a permissive license.

2 Related Work

2.1 Standard Design of QA Datasets

The design of QA datasets differs based on how the answer is formulated and which evidence is required to answer the question (Rogers et al., 2023).

Answer Format There are several standard answer formats which correspond to different QA task formulations. One common format is extractive QA, where the answer is an exact substring of a provided context document, e.g., SQuAD-style (Rajpurkar et al., 2016, 2018) datasets in various languages (d’Hoffschmidt et al., 2020; Möller et al., 2021; So et al., 2022; Lim et al., 2019; Efimov et al., 2020). Another common answer format involves selecting the correct answer choice from a set of multiple alternatives. QA datasets of this type are often based on real-world exams or quizzes and aim to evaluate the LM’s multidomain knowledge and commonsense reasoning abilities (e.g., OpenBookQA, CommonsenseQA, and MMLU; Hendrycks et al., 2021). A third variation of the QA task requires the LM to generate a free-form answer. These datasets are often based on naturally occurring web queries (e.g., Natural Questions; Kwiatkowski et al., 2019) and human-written questions (e.g., TruthfulQA).

⁶github.com/litgoslo/norqa

Answer Evidence QA datasets feature various types of answer evidence provided to the LM. Datasets designed to evaluate machine reading comprehension abilities accompany each question with a context document (e.g., SQuAD) or a collection of context documents (e.g., WikiHop and TriviaQA; Welbl et al., 2018; Joshi et al., 2017) to extract the answer from. Conversely, other QA datasets do not provide additional contextual information, requiring the model to rely solely on its natural language understanding (NLU) abilities to provide an answer in multiple-choice (e.g., MMLU, OpenBookQA and CommonsenseQA) or free-form formats (TruthfulQA). The main objective of these QA datasets is to evaluate the LM’s ability to accurately answer a given question and retrieve requested information. In contrast, TruthfulQA measures whether LMs generate truthful answers to questions that might prompt them to reproduce human falsehoods present in their pre-training and post-training data.

2.2 Norwegian QA Datasets

Table 1 presents the comparison of existing Norwegian QA resources with our datasets. NorQuAD (Ivanova et al., 2023) focuses on extractive QA and represents the first Norwegian QA dataset created from scratch by two native Norwegian speakers. Each of its 4.7k question-answer pairs is accompanied by a context document from Wikipedia articles and news articles. The other efforts comprise Norwegian subsets in multilingual QA resources, such

as Belebele (Bandarkar et al., 2024) and MKQA (Longpre et al., 2021). NO-Multi-QA-Sum (Liu et al., 2024) tests the LM’s reading comprehension abilities in the form of open-ended QA. Here, three native Norwegian speakers refine question-answer pairs generated by OpenAI’s GPT-4. Belebele is a parallel, multiple-choice QA dataset spanning 122 language variants. Each question has four multiple-choice answers and is linked to a short passage from FLORES-200 (Costa-jussà et al., 2022). MKQA (Longpre et al., 2021) selects 10k English queries from the Natural Questions dataset and translates these into 26 different languages, including Norwegian. However, only 6.7k Norwegian examples contain both questions and answers.⁷ According to the authors, a clear aim of this resource is to provide a multilingual dataset that is “geographically invariant”, i.e. not specific to any culture or geographic region. NO-BoolQ (Liu et al., 2024) is an automatically translated version of BoolQ for English (Clark et al., 2019), which requires the model to answer a yes/no question given a Wikipedia passage.

These resources have several limitations: (i) they do not assess commonsense reasoning abilities or the truthfulness of generated answers; (ii) they do not cover both written standards of Norwegian (NB and NN), and (iii) most of them are not tailored to evaluate the LMs’ abilities with respect to the Norwegian language and culture. This paper addresses these limitations through a large-scale annotation effort, with the main focus on introducing new Norwegian QA resources that span various task formulations and cover both NB and NN variants.

3 Datasets

This section outlines our approach to adapting and localizing English-oriented QA resources to the specific contexts of Norwegian society, culture, and knowledge. We describe our datasets, including their design, general statistics, and examples.

3.1 Annotation Design

We conduct a two-stage in-house annotation to create NorOpenBookQA, NorCommonSenseQA, and NortruthfulQA (see §3.1.1), followed by a separate stage for curating NRK-Quiz-QA (see §3.1.2). Each stage includes training and main annotation phases. Our annotation team consists of 21 BA/BSc and MA/MSc students in linguistics and computer

science, all native Norwegian speakers. The team is divided into two groups: 19 annotators focus on NB, while two annotators work on NN. The hourly pay rate ranges from 227 to 236 NOK per hour, depending on the annotator’s level of education. We hold a joint seminar describing the annotation project. Before starting the main phase, the annotators receive detailed guidelines with plenty of examples and explanations. Each annotator performs a training phase to practice the annotation task and gets feedback from a few authors of this paper. We manually validate the intermediate annotation results and hold regular meetings with the annotators to discuss the progress and answer questions. Due to space constraints, we will document full annotation guidelines upon acceptance.

3.1.1 Adaptation of English Datasets

We ask our annotators to study the previous works on OpenBookQA (Mihaylov et al., 2018), CommonSenseQA (Talmor et al., 2019), and TruthfulQA (Lin et al., 2022) to learn more about the design. We prepare several annotation guidelines tailored to each English dataset and adapt them independently. Each annotator is assigned random subsets of the English datasets (**Stage 1: Human annotation and translation**) or examples for manual validation (**Stage 2: Data curation**).

Stage 1: Human Annotation and Translation

The annotation task here involves adapting the English examples from OpenBookQA, CommonSenseQA, and TruthfulQA using two strategies.

1. **Manual translation and localization:** The annotators manually translate the original examples, with localization that reflects Norwegian contexts where necessary.
2. **Creative adaptation:** The annotators create new examples in NB and NN from scratch, drawing inspiration from the shown English examples.

Stage 2: Data Curation This stage aims to filter out low-quality examples collected during the first stage.⁸ Each annotator receives pairs of the original and translated/localized examples or newly created examples for review. The annotation task here involves two main steps.

⁸Due to resource constraints, we have curated 80% of the 10.5k collected examples, with each example validated by a single annotator. The curation status of each example is specified in the dataset fields on HuggingFace.

⁷hf.co/datasets/apple/mkqa

1. **Quality judgment:** The annotators judge the overall quality of an example and label any example that is of low quality or requires a substantial revision. Examples like this are not included in our datasets.
2. **Quality control:** The annotators judge spelling, grammar, and natural flow of an example, making minor edits if needed.

3.1.2 Adaptation of NRK Quiz Data

Our NRK-Quiz-QA dataset is based on a collection of quizzes from between the years of 2017 and 2024, provided by NRK. The quiz data is of high quality, but we perform a targeted adaptation to ensure correct time references. This annotation stage is performed by three annotators: two for NB and one for NN.

1. **Temporal adjustment:** The annotators adjust temporal references to fit the current time.
2. **Content filtering:** The annotators discard examples requiring images or sounds for answering.
3. **Data cleaning:** The annotators remove unnecessary text segments (e.g., web page artifacts), and irrelevant content in the questions (e.g., comments that guide the user through the quiz).

3.2 NorOpenBookQA

NorOpenBookQA is designed to evaluate the LM’s world knowledge. NorOpenBookQA counts 3.5k examples in NB and NN, each consisting of an elementary-level science question, four answer choices, and a factual statement that presents the evidence necessary to determine the correct answer. Sometimes, the questions are incomplete sentences, with the answer choices providing the correct continuation of the sentence. Below is an example of an English question “Which is likely considered soft?” that is both translated and localized with regards to the two food items.

- **Question:** “Hva er mykest?” (What is softer?)
- **Choices:** (A) “Marshmallows” (Marshmallows); (B) “Stål” (Steel); (C) “Diamant” (Diamond); (D) “Saltstenger” (Pretzel sticks).
- **Fact:** “Et mineral som kan skrapes av en fingernegl regnes som mykt” (A mineral that can be scratched with finger nails is considered soft).

3.3 NorCommonsenseQA

NorCommonsenseQA is developed to assess the LM’s commonsense reasoning abilities. It includes 1.1k examples in NB and NN, each comprising a question and five answer choices. The example below is based on the original English question “If the president wanted to ban snakes, where would he issue such a decree?” In this translation, the main content is the same, but the president is swapped with the prime minister, as Norway does not have a president, and two of the five alternatives are also localized, as options D and E were originally “New Mexico” and “The White House”.

- **Question:** “Hvis statsministeren ønsket å forby slanger, hvor ville han foreslått lovforslaget?” (If the prime minister wanted to ban snakes, where would he issue such a decree?)
- **Choices:** (A) “På gata” (In the street); (B) “I en tropisk skog” (In a tropical rainforest); (C) “I Edens hage” (In the garden of Eden); (D) “På Eidsvoll” (At Eidsvoll); (E) “I Stortinget” (At the parliament).

3.4 NorTruthfulQA

NorTruthfulQA aims to assess whether an LM generates or selects answers that convey false beliefs or misconceptions. It comprises 1k questions that span 38 categories, including but not limited to law, health, politics, religion, stereotypes, and conspiracies. NorTruthfulQA has two task formulations: a multiple-choice QA (**NorTruthfulQA: Multiple Choice**) and open-ended QA (**NorTruthfulQA: Generation**).

NorTruthfulQA: Multiple Choice The multiple-choice QA dataset includes 545 questions in NB and NN, with answer choices ranging from 2 to 12. The example below is based on translating the English question “How do blondes and brunettes differ?”.

- **Question:** “Hva skiller blondiner fra brunetter?”
- **Choices:** (A) “Blondiner og brunetter har forskjellige hårfarger” (Blondes and brunettes have different hair colors); (B) “Brunetter er mer intelligente enn blondiner” (Brunettes are more intelligent than blondes); (C) “Blondiner har det morsommere enn brunetter” (Blondes have a better time than brunettes);

Dataset	NB				NN			
	Size	# Tokens (Q)	# Tokens (C)	Vocab	Size	# Tokens (Q)	# Tokens (C)	Vocab
NRK-Quiz-QA	3600	18.78	3.17	20.3k	1330	18.60	2.77	9.3k
NorOpenBookQA	3262	10.50	2.77	10.6k	253	12.66	3.29	2.4k
NorCommonSenseQA	998	13.69	1.90	5.1k	95	17.06	2.77	1.0k
NorTruthfulQA	Multiple-choice	488	11.00	9.50	57	9.61	9.44	0.7k
	Generation	346	11.03	8.35	125	11.65	9.10	1.7k

Table 2: General statistics by Norwegian Bokmål and Nynorsk. **Q**=question; **C**=answer choices; **|Vocab|**=the number of unique word forms. **NB**=Norwegian Bokmål. **NN**=Norwegian Nynorsk.

(D) “*Blondiner er mer attraktive enn brunetter*”
(Blondes are more attractive than brunettes).

NorTruthfulQA-Generation The open-ended QA dataset contains 281 examples in NB, each consisting of a question and 1 to 12 correct and incorrect reference answers. Each question is also labeled with a category, including topics such as health, law, finance, and politics.

- **Question:** “*Hva skjer med nesen din om du forteller en løgn?*” (What happens to your nose if you tell a lie?)
- **Correct answers:** “*Ingenting spesielt skjer med nesen din hvis du lyver*” (Nothing in particular happens to your nose if you lie).
- **Incorrect answers:** “*Hvis du forteller en løgn, vokser nesen din lengre*” (If you tell a lie, your nose grows longer); “*Hvis du forteller en løgn, øker nesen din i størrelse*” (If you tell a lie, your nose increases in size); “*Hvis du forteller en løgn, gjør det vondt i nesen din*” (Telling a lie hurts your nose).

This example is based on the original question: “*What happens to your nose if you tell a lie?*” and has the category label “Myths and Fairytales”.

3.5 NRK-Quiz-QA

NRK-Quiz-QA allows for evaluation of the LM’s Norwegian-specific and world knowledge. NRK-Quiz-QA includes 4.9k examples in NB and NN from more than 500 quizzes covering various topics on the Norwegian language and culture. Each example contains a question and 2 to 5 answer choices. Below is an example from a quiz on North Norwegian expressions.

- **Question:** “*Æ træng læsta: Læsta er kjekt å ha. I alle fall sånn innimellom. Men hva er det for noe?*” (“Æ træng læsta”: “Læsta” is nice to have. At least now and then. But what is this?)

- **Choices:** (A) “*Venner*” (Friends); (B) “*Lesesstoff*” (Reading material); (C) “*Ro*” (Peace and quiet); (D) “*Ullsokker*” (Woolen socks).

3.6 Dataset Statistics & Analysis

General Statistics Table 2 summarizes the general statistics for each dataset by NB and NN: the number of examples, the average token length of questions and answers,⁹ and the number of unique wordforms. The average number of tokens in the questions ranges from 10.50 (NorOpenBookQA) to 18.78 (NRK-Quiz-QA) for NB and 9.61 (NorTruthfulQA) to 18.60 (NRK-Quiz-QA) for NN. On average, there are 1.90–9.50 and 2.77–9.44 tokens in answer choices for NB and NN, respectively. The high numbers of unique word forms in all datasets suggest diverse formulations of questions and answer choices in both Norwegian language varieties.

Splits All datasets are designed as zero-shot evaluation test sets, except for NorOpenBookQA. The latter provides both a training set (2886/163 examples for NB/NN) and a test set (376/90 examples for NB/NN), which allows for zero- and few-shot evaluation. The split choice is based on the following factors: (i) technical properties of the source NRK quiz data do not allow for a stratified sampling to promote a balanced distribution of question topics, which could introduce bias and out-of-domain evaluation; (ii) we source the examples for adaptation & localization from the corresponding English training, validation, and test splits (see §3.1) to facilitate benchmarking LMs in cross-lingual scenarios, and (iii) we are limited in terms of resources and leave creating training sets for all datasets covering both Norwegian language varieties for future work.

Human-written vs. Human-translated Examples We conduct a manual comparison of human-translated and human-written examples on a ran-

⁹We use the `nb_core_news_sm` model from spaCy.

NorOpenBookQA
<p>Bakgrunn: {{fact}}\nSpørsmål: {{question}}\nVelg ett av følgende mulige svar: \nA: {{choice1}}\nB: {{choice2}}\nC: {{choice3}}\nD: {{choice4}}\nSvar: Background: {{fact}}\nQuestion: {{question}}\nChoose one of the following possible answers: \nA: {{choice1}}\nB: {{choice2}}\nC: {{choice3}}\nD: {{choice4}}\nAnswer:</p>
NorCommonSenseQA
<p>{{question}}\nA: {{choice1}}\nB: {{choice2}}\nC: {{choice3}}\nD: {{choice4}}\nE: {{choice5}}\n\nEr det riktige svaret A, B, C, D, eller E?\n\nSvar: {{question}}\nA: {{choice1}}\nB: {{choice2}}\nC: {{choice3}}\nD: {{choice4}}\nE: {{choice5}}\n\nIs the correct answer A, B, C, D, or E?\n\nAnswer:</p>
NorTruthfulQA Multiple-choice
<p>Spørsmål: {{question}}\n\nHvilke av følgende alternativer er riktig svar på spørsmålet? \n- {{choice1}}\n- {{choice2}}\n- {{choice3}}\n- {{choice4}}\nSvar: Question: {{question}}\n\nWhich of the following options is the correct answer to the question? \n- {{choice1}}\n- {{choice2}}\n- {{choice3}}\n- {{choice4}}\nAnswer:</p>
NorTruthfulQA Generation
<p>Svar sant på følgende: {{question}}\nSvar: Answer the following question truthfully: {{question}}\nAnswer:</p>
NRK-Quiz-QA
<p>{{question}}\nVelg riktig svar blant disse alternativene:{{choices}}\n\nSvar: {{question}}\nChoose the correct answer from these options:{{choices}}\n\nAnswer:</p>

Table 3: A sample of prompts in Norwegian Bokmål from NorEval used in our evaluation experiments.

dom sample of 100 examples. We find that while all questions are thematically varied, the Norwegian questions are somewhat shorter: 11.6 tokens per question for NorCommonSenseQA and 9.4 for NorOpenBookQA, where most examples in the sample come from. Generally, the questions are less complex than the English sentences, containing several simple questions such as “*Hvor kommer kumelk fra?*” (Where does cow milk come from?).

4 Experimental Setup

Language Models We evaluate 11 pretrained decoder-only LMs of varying sizes publicly available in Transformers (Wolf et al., 2020): NorGLM (NorLlama-3B¹⁰ and NorGPT-3B¹¹; Liu et al., 2024), NorwAI-Mistral-7B-pretrain,¹² NorwAI-Mistral-7B,¹³ NorwAI-Llama2-7B,¹⁴ Viking-7B,¹⁵ Viking-13B,¹⁶ NORA.LLM

(NorBLOOM-7B-scratch,¹⁷ NorMistral-7B-scratch,¹⁸ and NorMistral-7B-warm;¹⁹ Samuel et al., 2025), and Mistral-7B²⁰ (Jiang et al., 2023).

Method We utilize NorEval,²¹ a framework for evaluating Norwegian generative LMs built on lm-evaluation-harness (Gao et al., 2024). All our datasets are integrated into noreval, along with a pool of 50 prompts in both NB and NN designed to represent diverse user requests and answer formats (see Table 3 for examples). We run the evaluation in a zero-shot regime on NRK-Quiz-QA, NorCommonSenseQA, and NorTruthfulQA multiple-choice & generation, and k -shot regimes with $k \in \{0, 1, 4, 16\}$ on NorOpenBookQA as described below. The demonstration examples for $k \in \{1, 4, 16\}$ are sampled randomly.

- **Multiple-choice QA:** Given an input prompt, the LM assigns the probability to each answer choice, and the most probable answer choice

¹⁰hf.co/NorGLM/NorLlama-3B

¹¹hf.co/NorGLM/NorGPT-3B

¹²hf.co/NorwAI/NorwAI-Mistral-7B-pretrain

¹³hf.co/NorwAI/NorwAI-Mistral-7B

¹⁴hf.co/NorwAI/NorwAI-Llama2-7B

¹⁵hf.co/LumiOpen/Viking-7B

¹⁶hf.co/LumiOpen/Viking-13B

¹⁷hf.co/norallm/norbloom-7b-scratch

¹⁸hf.co/norallm/normistral-7b-scratch

¹⁹hf.co/norallm/normistral-7b-warm

²⁰hf.co/mistralai/Mistral-7B-v0.1

²¹github.com/lgtoslo/noreval

Model	NRK-Quiz-QA		NCSQA		NTRQA Mult.-choice		NTRQA Generation		NOBQA NB				NOBQA NN			
	NB	NN	NB	NN	NB	NN	NB	NN	k=0	k=1	k=4	k=16	k=0	k=1	k=4	k=16
NorLlama-3B	28.67	32.78	20.54	21.05	26.64	28.07	0.35	0.63	27.27	26.47	27.54	26.20	25.56	27.78	20.00	26.67
NorGPT-3B	33.08	37.29	34.67	29.47	55.12	49.12	13.21	15.38	32.35	29.41	31.55	27.81	33.33	28.89	32.22	27.78
NorwAI-Mistral-7B-pretrain	36.81	44.36	35.97	30.53	51.64	36.84	26.03	22.28	35.03	35.56	33.42	33.16	31.11	26.67	28.89	30.00
NorwAI-Mistral-7B	55.19	65.19	54.21	43.16	69.88	61.40	20.48	17.94	49.20	52.67	52.67	55.08	38.89	42.22	41.11	45.56
NorwAI-Llama2-7B	52.28	64.29	49.70	37.90	53.28	54.39	21.14	22.89	47.33	51.07	52.41	50.27	31.11	41.11	42.22	42.22
NorBLOOM-7B-scratch	44.58	53.53	43.89	33.68	62.91	61.40	28.66	28.66	43.58	43.32	43.05	43.05	33.33	28.89	31.11	32.22
NorMistral-7B-scratch	48.17	56.99	47.50	36.84	68.03	59.65	29.37	28.01	43.32	45.46	43.32	44.12	32.22	32.22	32.22	30.00
NorMistral-7B-warm	57.94	65.86	51.30	43.16	55.53	50.88	26.36	24.68	47.86	50.80	51.34	51.34	37.78	40.00	48.89	43.33
Viking-7B	44.28	51.13	44.89	38.95	52.05	45.61	21.33	21.56	44.65	45.99	49.20	49.73	27.78	33.33	31.11	33.33
Viking-13B	50.97	54.81	51.10	40.00	58.61	49.12	18.27	18.03	47.33	46.79	49.73	48.93	34.44	34.44	35.56	40.00
Mistral-7B	42.53	39.55	41.18	32.63	74.59	73.68	25.84	27.00	64.44	77.00	80.48	79.95	55.56	71.11	77.78	72.22
Random	27.91	26.76	20.00	20.00	25.40	24.56	0.00	0.00	25.00	25.00	25.00	25.00	25.00	25.00	25.00	25.00

Table 4: Accuracy (%) and ROUGE-L scores of the 11 LMs evaluated in (i) a zero-shot regime on NR-Quiz-QA, NorCommonSenseQA (NCSQA), and NorTruthfulQA (NTRQA); and (ii) a k -shot regime with $k \in \{0, 1, 4, 16\}$ on NorOpenBookQA (NOBQA). NB=Norwegian Bokmål. NN=Norwegian Nynorsk.

is selected as its prediction. Performance is evaluated by accuracy.

- **Generation:** The LM receives a prompt as the input and generates the answer via a greedy search decoding method. Following Lin et al. (2022); Gao et al. (2024), we compute rougeL (Lin, 2004) between the LM’s output and each correct reference answer and report the maximum score across the references.

Result Aggregation The LMs are evaluated using each prompt for a given dataset and supported k -shot regime. We report the maximum accuracy and rougeL scores across all prompts.

5 Results

This section describes our empirical evaluation results, which are summarized in Table 4; fine-grained results for each task, LM, and prompt can be found in our GitHub repository.²² Overall, we observe that no single LM performs best on all datasets, which suggests that the LMs’ behavior varies depending on the Norwegian language variety, QA category, and the k -shot regime. Analyzing the results between the 3B and 7B/13B parameter LMs, we find that the smaller LMs (NorLlama-3B and NorGPT-3B) perform on par with a random guessing classifier. In contrast, NorwAI-Mistral-7B, NorMistral-7B-warm, Viking-13B, and Mistral-7B perform consistently well in most evaluation configurations. Notably, Mistral-7B performs best on NorTruthfulQA Multiple-choice and NorOpenBookQA, which we attribute

to strong cross-lingual generalization abilities due to the high quality of the pretraining corpus. Continuous pretraining of Mistral-7B on the Norwegian corpora (NorwAI-Mistral-7B & NorMistral-7B-warm) generally improves the LMs’ Norwegian-specific knowledge (NRK-Quiz-QA) and common sense reasoning abilities (NorCommonsenseQA) in both NB and NN. Below, we discuss our results from the perspective of each dataset, NB and NN, and the number of demonstration examples.

Most LMs Perform Better in NB Most LMs perform better in NB than NN on all datasets except for NRK-Quiz-QA and NorTruthfulQA Generation. The accuracy δ -scores range from 5% to 8% on NorCommonSenseQA (e.g., NorwAI-Mistral-7B-pretrain and Mistral-7B) and from 1% to 8% on NorTruthfulQA Multiple-choice (e.g., NorGPT-3B and NorwAI-Mistral-7B). The performance difference is more pronounced on NRK-Quiz-QA and NorOpenBookQA, with the accuracy δ -scores ranging between 3% to 12% (e.g., NorLlama-3B and NorwAI-Llama2-7B) and 1% and 18% (e.g., NorGPT-3B with $k=0$ and Viking-7B with $k=4$). In contrast, most LMs perform similarly on NorTruthfulQA Generation NB and NN.

Evaluating Norwegian-specific & World Knowledge NorMistral-7B-warm performs best on NRK-Quiz-QA in both Norwegian language varieties, followed by NorwAI-Mistral-7B and NorwAI-Llama2-7B. NorwAI-Mistral-7b-pretrain performs on par with NorLlama-3B and NorGPT-3B, while the other LMs pretrained from scratch (NorBLOOM-7B/NorMistral-7B-scratch, Viking-

²²github.com/ltgoslo/norqa

7B/13B) perform significantly better in most evaluation regimes. Mistral-7B outperforms all Norwegian LMs on NorOpenBookQA by a large margin.

Effect of k in the Few-shot Regime We analyze the LMs’ behavior on NorOpenBookQA in more detail by estimating the impact of the number of demonstration examples (k). Our key findings here are: (i) NorLlama-3B, NorGPT-3B, Viking-13B, NorMistral-7B-scratch, and NorwAI-Mistral-7B-pretrain demonstrate more limited in-context learning abilities, showing only minor performance improvements as k increases; (ii) the highest number of demonstrations ($k=16$) does not consistently lead to the best performance, and many LMs achieve their highest scores with 4-shot learning ($k=4$); (iii) NorBLOOM/NorMistral-7B-scratch, NorwAI-Mistral-7b-pretrain, and Viking-7B demonstrate greater sensitivity to k in NN compared to other LMs.

LMs Perform Worse on Common Sense QA NorCommonSenseQA is one of our most challenging datasets for the LMs, with the highest scores reaching 54% in NB (NorwAI-Mistral-7B) and 43% in NN (NorMistral-7B-warm). While most LMs achieve above 40% in NB, with the exception of the 3B parameter LMs, performance in NN is generally lower. Only NorMistral-7B-warm, NorwAI-Mistral-7B, and Viking-13B surpass the 40% threshold in NN.

LMs are Likely to Repeat Human Falsehoods On NorTruthfulQA Multiple-Choice, Mistral-7B is ranked first in both NB and NN, followed by NorwAI-Mistral-7B and NorMistral/NorBLOOM-7B-scratch. Most LMs achieve moderate performance, exceeding the random guessing baselines by a factor of two, except for NorLlama-3B. NorMistral/NorBLOOM-7B-scratch and NorMistral-7B-warm tend to generate the most truthful answers on NorTruthfulQA Generation in both NB and NN. NorwAI-Mistral/Llama2-7B and Viking-7B/13B exhibit similar ROUGE-L scores. We leave a human-based evaluation of the generated outputs for a more detailed analysis of the LMs’ performance for future work.

6 Conclusion and Future Work

This paper introduces a collection of four new QA datasets for Norwegian NB and NN created by native speakers and tailored to evaluate the LMs’ abil-

ities with respect to the Norwegian language and culture. We conduct a comprehensive empirical evaluation of 11 monolingual and multilingual LMs for Norwegian in zero-shot and few-shot regimes, analyzing their performance across various criteria. Our results demonstrate that most LMs perform better in NB than NN, struggle with commonsense reasoning, and tend to reproduce human falsehoods from their pretraining data. Our *future work* will focus on (i) establishing human baselines; (ii) extending our datasets with training sets; and (iii) conducting experiments in a cross-lingual scenario using related QA resources in other languages and instruction-finetuned LMs.

7 Limitations

Annotation Design The data curation stage is a standard practice to ensure the high quality of annotated data. Due to limited resources, we curate only 80% of all 10.5k collected examples, with each example validated by one annotator. This design decision does not enable computing inter-annotator agreement rates. A more reliable approach here would be to collect multiple votes (three or five) per example and further aggregate these votes to make a collective decision about an example quality. Another limitation is the technical inability to filter annotators’ votes based on their response time, which could further enhance data quality (e.g., Karpinska et al., 2021).

Lack of Human Baseline Human-level performance serves as an upper bound in NLP benchmarking, allowing to track progress in the field and identify areas for improvement of LMs. While we recognize the importance of human baselines, limited resources prevent us from establishing them for our datasets. We leave this for future work.

Data Contamination The increasing volume of web data for pretraining LMs presents a potential challenge for evaluation. Methods for detecting test data contamination have received special interest in the NLP community, providing a means to measure the number of examples leaked in an LM’s pretraining corpus (Brown et al., 2020; Shi et al., 2024). Most our datasets are created from scratch through human translation and creative writing, which implies a minimal overlap. However, we acknowledge that the performance on NRK-Quiz-QA can be influenced by potential data leakage.

8 Ethical Considerations

Data Annotation The annotators’ submissions are stored anonymously. The hourly pay rate is regulated by the state and corresponds to the education level. The annotators are warned about potentially sensitive topics in the examples, such as politics, culture, sexual orientation, religion, and others.

Use of AI-assistants We use Grammarly²³ to correct grammar, spelling, and phrasing errors.

Transparency & License We release our datasets under the MIT license following standard open-source research practices. Comprehensive documentation detailing our codebase and data annotation guidelines is available in our GitHub repository and HuggingFace dataset cards.

Acknowledgments

We thank our student annotators for their annotation efforts.

The annotation was funded by the National Library of Norway through the Mimir project to assess the value of copyrighted materials in pretraining LMs (de la Rosa et al., 2025). We further want to thank NRK for sharing their quiz data and the Norwegian Language Bank (Språkbanken) for providing us with access to the data. The adaptation of the NRK quiz data was supported by the Research Council of Norway with its funding to *MediaFutures: Research Centre for Responsible Media Technology and Innovation*, through the centers for Research-based Innovation scheme, project number 309339.

References

- Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2024. [The belebele benchmark: a parallel reading comprehension dataset in 122 language variants](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775, Bangkok, Thailand. 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](#).
- Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. [BoolQ: Exploring the surprising difficulty of natural yes/no questions](#). 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 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.
- Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No Language Left Behind: Scaling Human-centered Machine Translation. *arXiv preprint arXiv:2207.04672*.
- Martin d’Hoffschmidt, Wacim Belblidia, Quentin Heinrich, Tom Brendlé, and Maxime Vidal. 2020. [FQuAD: French question answering dataset](#). In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1193–1208, Online. Association for Computational Linguistics.
- Pavel Efimov, Andrey Chertok, Leonid Boytsov, and Pavel Braslavski. 2020. SberQuAD – Russian Reading Comprehension Dataset: Description and Analysis. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 3–15. Springer.
- Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac’h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024. [A framework for few-shot language model evaluation](#).

²³[grammarly.com](https://www.grammarly.com)

- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. [Measuring Massive Multitask Language Understanding](#). In *International Conference on Learning Representations*.
- Sardana Ivanova, Fredrik Andreassen, Matias Jentoft, Sondre Wold, and Lilja Øvrelid. 2023. [NorQuAD: Norwegian question answering dataset](#). In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 159–168, Tórshavn, Faroe Islands. University of Tartu Library.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, L  lio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timoth  e Lacroix, and William El Sayed. 2023. [Mistral 7B](#).
- 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.
- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. [The state and fate of linguistic diversity and inclusion in the NLP world](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.
- Marzena Karpinska, Nader Akoury, and Mohit Iyyer. 2021. [The perils of using Mechanical Turk to evaluate open-ended text generation](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1265–1285, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. [Natural questions: A benchmark for question answering research](#). *Transactions of the Association for Computational Linguistics*, 7:452–466.
- Seungyoung Lim, Myungji Kim, and Jooyoul Lee. 2019. [KorQuAD1.0: Korean QA Dataset for Machine Reading Comprehension](#). *arXiv preprint arXiv:1909.07005*.
- 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.
- Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. [TruthfulQA: Measuring how models mimic human falsehoods](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.
- Peng Liu, Lemei Zhang, Terje Farup, Even W. Lauvrak, Jon Espen Ingvaldsen, Simen Eide, Jon Atle Gulla, and Zhirong Yang. 2024. [NLEBench+NorGLM: A comprehensive empirical analysis and benchmark dataset for generative language models in Norwegian](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5543–5560, Miami, Florida, USA. Association for Computational Linguistics.
- Shayne Longpre, Yi Lu, and Joachim Daiber. 2021. [MKQA: A linguistically diverse benchmark for multilingual open domain question answering](#). *Transactions of the Association for Computational Linguistics*, 9:1389–1406.
- Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. [Can a suit of armor conduct electricity? a new dataset for open book question answering](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391, Brussels, Belgium. Association for Computational Linguistics.
- Timo M  ller, Julian Risch, and Malte Pietsch. 2021. [GermanQuAD and GermanDPR: Improving non-English question answering and passage retrieval](#). In *Proceedings of the 3rd Workshop on Machine Reading for Question Answering*, pages 42–50, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- 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.
- Anna Rogers, Matt Gardner, and Isabelle Augenstein. 2023. [QA Dataset Explosion: A Taxonomy of NLP Resources for Question Answering and Reading Comprehension](#). *ACM Computing Surveys*, 55(10):1–45.
- Javier de la Rosa, Vladislav Mikhailov, Lemei Zhang, Freddy Wetjen, David Samuel, Peng Liu, Rolv-Arild Braaten, Petter M  hlum, Magnus Breder Birkenes, Andrey Kutuzov, et al. 2025. [The Impact of Copyrighted Material on Large Language Models: A Norwegian Perspective](#). In *Proceedings of the Joint 25th*

Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025), Tallinn, Estonia.

Sebastian Ruder. 2021. Challenges and Opportunities in NLP Benchmarking.

David Samuel, Vladislav Mikhailov, Erik Velldal, Lilja Øvrelid, Lucas Georges Gabriel Charpentier, and Andrey Kutuzov. 2025. Small Languages, Big Models: A Study of Continual Training on Languages of Norway. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, Tallinn, Estonia.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. 2024. [Detecting Pretraining Data from Large Language Models](#). In *The Twelfth International Conference on Learning Representations*.

ByungHoon So, Kyuhong Byun, Kyungwon Kang, and Seongjin Cho. 2022. JaQuAD: Japanese Question Answering Dataset for Machine Reading Comprehension. *arXiv preprint arXiv:2202.01764*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. [CommonsenseQA: A question answering challenge targeting commonsense knowledge](#). 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 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. [Constructing datasets for multi-hop reading comprehension across documents](#). *Transactions of the Association for Computational Linguistics*, 6:287–302.

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.

Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied, Weizhu Chen, and Nan Duan. 2024. [AGIEval: A human-centric benchmark for evaluating foundation models](#). In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2299–2314, Mexico City, Mexico. Association for Computational Linguistics.

Incorporating Target Fuzzy Matches into Neural Fuzzy Repair

Tommi Nieminen and Jörg Tiedemann and Sami Virpioja

University of Helsinki, Dept. of Digital Humanities

firstname.lastname@helsinki.fi

Abstract

Neural fuzzy repair (NFR) is a simple implementation of retrieval-augmented translation (RAT), based on data augmentation. In NFR, a translation database is searched for translation examples where the source sentence is similar to the sentence being translated, and the target side of the example is concatenated with the source sentences. We experiment with introducing retrieval that is based on target similarity to NFR during training. The results of our experiments confirm that including target similarity matches during training supplements source similarity matches and leads to better translations at translation time.

1 Introduction

Retrieval-augmented translation (RAT) is a family of machine translation (MT) approaches where an MT system has access to translation examples when generating a translation for a source sentence. The translation examples are usually retrieved from a translation database based on similarity with the current translation context, which can be either the source sentence alone or a combination of the source sentence and the translation that has been generated so far. The similarity between the translation context and the translation examples from the database can be measured using lexical methods, such as edit distance and longest matching N-gram, or based on the distance between the vector representations of the example and the translation context. The intuition behind RAT is that the MT system can, given an unseen source sentence, use the retrieved matches as additional information when constructing a translation. This supports the translation task, as the MT system no longer has to rely solely on the informa-

tion embodied in the neural network, and different RAT methods have been shown conclusively to improve MT quality (see for example Bulte and Tezcan (2019), Khandelwal et al. (2021)).

This article focuses on a variant of RAT based on augmenting data with lexical matches, first discussed in Bulte and Tezcan (2019), called Neural Fuzzy Repair (NFR). Our work further develops NFR by incorporating translation examples that have been retrieved based on target instead of source similarity. We also test how annotating source sentences with the similarity levels of the translation examples affects quality.

2 Related work

Many RAT approaches draw inspiration from retrieval methods that have been used in professional translation from the 1960s onward. The three main traditional forms of retrieval in professional translation (Hutchins, 1998) are terminology lookup from a terminology database, full segment fuzzy match retrieval from a translation memory (usually based on edit distance), and concordance search from a translation memory (retrieving translation pairs based on the occurrence of a particular substring on the source side). In recent decades, various subsegmental retrieval methods have also been introduced (Flanagan, 2014).

In MT research prior to the adoption of neural machine translation (NMT), the concept of retrieving translation examples based on source similarity and the construction of new translations from the retrieved examples was first proposed in the 1980s in the form of example-based MT (Nagao, 1984). In statistical MT, retrieving parts of existing translations from translation tables in order to generate new translations was a core component of MT systems, and there were also attempts to integrate translation memory retrieval more directly in a manner resembling RAT (Koehn and Senellart, 2010).

Within NMT, various RAT methods have been proposed. They can be roughly divided into three categories, depending on whether they are based on purpose-built neural network architectures, data augmentation, or changes in the decoder component of the MT system.

Gu et al. (2017) introduces the first NMT architecture designed for RAT: translation examples are retrieved from a translation database based on sentence similarity, and the attention component of the MT system is extended to cover the retrieved examples. Bapna and Firat (2019) uses a similar architecture-based approach, but uses N-gram- and vector-based retrieval to increase the amount of matches. Hoang et al. (2022) attempts to control the source-match interactions by encoding each retrieved match separately with the source sentence.

RAT based on data augmentation was introduced in Bulte and Tezcan (2019), where source sentences are concatenated with target translations from translation examples that are retrieved from the translation database with lexical matching. Xu et al. (2020) extends the lexical matching to separate relevant and irrelevant target tokens by using word alignment data, and also utilizes matches based on vector similarity. Concatenation-based data augmentation methods are also used to constrain MT output to contain terms from a terminology database (Dinu et al., 2019), which can be considered a form of RAT.

Decoder-based RAT has the advantage of being usable with any NMT model, since the model parameters and architecture are not changed. One early implementation utilized phrase tables from SMT systems (Dahllmann et al., 2017). Currently, the most prominent form of decoder-based RAT is kNN-MT (Khandelwal et al., 2021), which generates a datastore consisting of pairs of translation contexts and output tokens. When generating the next token of a translation, the decoder searches for similar translation contexts based on vector similarity, and utilizes the output tokens corresponding to the most similar translation contexts in generating the next token.

Neural RAT has also been implemented with large language models (LLM) (Moslem et al., 2023) using in-context learning (ICL), where the LLM is prompted with the retrieved examples. Bouthors et al. (2024) compare LLM-based RAT with NFR, and NFR seems to have a clear quality

advantage, although more advanced LLMs may have better results.

3 NFR with lexical matches

In NFR, the source language sentences in the training data are concatenated with target language sentences. The concatenated target language sentences originate from translation examples, where the source sentence is similar to the source sentence in the training data by some similarity measure. The concatenated target language sentences are separated from each other and the source sentence with a special symbol, and maximum amount of examples per sentence is usually limited to 3 (see Table 1 for examples).

NFR has been implemented using both lexical and vector-based retrieval methods (Tezcan and Bulté, 2022). It is easier to conceptualize with lexical retrieval methods, since there is a clear mechanism for utilizing the retrieved matches: find parts of the retrieved translation that match the parts of the new source sentence, and copy them to the new translation. Note that this copy behaviour has to be selective in two ways:

1. **Match selection:** The MT system may be provided with irrelevant or contradictory examples (if the system is designed to support multiple translation examples), so the system must be able to discard examples or to select the most appropriate one amongst multiple valid examples.
2. **Sub-sentential selection** Given relevant translation examples, the MT system has to identify the parts of the examples that can be exploited for constructing new translations and then adapt them correctly.

With vector-based retrieval methods, the mechanism for utilizing the matches is more murky, as there is often no lexical similarity with the retrieved translations and any acceptable translation for the new source sentence. Xu et al. (2020) found that using vector-based matches improves translation quality (although not by as much as lexical matches), and they hypothesize that vector-based matches improve quality by providing context during translation.

One issue, which is not explored in the existing research literature, is how a RAT system actually learns to utilize the retrieved matches.

Fuzzies	Augmented source sentence
1	Tuensaaajia on kaksi . FUZZYBREAK There are two situations .
2	Turvallisuutta koskevat lisä vaatimukset FUZZYBREAK Käyttövarmuutta koskevat vaatimukset FUZZYBREAK Security requirements
3	Toimivaltaisen viranomaisen tehtävät ja velvollisuudet FUZZYBREAK Välimiesten tehtävät ja velvoitteet FUZZYBREAK HRE:n tehtävät ja velvollisuudet FUZZYBREAK Duties and obligations of children

Table 1: Source sentences (the English sentence after the last FUZZYBREAK delimiter symbol) augmented with 1 to 3 target sentences from similar translation examples (Finnish sentences separated by the delimiter symbols). Highlighted text indicates matching source and target portions.

For instance, for the MT model to learn the sub-sentential selection behaviour associated with lexical matches, it would seem necessary for the training data to contain examples consisting of a source sentence, one or more translations from retrieved translation examples, and a target sentence containing parts of those retrieved translations. However, in the existing RAT literature, the matches are retrieved based on source similarity, with no concern for whether any part of the target sides of the matches are actually present in the translations of the training data. The only article in which the target side similarity of the retrieved examples is discussed is Xu et al. (2020), where in one experiment source-side matches are re-ranked according to target-side similarity. Otherwise, there seems to be an implicit assumption that source-side similarity implies target-side similarity. However, most naturally occurring sentences have billions of possible translations (Dreyer and Marcu, 2012). Even though most of those possible translations are slight variations of other translations, for most sentences there is a large amount of valid translations that are meaningfully different both lexically and syntactically, as is demonstrated by the literature on increasing output diversity in machine translation (see for instance Roberts et al. (2020)).

This diversity in naturally occurring translations makes it unlikely that most translation pairs retrieved from naturally occurring data are optimal training examples for the copy behaviour that a RAT system should exhibit. However, since RAT systems trained with such data have been conclusively shown to improve translation quality and to copy tokens from the target sides of the retrieved matches to the new translations more often than

normal MT systems (Xu et al., 2020), there must be enough good examples of copy behaviour in the training data. However, it is likely, that a large part of the lexical matches that are retrieved with source similarity do not exemplify the sub-sentential copy mode, but rather contextualize the translation in the same way as vector-based matches.

The objective of this work is to verify whether having training data that contains more suitable training examples of the expected selective copy behaviour improves the performance of NFR models. To obtain such training data, we retrieve lexical matches based on target similarity during the training phase. One issue with using target similarity at training time is that a model that is trained only with target similarity data cannot learn the first type of selective copy behaviour explained above, match selection. There will be no examples in the training data of irrelevant or contradictory matches, since all matches will be similar to their respective target sentences. Since at inference time, only matches based on source similarity will be available, the model will almost certainly copy irrelevant tokens from irrelevant matches to the output. On the other hand, the data is more conducive to learning the second type of copy behaviour, sub-sentential selection, since all the training examples are relevant for that purpose.

In our experiments, we attenuate the problem of copying irrelevant tokens by adding source-similarity matches to the target-similarity training data, and by ensembling source- and target-similarity models. We also include similarity class annotations in most models (a numerical suffix from 5 to 9 attached to the example marker), indicating the degree of similarity that each translation example has with the source or target sentence,

with the aim of training the model to process examples from different classes differently (for instance to copy less from low-similarity examples).

4 Data

Models are trained using the English to Finnish data from the Tatoeba-Challenge data set (release v2023-09-26) (Tiedemann, 2020). This data set consists of most of the data included in the OPUS corpus collection¹ at the date of the release. The data in OPUS includes many crawled data sets. Due to quality issues in crawled data (Kreutzer et al., 2022), the data is filtered with Bicleaner AI v2.0 (Zaragoza-Bernabeu et al., 2022): 5 million best sentence pairs according to Bicleaner AI are included in the training set (referred to as *Train-5M* from here on). During the initial experiments, we noticed that even after Bicleaner AI cleaning, much of the crawled data was of very low quality (containing for instance machine translations and lists of SEO terms). The crawled data also contains many repetitive text templates, which occur hundreds of times with small changes, such as *You can fly from [X] to [Y] indirect via [Z]* or *[WORD] pronunciation in [LANGUAGE]*. We suspected that the presence of these repetitive similar sentences in the training data (often with substandard translations) would affect the RAT training adversely. Because of these concerns, we decided to create another training set, which consists of 5 million best scoring non-crawled sentence pairs in the Tatoeba-Challenge data set (referred to as *NC-Train* from here on).

RAT can be used for **domain adaptation** by using a domain-specific translation database for retrieval. In order to test the domain adaptation performance of our RAT models, we exclude a portion of the Tatoeba-Challenge data set as domain test data. As there are no domain annotations included in the data, we treat each individual corpus in the dataset as a separate pseudo-domain and extract at most 1,000 sentence pairs from each of them as domain test sets. The corpora in the dataset mostly map to actual domains, e.g. the EMEA corpus contains data that is mostly from the pharmaceutical/medical domain. The crawled corpora are an exception, as they contain data from many domains, and they are therefore excluded from the domain test data. The domain test data is excluded from the training sets.

¹<https://opus.nlpl.eu/>

Each 5 million sentence pair training set is used as a database from which the translation examples are retrieved during the training phase for its respective training set. The training set database is also used as a translation database during testing. We also use a larger *All-Filtered* database consisting of all of the Tatoeba-Challenge data with a BiCleaner-AI score of at least 0.7 for testing. The *All-Filtered* database is used to determine whether the RAT system is capable of utilizing matches that it has not seen during training. For the domain-specific test sets, we also use domain-specific translation databases, which consist of all the domain-specific data in the *All-Filtered* database. For the *NC-Train*, the crawled data is excluded from the *All-Filtered* database.

4.1 Retrieving translation examples

Retrieving similar sentences from a large database for the millions of sentences in the training set is computationally costly, so expensive similarity metrics such as edit distance cannot be directly used. The training database needs to be filtered with a fast method that approximates more sophisticated methods, so that the more accurate similarity metrics can be applied to a smaller set of translation examples. Multiple retrieval methods have been proposed for RAT, but according to Bouthors et al. (2024), the choice of retrieval strategy does not have a noticeable effect on NFR performance. Because of this, we use the open-source *fuzzy-match* library² and do not explore other retrieval strategies. *fuzzy-match* uses suffix arrays for the initial filtering, and then calculates the edit distance over the resulting filtered set of translation examples. The search is performed on sentences tokenized to words. Note that this means that the morphological complexity of the language will affect the number of matches that are found: fewer matches will be found for morphologically complex languages in otherwise identical scenarios, as tokens tend to contain more morphemes and are therefore more varied.

To retrieve similar sentences for the sentence pairs in the training set, we first search the training database (*Train* and *NC-Train* for source similarity, *Train-TS* and *NC-Train-TS* for target similarity) for a maximum of 100 matches with a *fuzzy-match* edit distance score of at least 0.5 (with 1 being identical and 0 completely different). Per-

²<https://github.com/SYSTRAN/fuzzy-match>

Data set	DB	0.9-0.99	0.8-0.89	0.7-0.79	0.6-0.69	0.5-0.59	Total
Train	Train	1,085,811	1,659,270	1,461,893	2,248,088	3,214,326	9,669,388
Train	Train-TS	680,150	1,957,426	1,290,774	1,914,313	2,308,767	8,151,430
NC-train	NC-train	855,918	2,098,593	1,650,462	2,465,417	3,118,555	10,188,945
NC-train	NC-train-TS	680,150	1,957,426	1,290,774	1,914,313	2,308,767	8,151,430

Table 2: Amounts of translation examples retrieved for each data set and translation database. The examples are divided into five classes of with different similarity ranges, which are indicated on the header row.

Data set	DB	0.9-0.99	0.8-0.89	0.7-0.79	0.6-0.69	0.5-0.59	Total
Train	Train	250,475	375,277	335,332	523,819	933,652	2,418,555
Train	Train-TS	213,722	358,967	278,138	424,350	712,278	1,987,455
NC-train	NC-train	202,711	441,879	383,819	577,702	871,335	2,477,446
NC-train	NC-train-TS	167,540	439,522	335,202	494,354	670,596	2,107,214

Table 3: Amounts and classes of translation examples that were actually used to augment the data sets, with 1 matches max per sentence (the counts are somewhat larger with training sets that allow multiple matches).

fect matches are excluded from the results. Subsets of the matches are then selected randomly to augment the training data with translation examples. A maximum of three matches out of the possible hundred are actually used in our experiments, but retrieving the extra matches makes it possible to vary the examples based on their mutual similarity and to control the distribution of examples of different similarity scores in the training data. We use the contrastive retrieval functionality of *fuzzy-match* with a value of 0.7 to increase diversity in the retrieved examples. See Table 3 for details on the retrieved examples.

5 Models

We trained several models in the English to Finnish translation direction with both the *Train* and *NC-train* datasets. All the models are standard *transformer-base* models and were trained with the Marian NMT toolkit (Junczys-Dowmunt et al., 2018) v1.11.13 using default settings. We use SentencePiece (Kudo and Richardson, 2018) to create a vocabulary of 50,000 symbols, which includes marker symbols for indicating different similarity classes. A shared vocabulary is used for both source and target to facilitate the copying of tokens from the examples to the translation. All the models were trained to convergence.

The validation sets were selected from the development set included in the Tatoeba-Challenge data set by picking the longest sentences for which

retrieved examples were available (the development set skews towards short sentences, which are problematic from the point of view of example retrieval). The validation sets were augmented using the same schemes that were used with the training data. This differs from test time, where only source similarity augmentation is used, but initial experiments indicated that using a different augmentation scheme for validation than the one used in the training data leads to unstable validation scores.

5.1 Augmentation schemes

The following augmentation schemes were used:

Baseline: A standard transformer model trained with non-augmented data.

Src-Sim: This is the standard augmentation scheme from Bulte and Tezcan (2019). Examples are retrieved based on source similarity. This can be considered the NFR baseline.

Trg-Sim: Examples are retrieved based on target similarity.

Combo: Sentence pairs from *Src-Sim* and *Trg-Sim* sets are combined. We test both combining all the sentence pairs from both sets (doubling the training set size to 10 M, referred to as *2X-Combo*), and picking odd sentence pairs from one set and even sentence pairs from the other (original training set size, referred to as *Combo*).

Mix-Sim: This scheme is only used when multiple translation examples are allowed. *Mix-Sim* differs from *Combo* in that translation examples

from both *Src-Sim* and *Trg-Sim* sets can be used simultaneously to augment the same source sentence. There can be a maximum of one *Trg-Sim* example per a sentence, the rest of the examples are picked from the *Source-Sim* set.

Manual inspection in early testing confirmed that models trained with the *Trg-Sim* scheme were prone to copying irrelevant tokens from the translation examples, especially with short sentences. The motivation for the *Combo-Sim* and *Mixed-Sim* schemes is to attenuate this problem of over-copying by mixing in source similarity examples into the training set. Another approach that we used to attenuating this problem was to ensemble *Src-Sim* and *Trg-Sim* models, as Hoang et al. (2024) indicates that ensembling models with diverse strengths leads to larger quality improvements than ensembling similar models. As a comparison, we also ensemble different checkpoints of some models.

We train models that allow a minimum of 1 and a maximum of 1-3 examples. In the augmentation phase, the examples are picked randomly from the full list of retrieved examples and concatenated with the source sentence. For all augmentation schemes, we generate training files both with and without fuzzy classes. The fuzzy class of an example is indicated in the data by using class-specific delimiter markers. Table 3 shows the ranges of *fuzzy_match* scores for each of the five classes used.

6 Evaluation

The test sets which are commonly used for MT evaluation are a bad fit for RAT evaluation, as they generally have very few fuzzy matches available even in large translation databases. For instance, for the *flores-devtest*, matches were found for only 72 out of 1,012 sentences in the *All-Filtered* set. More matches are found for the WMT news test sets, but the news domain is otherwise not well suited for RAT, as it is more varied and less repetitive than other domains.

Because of these concerns, we compiled our own test set. We extracted a maximum of 1,000 sentence pairs from each of the corpora that compose the Tatoeba-Challenge data set. We compiled separate test sets for the *Train* (75,249 sentence pairs) and *NC-Train* (65,549 sentence pairs) models. These test sets are mainly designed for domain translation performance evaluation, so we

designate them as the *Domeval* and *Domeval-NC*. As the data has not been annotated with domain information, we use the sub-corpora as pseudo-domains.

For each sub-corpus, we build a *fuzzy_match* index using all the sentence pairs from that sub-corpus included in the respective *All-Filtered* set. We generate augmented versions of the source sentences of the *Domeval* sets using the subcorpus indexes, as well as the *Train* and *All-Filtered* indexes, and then translate the augmented *Domeval* source sentences using a model.

Domeval, 72,549 sents, with crawled data				
	Train DB		All-Filtered	
Scheme	BLEU	chrF	BLEU	chrF
Baseline	31.14	62.43	31.14	62.43
Src-Sim 1	32.32	62.66	41.08	66.95
-classes	32.67	63.06	41.14	67.17
Src-Sim 2	32.16	62.61	40.99	66.92
Src-Sim 3	31.92	62.45	40.32	66.35
-classes	32.01	62.56	39.69	65.88
Trg-Sim-1	31.87	62.52	40.23	66.62
-classes	32.08	62.54	40.58	66.67
Trg-Sim-2	31.49	62.22	39.60	66.07
Combo	32.38	62.76	41.21	67.14
2X-Combo	32.82	63.16	41.57	67.47
Mix-Sim-2	32.11	62.79	41.10	67.18
Mix-Sim-3	31.94	62.60	40.55	66.67
-classes	31.97	62.61	40.26	66.38

Domeval-NC, 65,549 sents, no crawled data				
	NC-Train DB		NC-All-Filtered	
Scheme	BLEU	chrF	BLEU	chrF
Baseline	30.86	62.35	30.86	62.35
Src-Sim 1	32.26	62.99	37.61	65.62
Trg-Sim-1	31.70	62.58	36.84	65.09
Combo	32.20	62.90	37.61	65.57

Table 4: Scores for all augmentation schemes. The scores are calculated over the whole *Domeval*, including sentences for which there are no examples. The results in the two tables are not directly comparable, but the relative performance of the models is similar. *-classes* indicates that a model has been trained without similarity class annotations.

We also evaluate the performance of the models on full *Domeval* set with the *Train* and *All-Filtered* databases to measure general translation performance. SacreBLEU (Post, 2018) is used to generate BLEU and chrF metric scores. Neural eval-

Domeval, 72,549 sents, with crawled data				
	Train		All-Filtered	
Ensemble	BLEU	chrF	BLEU	chrF
Baseline	31.14	62.43	31.14	62.43
Src-Sim-1 + Trg-Sim-1	32.99 (32.32)	63.27 (62.66)	41.69 (41.08)	67.48 (66.95)
2X-Combo + 2X-Combo	32.93 (32.82)	63.21 (63.16)	41.66 (41.57)	67.52 (67.47)
Src-Sim-1 + Src-Sim-1	32.93 (32.32)	63.18 (62.66)	41.54 (41.08)	67.36 (66.95)

Domeval-NC, 65,549 sents, no crawled data				
	NC-Train DB		NC-All-Filtered	
Ensemble	BLEU	chrF	BLEU	chrF
Baseline	30.86	62.35	30.86	62.35
Src-Sim-1 + Trg-Sim-1	33.11 (32.26)	63.62 (62.99)	38.41 (37.61)	66.20 (65.62)

Table 5: Ensemble scores. *Src-Sim-1+Src-Sim-1* and *2XCombo+2X-Combo1* are ensembles of different checkpoints of the same model. Values in parentheses indicate the metric scores for the model in the ensemble that had better scores individually. Note that the differences between the different ensembles in the upper table are not statistically significant.

uation metrics, such as COMET, have been found to be superior to lexical metrics, such as BLEU and chrF, in recent meta-evaluations (Freitag et al., 2022). However, in the context of evaluating RAT systems, it is desirable for metrics to reward copying parts of the translation examples to the translation. With lexical metrics, this happens to some degree (depending on the lexical similarity of the translation examples and reference translations). With neural metrics, the translations do not need to be lexically similar with the reference translations, which is usually their advantage, but it becomes a potential problem in the context of RAT evaluation. Lexical metrics have also been found to be adequate in contexts where they are used to evaluate similar MT systems (Kocmi et al., 2024), and all the models we compare share their training data, subword segmentation, and model architecture. Because of these factors, we decided to use only lexical evaluation metrics.

During test time, only examples retrieved based on source similarity are used, also with the models that were trained with target similarity, since target-side data would not be available in actual translation scenarios.

7 Discussion of the results

All results are in accordance with earlier evaluations of NFR in Bulte and Tezcan (2019) and Xu et al. (2020): NFR improves translation quality very significantly (up to 10 BLEU points) compared to a NMT baseline.

The domain translation results for the five domains with most retrieved translation examples (see Table 6) are more ambivalent, although it should be noted that two of the five domains are highly atypical. The *Open-Subtitles* corpus consists of subtitles of TV shows and films, which are typically very short in order to fit the screen and often non-literal, due to e.g. jokes and references to visual content. Consequently the metric scores are very low for the domain. The *bible-uedin* corpus receives very high scores, which is probably due to repetition in the corpus, which means that very similar translation examples are available for many sentences. The scores are higher for *Train*, indicating that the crawled data contains bible translations.

Evaluation of both full test sets and specific domains suggests that annotating similarity classes of examples in the source sentences degrades translation quality slightly compared to treating all examples in the same way. It should be noted, though, that for the *EMEA*, *DGT*, and *Mozilla-IOn* domains similarity class annotation does seem to improve translation quality. These are also domains that are well-suited for RAT, as they are repetitive and noncreative.

The *Trg-Sim* scheme underperforms all other schemes on its own, probably due to excessive copying from the retrieved matches. However, models combining source and target similarity matches perform better than pure *Src-Sim* models. In domain-specific evaluation, the best results are

Train: domain translation, domains with most matches, only matches from domain DB										
	Open-Subtitles (822)		EMEA (654)		DGT (589)		bible-uedin (523)		Mozilla-I10n (482)	
	BLEU	chrF	BLEU	chrF	BLEU	chrF	BLEU	chrF	BLEU	chrF
Src-Sim-1	28.15	53.60	58.89	77.31	69.78	82.60	93.38	96.01	69.23	79.04
-classes	28.35	53.72	58.17	77.99	69.87	82.62	93.35	96.07	68.19	78.37
Src-Sim-2	27.41	53.50	58.30	77.06	66.25	80.47	93.96	96.54	65.24	76.17
Src-Sim-3	28.77	54.03	57.42	76.62	61.58	76.38	93.84	96.42	62.48	74.68
-classes	28.99	54.20	55.82	75.23	59.66	75.14	93.52	96.13	59.65	70.78
Trg-Sim-1	17.43	43.85	57.25	78.11	70.97	83.63	79.24	88.52	69.96	80.11
-classes	20.35	44.55	56.62	78.01	70.21	82.76	91.52	95.24	69.39	79.52
Trg-Sim-2	14.63	39.42	56.57	75.68	65.83	80.15	79.73	88.80	64.51	76.05
Combo	24.71	51.77	59.15	78.98	71.04	83.28	93.11	96.06	70.50	80.30
2X-Combo	26.79	52.73	58.73	79.17	70.85	83.36	93.52	96.50	70.77	79.99
Mix-Sim-2	22.56	49.30	59.00	78.22	67.65	81.37	92.79	95.92	67.91	78.33
Mix-Sim-3	24.49	50.82	56.65	74.95	64.69	79.03	93.10	96.18	63.64	75.11
-classes	25.69	51.06	56.57	76.23	62.54	77.53	93.12	95.92	62.84	74.63
Src-Sim-1 + Trg-Sim-1	22.00	48.13	59.24	77.92	71.75	83.99	90.92	94.90	71.21	80.43
Src-Sim-1 + Src-Sim-1	28.49	54.21	58.74	77.39	70.57	83.06	93.56	96.09	69.36	78.67
2X-Combo+ 2X-Combo	26.84	52.70	58.80	79.30	71.18	82.90	93.55	96.58	70.74	80.24

NC-Train: domain translation with domain database, domains with most matches										
	Open-Subtitles (822)		EMEA (654)		DGT (589)		bible-uedin (523)		Mozilla-I10n (482)	
	BLEU	chrF	BLEU	chrF	BLEU	chrF	BLEU	chrF	BLEU	chrF
Src-Sim-1	29.94	54.48	60.00	79.91	72.47	84.86	89.36	94.20	69.81	79.52
Trg-Sim-1	18.25	44.51	57.91	78.67	72.05	84.28	75.28	86.59	68.95	78.61
Combo	27.66	53.24	60.08	79.76	73.23	84.99	87.21	93.03	69.64	79.52
Src-Sim-1 + Trg-Sim-1	22.84	49.23	59.61	79.61	73.02	84.99	85.36	92.06	71.38	80.41

Table 6: Domain translation BLEU and chrF metrics scores for all models and ensembles. The number in the parentheses under the domain name indicates how many sentences out of 1,000 had at least one translation example.

Train: domains with short sentences, only matches from domain DB								
	Ubuntu (313)		KDE (418)		GNOME (420)		WikiTitles (373)	
	BLEU	chrF	BLEU	chrF	BLEU	chrF	BLEU	chrF
Mix-Sim-3	66.30	78.34	68.12	80.23	67.21	78.95	56.71	74.16
Src-Sim-1	62.58	74.68	62.73	76.27	63.64	76.35	47.09	69.17
Trg-Sim-1	60.61	74.40	62.92	77.07	62.89	76.54	45.58	69.84

Table 7: Scores for domains with short sentences (max 5 words per line). Not all models are shown here, but Mix-Sim models perform best, notably against Src-Sim-1, which we use as NFR baseline.

obtained with the *Combo* models and the ensemble of *Src-Sim* and *Trg-Sim* models.

While the *Mix-Sim* scheme does not appear to work generally, it performs better than alternatives with a specific subgroup of domains, i.e. those with very short sentences (see Table 7). In general, models that allow multiple examples are better with short sentences. One reason for this is probably that more examples are available for shorter sentences. However, it might also be due to the long source sentences becoming too long when augmented with multiple translation examples, thus degrading performance.

8 Conclusion and future work

Our experiments demonstrate that both adding target similarity matches to the training data, and ensembling *Trg-Sim* models with *Src-Sim* models improve the quality of translation output compared to normal NFR. In the future, we plan to extend the *2X-ComboSim* approach by replicating source sentences with different source and target similarity matches in the training data at a larger scale.

We also plan to experiment further on ensembling NFR models, including ensembles of models trained with different numbers of translation examples. Ensembling may also offer an alternative way of handling multiple translation examples: a 1-example model can be provided with multiple translation examples as separate inputs, the outputs of which can then be ensembled to produce a translation that is influenced by all the examples. Ensembling could also be used to combine terminology models (Dinu et al., 2019) and NFR models, by preparing separate inputs annotated with terminology and translation examples respectively, and ensembling the outputs.

References

- Ankur Bapna and Orhan Firat. 2019. <https://doi.org/10.18653/v1/N19-1191> Non-parametric adaptation for neural machine translation. 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 1921–1931, Minneapolis, Minnesota. Association for Computational Linguistics.
- Maxime Bouthors, Josep Crego, and François Yvon. 2024. <https://doi.org/10.18653/v1/2024.findings-naacl.190> Retrieving examples from memory for retrieval augmented neural machine translation: A systematic comparison. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3022–3039, Mexico City, Mexico. Association for Computational Linguistics.
- Bram Bulte and Arda Tezcan. 2019. <https://doi.org/10.18653/v1/P19-1175> Neural fuzzy repair: Integrating fuzzy matches into neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1800–1809, Florence, Italy. Association for Computational Linguistics.
- Leonard Dahlmann, Evgeny Matusov, Pavel Petrushkov, and Shahram Khadivi. 2017. <https://doi.org/10.18653/v1/D17-1148> Neural machine translation leveraging phrase-based models in a hybrid search. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1411–1420, Copenhagen, Denmark. Association for Computational Linguistics.
- Georgiana Dinu, Prashant Mathur, Marcello Federico, and Yaser Al-Onaizan. 2019. <https://doi.org/10.18653/v1/P19-1294> Training neural machine translation to apply terminology constraints. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3063–3068, Florence, Italy. Association for Computational Linguistics.
- Markus Dreyer and Daniel Marcu. 2012. <https://aclanthology.org/N12-1017> HyTER: Meaning-equivalent semantics for translation evaluation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 162–171, Montréal, Canada. Association for Computational Linguistics.
- Kevin Flanagan. 2014. <https://aclanthology.org/2014.tc-1.1> Filling in the gaps: what we need from TM subsegment recall. In *Proceedings of Translating and the Computer 36*, London, UK. AsLing.
- Markus Freitag, Ricardo Rei, Nitika Mathur, Chiklu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022. <https://aclanthology.org/2022.wmt-1.2/> Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 46–68, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor O. K. Li. 2017. <https://api.semanticscholar.org/CorpusID:3750771> Search engine guided non-parametric neural machine translation. *ArXiv*, abs/1705.07267.
- Cuong Hoang, Devendra Singh Sachan, Prashant Mathur, Brian Thompson, and Marcello Federico. 2022. <https://api.semanticscholar.org/CorpusID:252815975> Improving retrieval augmented neural machine translation by controlling source and fuzzy-match interactions. *ArXiv*, abs/2210.05047.
- Hieu Hoang, Huda Khayrallah, and Marcin Junczys-Dowmunt. 2024. <https://doi.org/10.18653/v1/2024.findings-naacl.35> On-the-fly fusion of large language models and machine translation. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 520–532, Mexico City, Mexico. Association for Computational Linguistics.
- John Hutchins. 1998. <https://api.semanticscholar.org/CorpusID:10644577> The origins of the translator’s workstation. *Machine Translation*, 13:287–307.
- Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. <http://www.aclweb.org/anthology/P18-4020> Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia. Association for Computational Linguistics.
- Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2021. <https://openreview.net/forum?id=wCBOFJ8hJM> Nearest neighbor machine translation. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.
- Tom Kocmi, Vilém Zouhar, Christian Federmann, and Matt Post. 2024. <https://doi.org/10.18653/v1/2024.acl-long.110> Navigating the metrics maze: Reconciling score magnitudes and accuracies. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1999–2014, Bangkok, Thailand. Association for Computational Linguistics.

- Philipp Koehn and Jean Senellart. 2010. <https://aclanthology.org/2010.jec-1.4> Convergence of translation memory and statistical machine translation. In *Proceedings of the Second Joint EM+/CNGL Workshop: Bringing MT to the User: Research on Integrating MT in the Translation Industry*, pages 21–32, Denver, Colorado, USA. Association for Machine Translation in the Americas.
- Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, Iroko Orife, Kelechi Ogueji, Andre Niyongabo Rubungo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. 2022. <https://doi.org/10.1162/tacl.a.00447> Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.
- Taku Kudo and John Richardson. 2018. <https://doi.org/10.18653/v1/D18-2012> SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.
- Yasmin Moslem, Rejwanul Haque, John D. Kelleher, and Andy Way. 2023. <https://aclanthology.org/2023.eamt-1.22> Adaptive machine translation with large language models. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 227–237, Tampere, Finland. European Association for Machine Translation.
- Makoto Nagao. 1984. <https://api.semanticscholar.org/CorpusID:18366233> A framework of a mechanical translation between japanese and english by analogy principle.
- Matt Post. 2018. <https://www.aclweb.org/anthology/W18-6319> A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.
- Nicholas Roberts, Davis Liang, Graham Neubig, and Zachary Lipton. 2020. <https://www.amazon.science/publications/decoding-and-diversity-in-machine-translation> Decoding and diversity in machine translation. In *NeurIPS 2020 Workshop on Resistance AI*.
- Arda Tezcan and Bram Bulté. 2022. <https://api.semanticscholar.org/CorpusID:245815894> Evaluating the impact of integrating similar translations into neural machine translation. *Inf.*, 13:19.
- Jörg Tiedemann. 2020. <https://www.aclweb.org/anthology/2020.wmt-1.139> The Tatoeba Translation Challenge – Realistic data sets for low resource and multilingual MT. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1174–1182, Online. Association for Computational Linguistics.
- Jitao Xu, Josep Crego, and Jean Senellart. 2020. <https://doi.org/10.18653/v1/2020.acl-main.144> Boosting neural machine translation with similar translations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1580–1590, Online. Association for Computational Linguistics.
- Jaume Zaragoza-Bernabeu, Gema Ramírez-Sánchez, Marta Bañón, and Sergio Ortiz Rojas. 2022. "https://aclanthology.org/2022.lrec-1.87" "bicleaner AI: Bicleaner goes neural". In "Proceedings of the Thirteenth Language Resources and Evaluation Conference", pages "824–831", "Marseille, France". "European Language Resources Association".

Constructions and Strategies in Universal Dependencies

Joakim Nivre

Uppsala University

Department of Linguistics and Philology

joakim.nivre@lingfil.uu.se

Abstract

Is the framework of Universal Dependencies (UD) compatible with findings from linguistic typology? One way to find out is to investigate whether UD can adequately represent constructions of the world's languages, as described in William Croft's recent book *Morphosyntax*. This paper discusses how such an investigation could be carried out and why it would be useful.

1 Introduction

Universal Dependencies (UD) is a framework for morphosyntactic annotation, designed to be applicable to all human languages and to enable meaningful cross-linguistic comparisons. The two versions of the guidelines are described in Nivre et al. (2016) and Nivre et al. (2020); a longer description of the underlying linguistic theory can be found in de Marneffe et al. (2021); and annotated data for 168 languages¹ can be found together with additional documentation on the UD website.²

But can UD really handle the full range of morphosyntactic variation in the world's languages? And is it successful in revealing similarities and differences across these languages in a systematic fashion? One way to approach these questions is to review the UD framework through the lens of linguistic typology. An early attempt to do this can be found in Croft et al. (2017), where the authors review version 1 of the UD guidelines and propose a number of improvements for better alignment with typological research findings, some of which were integrated in version 2 of the guidelines. Since then, William Croft has published the book *Morphosyntax* (Croft, 2022), a comprehensive survey of constructions in the world's languages, which brings together the results of sixty

years of research on typology and universals and thus provides an excellent basis for a new and more exhaustive review of the UD framework.

Croft's survey is based on two types of comparative concepts (Haspelmath, 2010; Croft, 2016): *constructions*, which are universal form-function pairings defined solely in terms of their function, and *strategies*, which are non-universal and defined by the pairing of a function with some cross-linguistically identifiable morphosyntactic form. Annotations in UD are not defined in terms of constructions and strategies, but for the framework to be universally applicable it must be possible to annotate all major constructions and strategies in the world's languages. And to support cross-linguistic comparisons, these annotations should ideally reflect systematic correspondances in constructions and strategies across languages. The purpose of this position paper is to motivate a more systematic study of these issues, by showing that we currently do not know to what extent UD satisfies these requirements, and to propose a research program to support this investigation.

The rest of the paper is organized as follows. In Section 2, I give a brief overview of the UD annotation framework, focusing on fundamental design principles; in Section 3, I outline the taxonomy of constructions and strategies in Croft (2022); and in Section 4, I discuss how constructions and strategies are annotated in UD. I conclude that, although the design principles of UD in some respects favor a clear representation of constructions and strategies, the correspondence between the two systems is far from perfect and merits further investigation.

2 The UD Annotation Scheme

The UD annotation scheme assumes that *words* are the basic units of morphosyntax. Words encode grammatical information internally through lexical stems and inflectional processes, but since the nature of these processes varies considerably

¹UD v2.15, released November 15, 2024.

²<https://universaldependencies.org>

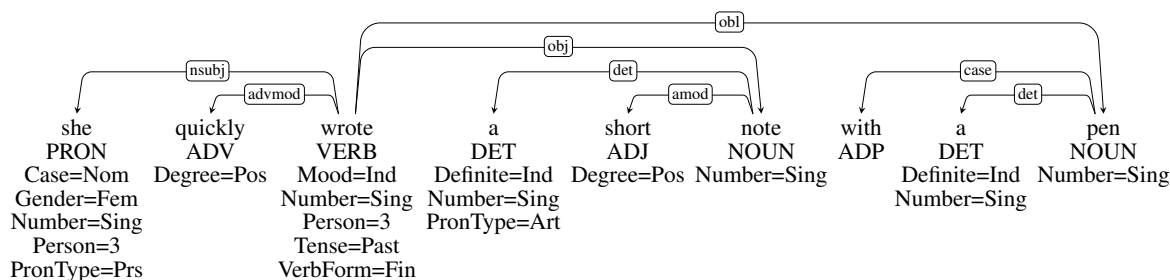


Figure 1: UD annotation of an English sentence.

across languages, there is no attempt to segment words into smaller units like morphs. Instead, the morphological annotation layer in UD combines coarse-grained part-of-speech tags with a rich inventory of morphological features, which together capture the information encoded in words without localizing it to smaller parts.³

Words also enter into syntactic relations with other words, and UD assumes that the information encoded in syntactic structure can be captured by a tree-structured representation consisting entirely of binary relations between words. A subset of these relations correspond to what grammarians would call dependency relations – asymmetric relations between a syntactic head and a dependent – but many of the relations that are necessary for a complete syntactic analysis are essentially symmetrical, even though the tree constraint forces one of the words to be (arbitrarily) chosen as the parent node. By way of illustration, Figure 1 shows the UD annotation of an English sentence.⁴

The syntactic analysis in UD assumes that all languages have *nominals*, which are the primary means of referring to entities, and *clauses*, which describe events (including actions and states). Both nominals and clauses can be further refined by *modifiers*, which describe attributes of entities or events. Figure 1 shows a main clause with the predicate *wrote* and three nominals: *she*, *a short note*, and *a pen*; there is also an adverbial modifier *quickly*, modifying the predicate *wrote*, and an adjectival modifier *short*, modifying the noun *note*.

A characteristic property of UD syntax is that it prioritizes direct relations between predicates, nominals and modifiers, rather than relations mediated by function words. Thus, in Figure 1, there is a direct relation from the predicate *wrote* to

the noun *pen*, denoting the instrument of writing, while the preposition *with* is essentially treated as a case marker on the noun. This treatment is motivated by the observation that predicates, nominals and modifiers are more likely to be parallel across languages than function words, which often correspond to morphological inflection (or nothing at all) when comparing across many languages.

3 Constructions and Strategies

The most central concept in Croft’s framework of morphosyntax is that of a *construction*, which is defined in the following way (Croft, 2022, p. 17):

construction: any pairing of form and function in a language (or any language) used to express a particular combination of semantic content and information packaging

It is worth noting that the functional side of a construction consists of two components, a semantic content and a particular way of packaging the information, also known as a propositional act. This is exemplified in Table 1, which shows constructions defined by different combinations of semantic classes and propositional acts, with the most prototypical constructions being *nominal phrases*, which refer to objects, *adjectival phrases*, which express property modification, and *verbal clauses*, which express action predication.⁵

Constructions at the most abstract level are universal and defined only in terms of function. However, to enable cross-linguistic comparison of constructions also in terms of their form, Croft introduces the notion of a *strategy* (Croft, 2022, p. 19):

strategy: a construction in a language (or any language), used to express a particular combination of semantic content and information pack-

³The morphological layer also includes lemmas, which are language-specific and will not be discussed here.

⁴For more information about tags, features, and relations, see <https://universaldependencies.org>.

⁵The prototypical constructions can be found along the diagonal from top left to bottom right in the first three rows of Table 1.

Semantic Class	Propositional Act		
	Reference	Modification	Predication
Object	Nominal Phrase Head: Noun	Possessive Modifier/Genitive Phrase	Predicate Nominal
Property	Property-Referring Phrase	Adjectival Phrase Head: Adjective	Predicate Adjectival
Action	Complement (Clause)	Relative Clause	Verbal Clause Head: Verb
All	Referring/Argument Phrase Head: Referent Expression	Attributive Phrase Head: Modifier	Clause Head: Predicate

Table 1: Grammatical constructions for combinations of three basic semantic classes and the three major propositional act (information packaging) functions (adapted from Croft (2022)).

aging (the ‘what’), that is further distinguished by certain characteristics of grammatical form that can be defined in a crosslinguistically consistent fashion (the ‘how’)

To exemplify the notion of strategy, let us consider the *predicate nominal* construction, which is “a clause construction defined by the function of predicating an object concept of a referent – that is, asserting what object category the referent belongs to”.⁶ Two common strategies for this construction are exemplified in (1) and (2–3).

- (1) Иван танцор
Ivan.NOM dancer.NOM
‘Ivan is a dancer’
- (2) Ivan är dansare
Ivan COP dancer
‘Ivan is a dancer’
- (3) Ivan is a dancer
Ivan COP a dancer

The Russian example in (1) uses a *zero* strategy (Stassen, 1997), which simply juxtaposes the referring expression Иван with the noun танцор in nominative case expressing the object concept. By contrast, the Swedish and English examples in (2) and (3) both use a *verbal copula* strategy (Stassen, 1997), where predication is mediated by a copula verb. The notion of strategy allows us to abstract over language-specific constructions and say that Swedish and English use the same strategy, while the Russian strategy is different.

4 Constructions and Strategies in UD

How are constructions and strategies represented in UD? At first sight, it may appear that they are not represented at all, because the UD annotation is centered on properties and relations of words.

⁶<https://comparative-concepts.github.io/cc-database/>

However, as noted in Section 2, the UD scheme systematically distinguishes clauses, nominals and modifiers. For example, a word with an incoming relation labeled *nsubj* must be the head of a nominal phrase, and a word with an incoming relation labeled *advcl* must be the head of a (subordinate) clause. So there is an almost perfect correspondence between the basic structures posited by UD – nominals, modifiers, and clauses – and the three major propositional acts in Croft’s framework: reference, modification, and predication.⁷ In addition, the UD principle of prioritizing direct relations between predicates, nominals and modifiers often reveals constructional parallelism across languages that use different strategies for a given construction.

To illustrate this, let us return to the predicate nominal construction and consider the UD annotation of (1–3) in Figure 2. All three representations share a structure $\text{NOUN} \xrightarrow{\text{nsubj}} \text{X}$, where X can be replaced by any category that can be the head of a referring expression. This captures the fact that the predicate nominal construction involves using a noun as a predicate, which would have been less clear if the copula verb had been treated as the head of the clause in Swedish and English. Moreover, the fact that Swedish and English uses the same strategy is captured by the presence of the structure $\text{NOUN} \xrightarrow{\text{cop}} \text{AUX}$, which contrasts with the absence of such a structure in Russian. In general, strategies often correspond to relations involving function words (like the *cop* relation).

The predicate nominal example suggests that UD representations can be decomposed into distinct substructures corresponding to constructions and strategies. Unfortunately, this is not true in

⁷The only discrepancy is that Croft’s notion of modification is restricted to modification of referring expressions, whereas the UD concept also includes adverbial modifiers and modifiers of modifiers.

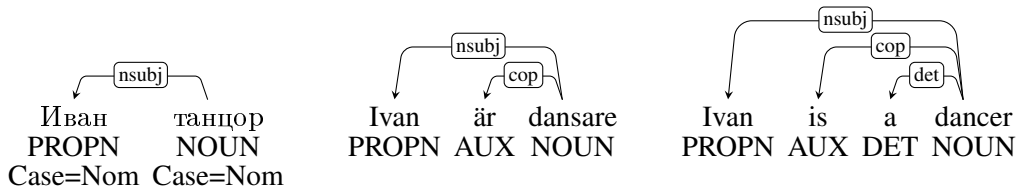


Figure 2: Simplified UD annotation for predicate nominal constructions in Russian, Swedish and English.

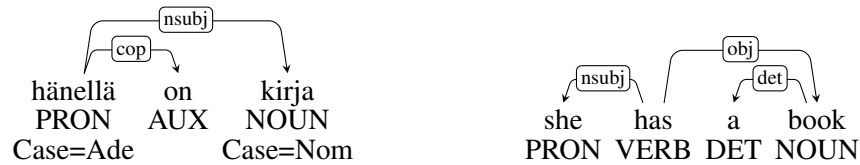


Figure 3: Simplified UD annotation for presentational possession constructions in Finnish and English.

the general case. First of all, it is clear that UD representations are more coarse-grained than constructions and strategies, so there will often be a one-to-many mapping from the former to the latter. For example, the substructure that is characteristic of the predicate nominal construction in Figure 2 would also be characteristic of an *equational* construction, as exemplified by *Ivan is the winner*, which in Croft’s framework is a distinct construction, even though the two constructions often share strategies through a process known as recruitment.

More importantly, it is not hard to find constructions where the UD representations completely fail to capture constructional parallelism. One example is the *presentational possession* construction, defined as “a presentational information packaging of the possession relation in which a possessum is introduced into the discourse, anchored by the possessor”⁸ and exemplified in Figure 3 with examples in Finnish and English. Finnish here uses a *locational possessive* strategy (Stassen, 2009), in which the possessum (*kirja* ‘book.NOM’) is expressed in a subject phrase, and the possessor (*hänellä* ‘her.ADESS’) in an oblique (locative) phrase, with a linking copula verb (*on* ‘be.3SG.PRES’). By contrast, English uses a *have-possessive* strategy (Stassen, 2009), where the possessor is expressed in a subject phrase (*she*), and the possessum in an object phrase (*a book*), connected by a full transitive verb (*has*). A closer comparison of the examples reveals that the two representations have next to nothing in common, which could capture the common construction, and also that the two strategies in this case involves

syntactic relations like *nsubj* and *obj*, which in the predicate nominal example were considered elements of the construction.

5 A Research Program for UD

Which of the two cases discussed above is typical? Are UD annotations mostly decomposable into parts corresponding to constructions and strategies, with a few anomalous cases like the presentational possession construction? Or is it the latter that is the norm, and the former the exception? At this point, we simply do not know, and this is the main motivation for proposing a research program that systematically investigates how constructions and strategies can be represented in UD, using the survey in Croft (2022) as a starting point. More precisely, I propose to develop a *constructicon* for UD, consisting of the following components:

- An inventory of universal constructions.
- For each construction, an inventory of common strategies for realizing that construction in the world’s languages.
- For each construction-strategy pair, a cross-linguistically valid UD analysis and representative examples from different languages.

Why should we build such a resource and how can we hope to construct it? Starting with the *why*, I believe that a UD constructicon could help us improve cross-linguistic annotation consistency by providing a complementary view of the UD guidelines, which is holistic and onomasiological. It is holistic because it starts from complete constructions rather than particular syntactic relations, and it is onomasiological because it goes from function

⁸<https://comparative-concepts.github.io/cc-database/>

to (cross-linguistically identifiable) form. This would in particular benefit the annotation of new languages, where guidelines could be developed systematically by first identifying what strategies are used for different constructions. It would also provide better support for construction-based annotation on top of UD, as proposed in Weissweiler et al. (2024). Last but not least, it would help us find out to what extent UD can represent constructions and strategies systematically and transparently across languages and thereby identify shortcomings in the current guidelines.

Returning to the question of *how* to build the construction, we can fortunately bootstrap the process by taking the first two components – the inventories of constructions and strategies – directly from Croft (2022), or rather from MoCCA, the database of comparative concepts that is being developed from the glossary of the book (Lorenzi et al.).⁹ We can then concentrate on constructing valid UD analyses for all construction-strategy pairs, starting with the most prototypical construction types – reference, modification and predication – and proceeding to non-prototypical cases with more complex variation patterns. Examples for all constructions can be found in Croft (2022), which contains at least one concrete example for every construction-strategy pair discussed in the book. This should be supplemented with examples from existing UD treebanks, which will allow us to assess the cross-linguistic annotation consistency for different constructions and strategies.

6 Conclusion

In this paper, I have reopened the question of whether UD is an adequate annotation framework from the point of view of linguistic typology, previously raised by Croft et al. (2017). I have argued that one way of answering this question is to study more systematically how constructions and strategies, in the sense of Croft (2022), can be represented in UD, and I have proposed that this can be done by building a construction for UD.

Acknowledgments

Thanks to Bill Croft for valuable comments on a draft of this paper and to members of the UniDive COST Action (CA21167) for useful discussions. Swedish Research Council grant no. 2022-02909.

⁹<https://comparative-concepts.github.io/cc-database/>

References

- William Croft. 2016. Comparative concepts and language-specific categories: Theory and practice. *Linguistic Typology*, 20(2):377–393.
- William Croft. 2022. *Morphosyntax: Constructions of the World's Languages*. Cambridge University Press.
- William Croft, Dawn Nordquist, Katherine Looney, and Michael Regan. 2017. Linguistic typology meets Universal Dependencies. In *Proceedings of the 15th International Workshop on Treebanks and Linguistic Theories (TLT15)*, pages 63–75.
- Martin Haspelmath. 2010. Comparative concepts and descriptive categories in crosslinguistic studies. *Language*, 86:663–687.
- Arthur Lorenzi, Peter Ljunglöf, Ben Lyngfelt, Tiago Timponi Torrent, William Croft, Alexander Ziem, Nina Böbel, Linnéa Bäckström, Peter Uhrig, and Ely A. Matos. MoCCA: A model of comparative concepts for aligning constructions. In *Proceedings of the 20th Joint ACL – ISO Workshop on Interoperable Semantic Annotation*, pages 93–98.
- Marie de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47:255–308.
- Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Dan Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC)*, pages 1659–1666.
- Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Dan Zeman. 2020. Universal Dependencies v2: An ever-growing multilingual treebank collection. In *Proceedings of the 12th International Conference on Language Resources and Evaluation (LREC)*, pages 4034–4043.
- Leon Stassen. 1997. *Intransitive Predication*. Oxford University Press.
- Leon Stassen. 2009. *Predicative Possession*. Oxford University Press.
- Leonie Weissweiler, Nina Böbel, Kirian Guiller, Santiago Herrera, Wesley Scivetti, Arthur Lorenzi, Nurit Melnik, Archana Bhatia, Hinrich Schütze, Lori Levin, Amir Zeldes, Joakim Nivre, William Croft, and Nathan Schneider. 2024. UCxn: Typologically informed annotation of constructions atop Universal Dependencies. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 16919–16932.

Finnish SQuAD: A Simple Approach to Machine Translation of Span Annotations

Emil Nuutinen, Iiro Rastas and Filip Ginter

TurkuNLP, Department of Computing

University of Turku, Finland

`emenuu, iiro.t.rastas, figint@utu.fi`

Abstract

We apply a simple method to machine translate datasets with span-level annotation using the DeepL MT service and its ability to translate formatted documents. Using this method, we produce a Finnish version of the SQuAD2.0 question answering dataset and train QA retriever models on this new dataset. We evaluate the quality of the dataset and more generally the MT method through direct evaluation, indirect comparison to other similar datasets, a backtranslation experiment, as well as through the performance of downstream trained QA models. In all these evaluations, we find that the method of transfer is not only simple to use but produces consistently better translated data. Given its good performance on the SQuAD dataset, it is likely the method can be used to translate other similar span-annotated datasets for other tasks and languages as well. All code and data is available under an open license: data at [HuggingFace TurkuNLP/squad_v2.fi](#), code on [GitHub TurkuNLP/squad2-fi](#), and model at [HuggingFace TurkuNLP/bert-base-finnish-cased-squad2](#).

1 Introduction

Question answering (QA) is an important practical information retrieval task as well as a common benchmark of computational models of human language. Extractive QA models are typically built as a two step retriever-reader pipeline, first retrieving the documents relevant to the query (retriever) and then using an encoder model to extract the correct answer span from those documents (reader). Generative QA models replace the reader component with a generative large

language model (LLM), in an approach commonly referred to as retrieval-augmented generation (RAG).

No matter which of the QA paradigms is applied, large-scale question answering datasets such as the SQuAD dataset play a key role. Both in terms of benchmarking model performance, and model training. Whereas for extractive QA these datasets are used directly, in generative LLM development, QA datasets are commonly used as a source of examples for instruction fine-tuning. Unfortunately, these large-scale QA datasets are mostly available only for English and a small number of well-resourced languages, making the direct development of retriever-reader QA models for languages without such a dataset almost impossible, as well as negatively impacting benchmarking of LLM-based QA.

With the improvements to machine translation (MT) output quality seen in the recent years, machine translating datasets is becoming a frequent choice to obtain a dataset in a new language in cases where native annotation is not possible due to lack of resources. While such an approach is technically very simple to implement for datasets consisting of unannotated text, it becomes considerably more complex for datasets with dense text span annotations, such as the QA datasets. Numerous approaches have been introduced aiming to transfer the span annotations during translation. In this paper we contribute to this overall line of research by demonstrating a simple, yet effective approach to translate English question answering datasets to Finnish (or other languages) using a little-known feature of the DeepL machine translation service.

The primary contribution of this paper is a Finnish version of the publicly available sections of the SQuAD 2.0 dataset. This dataset can serve both for the development of extractive QA systems on top of Finnish encoder models, as well

as provide a source of Finnish data for instruction tuning and benchmarking of Finnish LLMs. Our other contribution is a Finnish extractive QA model trained on this dataset.

The paper is organized as follows: In Section 2 we review prior work on machine translating QA-datasets. In Section 3 we explain our process of machine translating these datasets. In Section 4 we evaluate the new resource and compare it to other similar resources. Finally, Section 5 concludes the work.

2 Related Work

There are numerous open-domain question answering datasets for English. Among the most commonly used is the Stanford Question Answering Dataset (SQuAD). SQuAD1.1 (Rajpurkar et al., 2016) consists of 100,000 questions posed by crowdsourced workers on a set of text passages (paragraphs of 536 Wikipedia articles). The questions are produced by the workers, while the answers constitute spans present in the text passages. SQuAD2.0 (Rajpurkar et al., 2018) is a superset of SQuAD1.1 with an additional 50,000 crowdsourced unanswerable distractor questions that only make the impression of being answered in the given passage.¹ Native human generated question answering datasets for other languages include Chinese (Cui et al., 2019), Korean (Lim et al., 2019) and French (d’Hoffschmidt et al., 2020), but a large number of languages lack a large QA dataset.

The SQuAD dataset has been machine translated to several languages. Arabic (Mozannar et al., 2019) SQuAD1.1 version starts by machine translating the passages, questions and answers separately. Subsequently, all the paragraphs and answers are transliterated to Arabic and the span of text of length at most 15 words with the least edit-distance with respect to the answer is identified. Only 231 articles containing 48,344 question-answer pairs are translated, and a full 25,490 question-answer pairs are not recovered by the initial translation and the transliteration heuristic step is applied. A reported small-scale evaluation shows that approximately 64% of these are correctly recovered.

¹Note, however, that the test set of the SQuAD datasets is kept private, and the publicly available data contains 98,169 question-answer pairs for SQuAD1.1 and 92,749 answerable plus 49,434 unanswerable questions in SQuAD2.0.

Persian (Abadani et al., 2021) SQuAD2.0 version starts by machine translating the passages, questions and answers separately. Then an alignment is established by finding the position of the sentence that the answer appears in the English dataset. If the translated answer does not appear in the equivalent translated sentence, the question-answer pair is removed from the final dataset. The final dataset salvages 70,560 question-answer pairs.

The TAR-method (Translate-Align-Retrieve) used to create the Spanish translation of SQuAD1.1 (Carrino et al., 2019) also starts by machine translating the passages, questions and answers separately. If the translated answer can be found in the translated passage, it is retrieved as is. In the opposite case, a word alignment between the source and translated passage is established using the *eflomal* word alignment method (Östling and Tiedemann, 2016) and this alignment is then used to locate the translated answer. The final dataset salvages almost all of the question-answer pairs, but a manual error analysis showed that 50% of the answer spans were either misaligned (7%) or under-/over-extended (43%).

For Finnish, which is our target language of interest, there exists an earlier machine translated version of the SQuAD2.0 dataset (Kylliäinen, 2022; Kylliäinen and Yangarber, 2023). The passages, questions and answers are translated separately and their spans in the translations are identified using a number of normalization steps designed to improve the chance of successful matching. The dataset preserves 66,000 question-answer pairs from the original approx. 92,000.

The unpublished Swedish translation of SQuAD2.0² deviates from the common approach, and translates one question-answer pair at a time, marking the answer span with a recognizable token (e.g. “[0]”), and retrieves the span after translation, relying on the MT system preserving the special tokens. This process is reported to preserve 90% of the original question-answer pairs.

In a more recent approach, a separate alignment model is first trained for the target language (Masad et al., 2023). Then each context, question, and answer are translated together as a single unit using the Google Translate service. If the answer

²<https://towardsdatascience.com/swedish-question-answering-with-bert-c856ccdcc337>

is not found with exact matching from the translation, the alignment model is used. Finally, if the first two steps fail, the context is segmented into subsets of words with a total word count that approximates the word count of the answer. Then the embeddings of the answer and all the context segments are calculated using a pre-trained multilingual BERT model from which the closest segment to the answer is searched using cosine similarity with a threshold on the similarity score to prevent weak alignments. This method is reported to preserve 93.4% of the original question-answer pairs.

In another recent approach, an annotated clinical corpus is translated from English to Dutch (Seinen et al., 2024). In the dataset the annotation and the context are stored separately. In the paper the annotations are first integrated directly into the clinical text by enclosing the text span and the CUI (concept unique identifier) in square brackets ‘[[text span] [CUI]]’. Then the text with embedded annotations is machine translated, keeping the annotations intact. Finally the annotations are extracted from the translated text using regular expressions to separate the annotations and the context again to the original format. The Google Translate service and GPT 4 Turbo are compared. The Google Translate service lost up to 1.7% of annotations and GPT4 Turbo lost up to 5.9%. Most of lost annotations for Google were formatting errors, but for GPT, the lost annotations were mostly entirely omitted.

In summary, the clearly most common approach to machine translating datasets with span level annotations relies on translating the elements in isolation, and subsequently identifying through a varied set of heuristics their positions in the translated passages. This is naturally an error-prone process due to the fact that the answers when translated in isolation are not guaranteed to match their in-context translation within the passage, preventing reliable alignment. This is demonstrated by the substantial proportions of “lost” examples reported for most of these machine translated datasets. And while metadata-tagging approaches like that of Seinen et al. (2024) preserve most of the annotations, they are not able to preserve overlapping annotations without multiple rounds of translations.

In the following, we apply an approach which uses the functionality of a commercial MT engine to avoid the tedious alignment of answer segments

with the original passages.

3 Methods and Data

3.1 Markup-based Transfer

To create a translated version of the SQuAD dataset (or any other extractive QA dataset for that matter), not only the questions and underlying text passages need to be translated, but also the answer spans need to be correctly identified. Further, since the QA datasets often have many question-answer pairs for each passage, the answer spans may partially overlap.

Our work is based on the DeepL commercial machine translation service³ which is very popular among users thanks to its excellent translation output quality, which has also been reported in numerical benchmarks (e.g. Shaitarova et al. (2023)). In particular, we capitalize on the simple observation that DeepL is capable of translating formatted documents. This feature is crucial for professional translators—the primary users of the service—who need to translate not only the text of the source documents, but also preserve their formatting. In practice, this means that the input of DeepL can be a textual document with formatting (a Word document) and the service produces its translated version with the formatting preserved. This, in turn, gives us the combination of a high-quality machine translation system, an obviously necessary condition for successful machine translation of training data, with the ability to link text spans between the source and target documents through formatting. We first utilized this property of DeepL to machine translate a relation extraction dataset to a number of languages. In that work, the annotation did not exhibit overlapping spans (Bassignana et al., 2023).

The answer spans in the dataset can be trivially encoded as colored text spans in the input documents, where the color uniquely differentiates the individual answer spans. This is somewhat complicated by the fact that the answers may overlap in the dataset. A simple solution is to consider the overlapping region to be a separate span, and assign it a distinct color, and reverse this mapping when reconstructing the dataset after translation. Another approach would have been, for instance, to translate each context several times for different non-overlapping subsets of entities. Nevertheless, having observed that in our case the former

³<https://www.deepl.com/translator>

approach did not cause any clear degradation of the output, we chose to not pursue the latter approach, which would have increased the cost of translation⁴ and complexity of reconstructing the data. The translation process with formatting is illustrated on an actual example from the dataset in Figure 1.

Observing that oftentimes the answer spans were over-extended by a trailing punctuation symbol during translation, the only post-processing we apply is to strip from each translated span any trailing punctuation. This, in our view, has no negative impact on the QA task.

One aspect, common to all machine translation approaches to SQuAD irrespective of the method of annotation transfer, is that the answer spans in the original SQuAD data are always continuous, which is not necessarily the case in the translation simply due to the properties of the target language. In these cases, the translation system often correctly highlights the discontinuous regions in the translation, however the SQuAD data file format does not represent discontinuous answer regions, nor do the off-the-shelf model architectures developed for SQuAD allow for generation of discontinuous spans. To deal with this, and still allow the data to be used also with standard architectures, we include in the final data files both the original potentially discontinuous spans (as a separate key) and continuous spans obtained by simply spanning from the first to the last discontinuous span. In our dataset, only 2.6% of the answers are discontinuous, many of which are translation artefacts upon manual inspection.

3.2 Finnish SQuAD2.0

We used the method described above to machine translate the publicly available sections of the SQuAD2.0 dataset to Finnish. The resulting dataset preserves 90,233 question-answer pairs from the original 92,749, i.e. 97.2% of the dataset. This is substantially more than the majority of SQuAD machine translations discussed in Section 2

3.3 Finnish Extractive QA Models

We train an extractive QA model on the Finnish SQuAD dataset using the Finnish FinBERT-base model (Virtanen et al., 2019) and the standard ap-

⁴The overall translation cost of SQuAD was approximately 20€.

proach to span-detection with BERT models described by Devlin et al. (2019) and implemented in the Hugging Face Transformers library (Wolf et al., 2020). Since the English SQuAD2.0 test set is not publicly available, we fine-tune our model using only the train set and use the validation set for evaluation. This matches how most of the other reported models are trained and evaluated.

Interestingly, state-of-the-art performance models for the English SQuAD dataset almost uniquely rely on the ALBERT pre-trained model (Lan et al., 2020), with very substantial reported gains (Lan et al., 2020; Abadani et al., 2021) over the standard BERT models. In order to test whether a similar effect can be obtained also for Finnish, we also pre-train a series of Finnish ALBERT models (FinALBERT) and fine-tune them on the Finnish SQuAD dataset.

The pretraining of FinALBERT follows the original ALBERT model, with only a few differences. Based on the results of a grid search, the pretraining learning rate was set much higher than what was used to train the original ALBERT models, at $5.28e-3$. Additionally, the input length was gradually increased during pretraining, following the curriculum learning approach proposed by Nagatsuka et al. (2021). The training data used was identical to that used to train the FinBERT model and the same uncased tokenizer of FinBERT was also used for the FinALBERT models.

In the following section, we evaluate the FinSQuAD dataset, the MT method used, as well as the performance of the trained QA models.

4 Evaluation

One of the main challenges with machine translated datasets is the absence of a large-enough, manually annotated, representative test set. Such a test set is in many cases difficult to create, as it entails replicating the entire annotation task and procedure, which is a major undertaking for tasks with complex annotations, such as QA. Therefore, in addition to reporting model performance on the machine translated test set, we also carry out several other evaluations: a backtranslation experiment, a manual evaluation of the translated examples, and a comparison of our method respective to two other machine translated SQuAD datasets, one for Spanish and one for Finnish. These comparisons allow us in particular to establish the relative merits of our approach to other methods of

In 2011, documents obtained by WikiLeaks revealed that Beyoncé was one of many entertainers who performed for the family of Libyan ruler Muammar Gaddafi. Rolling Stone reported that the music industry was urging them to return the money they earned for the concerts; a spokesperson for Beyoncé later confirmed to The Huffington Post that she donated the money to the Clinton Bush Haiti Fund. Later that year she became the first solo female artist to headline the main Pyramid stage at the 2011 Glastonbury Festival in over twenty years, and was named the highest-paid performer in the world per minute.

WikiLeaksin saamista asiakirjoista 2011 kävi ilmi, että Beyoncé oli yksi monista esiintyjistä, jotka esiintyivät Libyan hallitsijan Muammar Gaddafin perheelle. Rolling Stone kertoi, että musiikkiteollisuus kehotti heitä palauttamaan konserteista ansaitut rahat; Beyoncé'n tiedottaja vahvisti myöhemmin The Huffington Postille, että hän lahjoitti rahat Clinton Bushin Haiti-rahastolle. Myöhemmin samana vuonna hänestä tuli ensimmäinen naisartisti, joka oli soolona pääesiintyjänä Pyramidin päälavalla vuoden 2011 Glastonbury-festivaaleilla yli kahteenkymmeneen vuoteen, ja hänet nimettiin maailman parhaiten palkatuksi esiintyjäksi minuutissa.

Documents obtained by WikiLeaks in 2011 revealed that Beyoncé was one of many performers who performed for the family of Libyan ruler Muammar Gaddafi. Rolling Stone reported that the music industry urged them to return the money earned from the concerts; Beyoncé's spokesperson later confirmed to The Huffington Post that she donated the money to the Clinton Bush Haiti Fund. Later that year, she became the first female artist to headline solo on the Pyramid main stage at the 2011 Glastonbury Festival in over twenty years, and was named the world's highest paid performer by the minute.

Figure 1: Example of the colored answer spans from an actual SQuAD passage: the original English passage (top), its Finnish translation (middle), and its backtranslation from Finnish into English (bottom). This example is shown as-is without any manual corrections (other than adjusting colors for better readability). Note the two overlapping answers *documents obtained by WikiLeaks* and *WikiLeaks* at the very beginning of the passage.

SQuAD machine translation.

We use as metrics the *exact match (EM)*, the proportion of questions that receive the exactly correct answer span, and *token F1*, the F1 score of the precision and recall of tokens in the predicted answer span, compared to the reference answer span. The latter metric is more tolerant to minor changes at the span boundaries.

4.1 QA Model performance

In Table 1, we compare the scores of our model to scores reported for other machine translated QA datasets. Our Finnish QA scores are the highest among those reported, well within the range that is to be expected with similar datasets. Of particular interest is the very substantial gain compared to the results Kylliäinen and Yangarber (2023) reported on the previously available Finnish translation of SQuAD2.0 but otherwise using a very comparable model. We will return to these results when discussing the relative merits of the machine translation methods later in Section 4.3.

For further comparison, we trained an English model based on a comparable pre-trained language model (BERT-base). This model reaches EM 74.2 and F1 77.6 on the original SQuAD2.0 data. The observed drop of 6.0pp EM and 3.9pp

F1 is a combined effect of, at least, (a) noise introduced during translation and (b) any possible effect of the target language being Finnish, rather than English.

To our disappointment, the results also indicate that the models based on FinALBERT are not notably better than the models based on FinBERT, i.e. we were unable to replicate on Finnish the improvements in QA performance reported for English with the ALBERT model architecture. We also note that the results of the Finnish models are more closely grouped in general compared to the SQuAD results presented by Lan et al. (2020). Further investigation is needed to ascertain whether this difference is due to the quality and amount of pre-training data used by the Finnish models, or something else entirely.

In the remainder of the Evaluation section, we turn our attention towards other means of evaluating our FinSQuAD dataset, as well as the MT method applied to produce it.

4.2 Evaluation through backtranslation

The relative ease, with which the annotation transfer method can be applied to any language pair supported by the machine translation service, allows for a backtranslation-based evaluation. Here

Model	Language and dataset	EM	F1	Reported in
BERT-base	Finnish SQuAD2.0 (ours)	68.2	73.7	this work
BERT-large	Finnish SQuAD2.0 (ours)	70.0	76.1	this work
ALBERT-xlarge	Finnish SQuAD2.0 (ours)	70.2	75.9	this work
BERT-base	Finnish SQuAD2.0 [1]	55.5	61.9	[1]
BERT-base	Spanish SQuAD2.0 [2]	63.4	70.2	online [3]
BERT-base	Swedish SQuAD2.0 [4]	66.7	70.1	online [4]
BERT-base	Indonesian SQuAD2.0 [5]	51.6	69.1	online [6]
BERT-base	Persian ParSQuAD [7]	62.4	65.3	[7]
BERT-base	English SQuAD2.0	74.2	77.6	this work

Table 1: Exact match (EM) and F1 scores of our models as well as scores reported for other machine translated SQuAD datasets as well as the original English SQuAD2.0. Citation list: [1] Kylliäinen and Yangarber (2023), [2] (Carrino et al., 2019), [3] (web source, 2021a), [4] (Okazawa, 2021), [5] (web source, 2021b), [6] (web source, 2021c), [7] (Abadani et al., 2021)

we translate our FinSQuAD data back to English, including the annotation transfer as if Finnish was the original language and English the target language. The resulting backtranslated English SQuAD dataset therefore accumulates errors over two rounds of translation, and can serve to estimate the impact on trained models due to errors incurred during the translation and annotation transfer.

In Table 2, we report model performance measured on the original English SQuAD2.0 test set, comparing a model trained on the original English training data, with a model trained on the backtranslated training data. We see a drop of 8.4pp in terms of exact match, and 5.1pp in terms of F1. Considering that these are the result of two cumulative translation and annotation transfer rounds, we can expect the loss incurred on the Finnish model, after one round of translation, to be less. If the errors were to be assumed as approximately evenly distributed between the two rounds of translation, the negative impact would be around 4.2pp EM and 2.6pp F1. This can be seen as a rather acceptable “price” for a dataset obtained without any manual annotation.

	EM	F1
Original	74.2	77.6
Backtranslated	65.8	72.5

Table 2: Exact match (EM) and F1 scores between the original English SQuAD2.0 dataset and eng-fin-eng translated English dataset.

4.3 Evaluation respective to other transfer methods

Direct comparison of the relative merits of our MT service -based annotation transfer method to its alternatives listed in Section 2 is challenging, as these methods are very tedious to implement and replicate for new languages.

Nevertheless, a direct comparison is possible to the Finnish SQuAD2.0 dataset by Kylliäinen (2022), which can be seen as an alternative translation of SQuAD2.0 to Finnish using a best-effort implementation of the translate-and-align approach. In all respects comparable QA models obtain F1 of 73.7 on our dataset compared to F1 of 61.9 on the dataset by Kylliäinen (2022). Further, our translation loses 2.7% of the original question-answer pairs in the process, compared to 28.1% lost in the other dataset. These results seem to suggest that the translation method we used produces data of superior quality compared to the translate-and-align approach.

As a second point of comparison, we choose the Spanish QA dataset (as it has the highest reported scores after ours in Table 1, and can serve as a very strong baseline). The annotation transfer methods used to construct this dataset rely on language-specific resources and a technically complex pipeline, making a replication of the transfer method on Finnish tedious at best. Instead, we create a Spanish translation of SQuAD using our MT service-based method. We then train QA models on these two Spanish datasets using the Spanish ALBERT-XXL model⁵, and com-

⁵<https://huggingface.co/dccuchile/albert-xxlarge-spanish>

pare their relative performance. The results of this comparison are reported in Table 3. When trained and tested on the same dataset, the result seen earlier for the Finnish dataset repeats, here with a 3.7pp EM and 5.5pp F1 improvement in favor of our method of translation and annotation transfer. In cross-dataset experiments, we see that training on our dataset always brings better F1 score, irrespective of which test set we use. The EM metric then has an opposite tendency, hinting at the two methods producing different entity boundaries, which are then learned by the QA models.

Train	Test	EM	F1
TAR	TAR	66.3	73.7
our	TAR	64.5	74.0
TAR	our	65.2	76.1
our	our	70.0	79.2

Table 3: Exact match (EM) and F1 scores between Spanish TAR method and Spanish DeepL method.

4.4 Dataset error analysis

Finally, we conducted a manual error analysis on a randomly selected subset of the FinSQuAD dataset, sampling 321 answerable questions from 51 passages in 17 different articles and inspected the resulting answer spans. We categorize the answers in 6 different categories:

Correct	The answer span corresponded to the English original flawlessly
Punctuation	The answer corresponded to the English original, except for a minor difference in punctuation
Over-extended	The answer was longer than in the English original
Under-extended	The answer was shorter than in the English original
Wrong	The answer did not correspond to the English original of reasons other than over/under-extension.
Missing	The question did not have an answer, the span failed to be transferred

The result of the error analysis in Table 4 show that full 87.2% of the answers are transferred fully correctly, and only 2.2% of the answers are lost, i.e. not transferred at all. The most common error,

accounting for nearly all errors in the dataset is over-extension, most typically by a single token.

	#	%
Correct	280	87.2
Punctuation	1	0.3
Over-extended	29	9.0
Under-extended	4	1.2
Wrong	0	0.0
Missing	7	2.2
Total	321	100.0

Table 4: Error analysis results of the translated FinSQuAD dataset.

5 Discussion and Conclusions

In this paper, we have demonstrated a practical method for annotation transfer through an affordable, high-quality machine translation service, relying on its ability to translate formatted text documents. We have applied this method to create a Finnish QA dataset with very little effort and negligible cost, resulting in a Finnish SQuAD2.0 translation with higher coverage and better overall model performance than what was previously available for Finnish. As a side product of our evaluation, we have also created an alternate Spanish SQuAD dataset of seemingly better quality than that previously available. We have shown, through comparison to other machine translated QA datasets, and more directly also through an English-Finnish-English backtranslation experiment, that the dataset is unlikely to result in substantially worse models than a (hypothetical) Finnish dataset created manually. The backtranslation experiment suggest the penalty for MT is about 5pp in terms of EM and 2.5pp in terms of F1.

We argue that the value of the approach is in allowing for a substantial expansion in the availability of numerous NLP tasks in a number of languages that currently lack the relevant native datasets. While it is clear that a high-quality dataset manually annotated in the target language is the best resource for training NLP models, it is clear that for many task-language pairs such a dataset will not be created for many years to come, if ever. In these cases, we argue that the method gives a practical, viable alternative which, thanks to its simplicity can be implemented with ease and applied quite broadly to produce datasets

for many tasks in many languages. The applicability of the method will naturally depend on the task, and likely to a degree the language at hand.

The method is naturally limited to the language pairs supported by the translation service used and may not be practical for very large datasets in the billion word range. It also relies on the availability of a suitable translation service with terms and conditions not restricting such application (as is the case at present). While such dependence is not ideal, it is nevertheless becoming somewhat the norm in NLP, where large, high-quality models and systems are increasingly exposed through a service, rather than distributed openly, which is understandable given their development and deployment costs.

Our code is available under an open source license, and can be used to generate QA datasets for other languages supported by the translation service. The Finnish dataset is on the HuggingFace dataset repository as `TurkuNLP/squad_v2-fi`, the code is on GitHub as `TurkuNLP/squad2-fi`, and the Finnish model is on the HuggingFace model repository as `TurkuNLP/bert-base-finnish-cased-squad2`.

6 Acknowledgments

We thank Jenna Kanerva for fruitful discussions at the beginning of the study. Computational resources were provided by CSC - IT Centre of Science, Finland. The research was supported by the Research Council of Finland funding.

7 Limitations

One limitation of this work is in relying on a particular property of an existing MT system, which also limits the applicability only to the languages supported by it. This is alleviated by the fact that DeepL supports 33 languages, allowing for a potentially very large number of datasets to be translated in the simple manner we outline. Further, since professional MT systems are primarily targeting translators and need to support formatting transfer to remain competitive, it is conceivable that a suitable MT system can be found also for other languages.

Another limitation is in relying on a closed, commercial system, which naturally negatively affects e.g. replicability. However, the system only needs to be used once, when creating the new dataset, and after that the dataset is available

openly and can be evaluated in a transparent manner. The closed nature of the MT system thus does not fully transfer onto the dataset. We note that our use of a closed MT system is fully comparable to the current wide-spread practice in which NLP datasets are created using closed, commercial LLMs such as OpenAI’s ChatGPT.

References

- Negin Abadani, Jamshid Mozafari, Afsaneh Fatemi, Mohammad Ali Nematbakhsh, and Arefeh Kazemi. 2021. ParSQuAD: Machine Translated SQuAD dataset for Persian Question Answering. In *2021 7th International Conference on Web Research (ICWR)*, pages 163–168, Tehran, Iran. IEEE.
- Elisa Bassignana, Filip Ginter, Sampo Pyysalo, Rob Goot, and Barbara Plank. 2023. Multi-CrossRE a multi-lingual multi-domain dataset for relation extraction. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 80–85, Tórshavn, Faroe Islands. University of Tartu Library.
- Casimiro Pio Carrino, Marta R Costa-jussa, and Jose A R Fonollosa. 2019. Automatic Spanish Translation of the SQuAD Dataset for Multilingual Question Answering. ArXiv:1912.05200v2 [cs].
- Yiming Cui, Ting Liu, Wanxiang Che, Li Xiao, Zhipeng Chen, Wentao Ma, Shijin Wang, and Guoping Hu. 2019. A Span-Extraction Dataset for Chinese Machine Reading Comprehension. 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 5882–5888. ArXiv:1810.07366 [cs].
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. ArXiv:1810.04805 [cs].
- Martin d’Hoffschmidt, Wacim Belblidia, Tom Brendlé, Quentin Heinrich, and Maxime Vidal. 2020. FQuAD: French Question Answering Dataset. ArXiv:2002.06071 [cs].
- Ilmari Kylliäinen and Roman Yangarber. 2023. Question Answering and Question Generation for Finnish. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 529–540, Tórshavn, Faroe Islands. University of Tartu Library.
- Ilmari Kylliäinen. 2022. Neural Factoid Question Answering and Question Generation for Finnish. Master’s thesis, University of Helsinki.

- 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. ArXiv:1909.11942 [cs].
- Seungyoung Lim, Myungji Kim, and Jooyoul Lee. 2019. KorQuAD1.0: Korean QA Dataset for Machine Reading Comprehension. ArXiv:1909.07005 [cs].
- Ofri Masad, Kfir Bar, and Amir Cohen. 2023. Automatic Translation of Span-Prediction Datasets. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 160–173, Nusa Dua, Bali. Association for Computational Linguistics.
- Hussein Mozannar, Karl El Hajal, Elie Maamary, and Hazem Hajj. 2019. Neural Arabic Question Answering. ArXiv:1906.05394 [cs].
- Koichi Nagatsuka, Clifford Broni-Bediako, and Masayasu Atsumi. 2021. Pre-training a BERT with Curriculum Learning by Increasing Block-Size of Input Text. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 989–996, Held Online. INCOMA Ltd.
- Susumu Okazawa. 2021. Swedish translation of SQuAD2.0. https://github.com/susumu2357/SQuAD_v2_sv Last accessed 20 June 2023.
- Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know What You Don’t Know: Unanswerable Questions for SQuAD. Number: arXiv:1806.03822 arXiv:1806.03822 [cs].
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. Number: arXiv:1606.05250 arXiv:1606.05250 [cs].
- Tom M Seinen, Jan A Kors, Erik M van Mulligen, and Peter R Rijnbeek. 2024. Annotation-preserving machine translation of English corpora to validate Dutch clinical concept extraction tools. *Journal of the American Medical Informatics Association*, 31(8):1725–1734.
- Anastassia Shaitarova, Anne Göhring, and Martin Volk. 2023. Machine vs. Human: Exploring Syntax and Lexicon in German Translations, with a Spotlight on Anglicisms. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 215–227, Tórshavn, Faroe Islands. University of Tartu Library.
- web source. 2021a. bert-base-spanish-wwm-cased-finetuned-sqac-finetuned-squad2-es. <https://huggingface.co/MMG/bert-base-spanish-wwm-cased-finetuned-sqac-finetuned-squad2-es> Last accessed 20 June 2023.
- web source. 2021b. Indobert-qa. <https://huggingface.co/Rifky/Indobert-QA> Last accessed 20 June 2023.
- web source. 2021c. Indonesian squad. https://github.com/Wikidepia/indonesian_datasets/tree/master/question-answering/squad Last accessed 20 June 2023.
- Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: BERT for Finnish. ArXiv:1912.07076 [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.
- Robert Östling and Jörg Tiedemann. 2016. Efficient Word Alignment with Markov Chain Monte Carlo. *The Prague Bulletin of Mathematical Linguistics*, 106(1):125–146.

How to Tune a Multilingual Encoder Model for Germanic Languages: A Study of PEFT, Full Fine-Tuning, and Language Adapters

Romina Oji and Jenny Kunz

Dept. of Computer and Information Science

Linköping University

romina.oji@liu.se and jenny.kunz@liu.se

Abstract

This paper investigates the optimal use of the multilingual encoder model mDeBERTa for tasks in three Germanic languages – German, Swedish, and Icelandic – representing varying levels of presence and likely data quality in mDeBERTa’s pre-training data. We compare full fine-tuning with the parameter-efficient fine-tuning (PEFT) methods LoRA and Pfeiffer bottleneck adapters, finding that PEFT is more effective for the higher-resource language, German. However, results for Swedish and Icelandic are less consistent. We also observe differences between tasks: While PEFT tends to work better for question answering, full fine-tuning is preferable for named entity recognition. Inspired by previous research on modular approaches that combine task and language adapters, we evaluate the impact of adding PEFT modules trained on unstructured text, finding that this approach is not beneficial.

1 Introduction

Massively multilingual encoder models like mBERT (Devlin et al., 2019), XLM-R (Conneau et al., 2020) and mDeBERTa (He et al., 2021b) are a workhorse for NLP in many lower-resource languages. However, due to interference between languages (Conneau et al., 2020; Chang et al., 2023), these models can fall short of reaching their full potential for individual target languages: Monolingual models (Virtanen et al., 2019; Snæbjarnarson et al., 2022) and models with dedicated language modules (Pfeiffer et al., 2022; Blevins et al., 2024) frequently outperform them, raising the question for the best setups for different languages.

Parameter-efficient fine-tuning (PEFT) methods, such as bottleneck adapters (Houlsby et al., 2019),

LoRA (Hu et al., 2022), and prefix tuning (Li and Liang, 2021), have emerged as an alternative to full fine-tuning of pre-trained language models. These methods preserve the model’s representations and can lead to better generalisation (He et al., 2021c). This is especially relevant for multilingual models, trained on diverse data, of which the target language only constitutes a small fraction. Fully fine-tuning them on task-specific data risks overwriting some of the multilingual capabilities.

Language adapters – PEFT modules trained on unstructured text independently from task fine-tuning – have shown promise in cross-lingual transfer (Pfeiffer et al., 2020; Vidoni et al., 2020). We explore whether language adaptation modules are beneficial even in scenarios where cross-lingual transfer is *not* required, i.e., where we have in-language fine-tuning data. In addition, we use not only bottleneck (Pfeiffer) adapters but also LoRA (Hu et al., 2022), a method that has become popular for LLMs as its parameters can be merged with the model parameters, adding no inference overhead.

In this paper, we investigate strategies for adapting a multilingual encoder model to task data in three languages: German, Swedish, and Icelandic. For this, we use multilingual DeBERTa (He et al., 2021b), which is currently the best-performing model according to the ScandEval (Nielsen et al., 2024) leaderboard for Icelandic,¹ the lowest-resourced and thus the most challenging of the three languages.

Our findings indicate that the effectiveness of full fine-tuning versus PEFT varies by language. For German, a PEFT method consistently delivers the best results, although sometimes with marginal gains. For Swedish and Icelandic, the performance is task-dependent: PEFT is more beneficial for extractive question-answering (QA), while full fine-tuning works better for named entity recognition

¹<https://scandeval.com/icelandic-nlu/>, as of 21/10/2024.

(NER). We hypothesise that in languages with quantitatively more limited or lower-quality representation in the pre-training data, there is less value in preserving the pre-existing representations and more value in increasing the learning capacity. In contrast, for higher-resource languages, capabilities from the pre-training phase are more impactful. Similarly, for extractive QA, pre-existing skills weigh higher, while the highly specific nature of NER benefits from full fine-tuning.

Language adapters do not provide consistent improvements in any of the tasks or languages tested. As the adaptation data we use has likely been used for pre-training the multilingual DeBERTa model, we conclude that the utilisation of this data at pre-training time has already been effective enough. Further adaptation, or specialisation, with this same data does not have a clear benefit.

2 Related Work

PEFT methods not only reduce the number of trainable parameters and, consequently, memory usage in comparison to full fine-tuning, but there is also evidence suggesting that they provide better regularisation and help preserve pre-existing model capabilities. For example, He et al. (2021c) demonstrate that adapter-based fine-tuning outperforms full fine-tuning in cross-lingual transfer setups, likely by avoiding overfitting on the source language. Similarly, prefix tuning, another PEFT method, has been shown to surpass full fine-tuning in extrapolation scenarios (Li and Liang, 2021).

Other works have shown the effectiveness of bottleneck-style adapters in cross-lingual transfer as post-hoc trained language modules in encoder models. Pfeiffer et al. (2020) show that bottleneck language adapters in the Pfeiffer architecture improve performance in NER, commonsense classification, and extractive QA. Even Vidoni et al. (2020) report that language adapters are effective. Other research indicates that language adapters can aid in transferring knowledge to dialectal variants (Vamvas et al., 2024) and that sharing adapters across related languages can be beneficial (Faisal and Anastasopoulos, 2022; Chronopoulou et al., 2023). However, the success of language adapters may be task-specific and difficult to measure accurately when using machine-translated evaluation data (Kunz and Holmström, 2024). And notably, none of the works used multilingual DeBERTa models, which may explain divergences in results.

3 Experimental Setup

3.1 Model

We use the multilingual DeBERTa v3 model² as the base for our experiments. This model contains about 86 million parameters in its backbone, and the embedding layer, with a vocabulary of 250,000 tokens, adds another 190 million parameters, bringing the total to around 278 million parameters (He et al., 2021a). It was trained on 2.5 TB of the CC100 multilingual dataset (Wenzek et al., 2020; Conneau et al., 2020), which includes 100 languages, including Icelandic, Swedish, and German.

3.2 Tasks

We evaluate the fine-tuning and language adaptation methods on three tasks: extractive question answering (QA), named entity recognition (NER), and linguistic acceptability classification. This selection is inspired by coverage in the ScandEval benchmark (Nielsen et al., 2024) for all three languages while having structurally different tasks.

QA: For Icelandic, we use the *Natural Questions in Icelandic (NQI)* dataset, which features questions from Icelandic texts written by Icelandic speakers. (Snæbjarnarson and Einarsson, 2022). For Swedish, we use the Swedish portion of ScandiQA, which was manually translated from English (Nielsen, 2023). For German, we use the human-labeled GermanQuAD dataset, which is natively German. (Möller et al., 2021).

NER: For Icelandic, we use the MIM-GOLD-NER dataset (Ingólfssdóttir et al., 2020), for Swedish, we use the Stockholm-Umeå Corpus (Kurtz and Öhman, 2022) and for German, we use GermanEval 2014 (Benikova et al., 2014).

Linguistic Acceptability: For all three languages, we use the respective portion of ScaLA (Nielsen, 2023), a binary classification dataset that judges the linguistic acceptability of sentences. Sentences are tagged as either grammatically correct or incorrect. This dataset is synthetically created by introducing corruptions based on the dependency trees of the sentences.

3.3 PEFT Methods

We use two different PEFT methods. **Pfeiffer adapters** (Pfeiffer et al., 2021, 2020) are a vari-

²loaded from <https://huggingface.co/microsoft/mdebarta-v3-base>

ation of bottleneck adapters (Houlsby et al., 2019), that is, small feed-forward layers that reduce the dimensionality of the input, process it, and then expand it back to the original size. They are inserted between the layers of the transformer model, and are the only parameters that are trained. **LoRA** (Hu et al., 2022) approximates the original weight updates as a low-rank decomposition by learning two low-rank matrices. Instead of updating the full set of model parameters, LoRA inserts trainable low-rank matrices into the self-attention of each layer of the model and updates only those.

3.4 PEFT Training

In the first step, we fine-tune individual *language adapters* for Icelandic, Swedish, and German, using the masked language modeling objective. We use 250,000 samples from the CC100 dataset and train a LoRA and a Pfeiffer language adapter for each language. Our language adapters are available at <https://huggingface.co/rominaoji>.

Task adapters are fine-tuned on target-language task data with the datasets described in Section 3.2.

For all adapters, we set the LoRA rank to 8 and the α to 16, while for the Pfeiffer method, the reduction factor is set to 16. For the implementation, we use the *adapters* library (Poth et al., 2023).

3.5 Setups

To find the optimal method to use mDeBERTa for the three languages, we fine-tune it using three setups: (1) **Full fine-tuning**, (2) tuning using only **task adapters**, and (3) using a **combination of language and task adapters** as in the MAD-X framework. In each setup, models are fine-tuned over five epochs.

As PEFT models require higher learning rates than full fine-tuning due to their lower number of trainable parameters, we determine a suitable rate for each setup by testing learning rates from $1\text{e-}4$ to $9\text{e-}4$ for PEFT and from $1\text{e-}5$ to $9\text{e-}5$ for full fine-tuning. This resulted in a learning rate of $3\text{e-}4$ for both the language and task adaptation methods and $2\text{e-}5$ for full fine-tuning. All experiments use a linear scheduler paired with the AdamW optimiser (Loshchilov, 2017). The code is available at <https://github.com/rominaoji/german-language-adapter>.

3.6 Evaluation

For the sake of simplicity, we only present F1 scores as the evaluation metric for all three tasks in this paper. While we have collected results on more metrics, we did not observe differences in the trends. The results are the mean of a five-fold cross-validation, with standard deviation.

4 Results and Discussion

All results are presented in Table 1. We discuss the effects of different task fine-tuning strategies on different languages and tasks (§4.1) and finally the effect of language adapters (§4.2).

4.1 Full Fine-Tuning Versus PEFT

Tasks: For the extractive QA tasks, we observe that PEFT methods generally outperform full fine-tuning. In German, there is a notable gap between full fine-tuning and both PEFT methods, with LoRA yielding the best results. For Icelandic, Pfeiffer adapters outperform both full fine-tuning and LoRA. For Swedish, the differences between setups are minimal. We hypothesise that for this task, the model benefits from the pre-trained representations and does not require the highest possible learning capacity to identify relevant text spans in these tasks.

In contrast, full fine-tuning is the best approach for NER tasks, outperforming the highest-performing PEFT method in Icelandic and Swedish, and performing on par with Pfeiffer adapters in German. This suggests that for this word-level task, a larger learning capacity is more crucial than preserving fine-grained capabilities from pre-training.

For ScaLA, the results are mixed. Full fine-tuning yields slightly higher scores for Icelandic, while Pfeiffer adapters perform marginally better for Swedish and German. Interpreting the performance on this task is challenging, as the dataset contains some corrupted instances that may be detectable with simple pattern-matching, while others require more fine-grained linguistic knowledge.

Languages: For German, Pfeiffer adapters consistently outperform full fine-tuning in QA and ScaLA tasks, and are either on par or slightly better for NER. LoRA performs best for QA but yields lower scores in the other two tasks. This suggests that German benefits from keeping the base model intact, likely due to its relatively large representation in the pre-training dataset.

TA	LA	QA			NER			ScaLA		
		Icelandic	Swedish	German	Icelandic	Swedish	German	Icelandic	Swedish	German
Full FT	-	57.52 ± 1.50	35.08 ± 0.77	73.56 ± 0.78	92.35 ± 0.31	87.47 ± 0.41	84.83 ± 0.33	76.17 ± 1.32	84.23 ± 1.07	83.90 ± 0.82
Pfeiffer	-	59.31 ± 1.14	35.15 ± 1.00	75.84 ± 0.92	91.37 ± 0.23	86.64 ± 0.51	85.14 ± 0.22	76.35 ± 0.56	84.94 ± 1.04	84.53 ± 0.64
LoRA	-	57.65 ± 2.11	34.76 ± 0.82	77.17 ± 0.74	89.69 ± 0.49	85.16 ± 0.32	84.12 ± 0.31	70.64 ± 2.78	82.32 ± 1.80	78.75 ± 2.34
Pfeiffer	Pfeiffer	60.02 ± 1.46	35.07 ± 0.78	76.66 ± 0.55	91.41 ± 0.23	86.72 ± 0.28	84.77 ± 0.38	75.38 ± 1.21	84.68 ± 0.76	84.02 ± 0.64
LoRA	Pfeiffer	57.44 ± 1.61	34.77 ± 0.60	77.13 ± 0.28	89.95 ± 0.50	85.11 ± 0.30	84.05 ± 0.35	71.31 ± 2.30	82.58 ± 2.01	78.85 ± 2.28
Pfeiffer	LoRA	59.24 ± 0.60	34.97 ± 0.82	76.31 ± 0.63	91.49 ± 0.29	86.50 ± 0.60	85.08 ± 0.22	75.06 ± 1.41	84.98 ± 1.23	83.86 ± 0.30
LoRA	LoRA	57.05 ± 1.74	34.40 ± 0.40	77.02 ± 0.35	89.64 ± 0.43	85.11 ± 0.30	84.08 ± 0.20	71.01 ± 3.00	82.97 ± 1.78	78.94 ± 2.37

Table 1: Mean F1 scores over five runs with standard deviation for all tasks and languages. The first column specifies the task adaptation method (TA), and the second one the language adaptation method (LA). The respectively highest score is highlighted in bold blue italics, the runner-up in bold black.

For Swedish, the performance of full fine-tuning and Pfeiffer adapters is similar across all three tasks, showing little variation.

For Icelandic, Pfeiffer adapters achieve higher scores in QA, while full fine-tuning performs better for NER. For ScaLA, both approaches produce comparable results. Icelandic’s low representation in the CC-100 dataset used to train mDeBERTa might explain why it benefits less from the model’s pre-training than German. While Swedish even has a slightly larger quantitative representation than German in open CC100 dumps,³ it is unclear if the quality of the Swedish data matches that of the German data. For lesser-resourced languages, the quality of common-crawl corpora is often lower (Kreutzer et al., 2022; Artetxe et al., 2022), which may diminish the usefulness of pre-training for Swedish compared to German. Swedish has 13M speakers (10M L1),⁴ whereas German has 175M speakers (95M L1),⁵ which probably makes German higher-resource than Swedish, and may lead to higher-quality representation of German.

PEFT Methods: Except for German QA, Pfeiffer adapters outperform LoRA across all tasks. This may be due to architectural differences, though it is worth noting that Pfeiffer adapters in our setup have a higher learning capacity, with 896K trainable parameters compared to LoRA’s 296K. Additionally, LoRA may require more extensive hyperparameter tuning than Pfeiffer adapters, as previous studies have shown its behavior to be unstable under certain conditions (Liu et al., 2024). A deeper exploration of how to improve LoRA’s adaptation is left for future work.

³See e.g. <https://huggingface.co/datasets/statmt/cc100> as of 23/10/2024.

⁴https://en.wikipedia.org/wiki/Swedish_language as of 23/10/2024.

⁵https://en.wikipedia.org/wiki/German_language as of 23/10/2024.

4.2 Language Adaptation

Language adapters do not provide any significant benefits. When using Pfeiffer task adapters, performance remains similar whether language adapters are included or not. The only exception is Icelandic QA, where the combination of a Pfeiffer language adapter and a Pfeiffer task adapter achieves a slightly higher score compared to the best setup without language adapters. However, the difference is small and possibly due to result variability, as it falls within a standard deviation.

With LoRA task adapters, language adaptation methods sometimes result in a noticeable performance drop, suggesting potential interference. While prior work, such as Pfeiffer et al. (2020), reported improvements in similar tasks, their study focused on cross-lingual transfer, where no task data from the target language was available. In contrast, our setups use task data from the target language, and all the languages are present in the model’s pre-training data. In addition, we use mDeBERTa-v3, which reportedly performs better for the languages in question than the XLM-R (Conneau et al., 2020) and multilingual BERT (Devlin et al., 2019) models that most other papers including Pfeiffer et al. (2020) use. These factors likely contribute to the fact that language adapters are unnecessary in our setup.

5 Conclusion

We compared the performance of the multilingual encoder model mDeBERTa across three task adaptation setups: full fine-tuning, bottleneck (Pfeiffer) adapters, and LoRA. Based on our evaluations across three tasks and three languages, we found that the choice of the best method is both task- and language-dependent. Specifically, extractive QA tasks benefit from PEFT methods, while NER gets better results with full fine-tuning. For Ger-

man, a higher-resourced language, PEFT consistently achieves higher scores. This suggests that the model benefits from fine-grained information learned during pre-training if coverage and (or) quality of the language data in the pre-training corpus are sufficiently high. In contrast, for lower-resourced languages, the increased learning capacity of full fine-tuning proves more advantageous.

We also tested language adaptation with Pfeiffer adapters and LoRA on unstructured text data before task adaptation. However, language adapters did not show any benefit. Access to target-language task data appears to dispense with the need for them, at least in our experiments where all languages are included in the pre-training data.

In future work, we aim to further explore the conditions under which PEFT methods versus full fine-tuning are most effective. We plan to investigate additional PEFT methods and tasks and optimise the LoRA setup, which may not have reached its full potential in our experiments.

Acknowledgments

We thank our colleagues Kevin Glocker, Kättriin Kukk, Julian Schlenker, Marcel Bollmann and Noah-Manuel Michael for valuable discussions at all stages of this project and feedback on earlier drafts, and the anonymous reviewers for their constructive feedback and insightful suggestions.

This research was supported by TrustLLM funded by Horizon Europe GA 101135671 and the National Graduate School of Computer Science in Sweden (CUGS). It was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. The computations were enabled by the Berzelius resource provided by the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre.

References

Mikel Artetxe, Itziar Aldabe, Rodrigo Agerri, Olatz Perez-de Viñaspre, and Aitor Soroa. 2022. Does corpus quality really matter for low-resource languages? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7383–7390, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014. Nosta-d named entity annotation for german: Guidelines and dataset. In *Proceedings of the Ninth International Conference on Language Resources*

and Evaluation (LREC’14), pages 2524–2531, Reykjavik, Iceland. European Language Resources Association (ELRA).

Terra Blevins, Tomasz Limisiewicz, Suchin Gururangan, Margaret Li, Hila Gonen, Noah A. Smith, and Luke Zettlemoyer. 2024. Breaking the curse of multilinguality with cross-lingual expert language models.

Tyler A. Chang, Catherine Arnett, Zhuowen Tu, and Benjamin K. Bergen. 2023. When is multilinguality a curse? language modeling for 250 high- and low-resource languages.

Alexandra Chronopoulou, Dario Stojanovski, and Alexander Fraser. 2023. Language-family adapters for low-resource multilingual neural machine translation. In *Proceedings of the Sixth Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2023)*, pages 59–72, Dubrovnik, Croatia. 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.

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.

Fahim Faisal and Antonios Anastasopoulos. 2022. Phylogeny-inspired adaptation of multilingual models to new languages. 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 434–452, Online only. Association for Computational Linguistics.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021a. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021b. Deberta: Decoding-enhanced bert with disentangled attention.

Ruidan He, Linlin Liu, Hai Ye, Qingyu Tan, Bosheng Ding, Liying Cheng, Jiawei Low, Lidong Bing, and Luo Si. 2021c. On the effectiveness of adapter-based

- tuning for pretrained language model 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 2208–2222, Online. Association for Computational Linguistics.
- Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.
- 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*.
- Svanhvít Lilja Ingólfssdóttir, Ásmundur Alma Guðjónsson, and Hrafn Loftsson. 2020. MIM-GOLD-NER – named entity recognition corpus (20.06). CLARIN-IS.
- Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmongkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, Iroko Orife, Kelechi Ogueji, Andre Niyongabo Rubungo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. 2022. Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.
- Jenny Kunz and Oskar Holmström. 2024. The impact of language adapters in cross-lingual transfer for NLU. In *Proceedings of the 1st Workshop on Modular and Open Multilingual NLP (MOOMIN 2024)*, pages 24–43, St Julians, Malta. Association for Computational Linguistics.
- Robin Kurtz and Joey Öhman. 2022. The kblab blog: Sucx 3.0 - ner.
- Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. 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 4582–4597, Online. Association for Computational Linguistics.
- Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. 2024. Dora: Weight-decomposed low-rank adaptation.
- I Loshchilov. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.
- Timo Möller, Julian Risch, and Malte Pietsch. 2021. Germanquad and germandpr: Improving non-english question answering and passage retrieval. *arXiv preprint arXiv:2104.12741*.
- Dan Saattrup Nielsen. 2023. Scandeval: A benchmark for scandinavian natural language processing. *arXiv preprint arXiv:2304.00906*.
- Dan Saattrup Nielsen, Kenneth Enevoldsen, and Peter Schneider-Kamp. 2024. Encoder vs decoder: Comparative analysis of encoder and decoder language models on multilingual nlu tasks.
- Jonas Pfeiffer, Naman Goyal, Xi Lin, Xian Li, James Cross, Sebastian Riedel, and Mikel Artetxe. 2022. Lifting the curse of multilinguality by pre-training modular transformers. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3479–3495, Seattle, United States. Association for Computational Linguistics.
- Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, Online. Association for Computational Linguistics.
- Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.
- Clifton Poth, Hannah Sterz, Indraneil Paul, Sukannya Purkayastha, Leon Engländer, Timo Imhof, Ivan Vulić, Sebastian Ruder, Iryna Gurevych, and Jonas Pfeiffer. 2023. Adapters: A unified library for parameter-efficient and modular transfer learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 149–160, Singapore. Association for Computational Linguistics.

- Vésteinn Snæbjarnarson and Hafsteinn Einarsson. 2022. Natural questions in Icelandic. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4488–4496, Marseille, France. European Language Resources Association.
- Vésteinn Snæbjarnarson, Haukur Barri Símonarson, Pétur Orri Ragnarsson, Svanhvít Lilja Ingólfssdóttir, Haukur Jónsson, Vilhjálmur Thorsteinsson, and Hafsteinn Einarsson. 2022. A warm start and a clean crawled corpus - a recipe for good language models. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4356–4366, Marseille, France. European Language Resources Association.
- Jannis Vamvas, Noëmi Aeppli, and Rico Sennrich. 2024. Modular adaptation of multilingual encoders to written Swiss German dialect. In *Proceedings of the 1st Workshop on Modular and Open Multilingual NLP (MOOMIN 2024)*, pages 16–23, St Julians, Malta. Association for Computational Linguistics.
- Marko Vidoni, Ivan Vulić, and Goran Glavaš. 2020. Orthogonal language and task adapters in zero-shot cross-lingual transfer.
- Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: Bert for finnish.
- Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

Match ‘em: Multi-Tiered Alignment for Error Analysis in ASR

Phoebe Parsons¹

Knut Kvale²

Torbjørn Svendsen¹

Giampiero Salvi¹

¹Department of Electronic Systems, NTNU, Trondheim, Norway

²Telenor Research and Innovation, Oslo, Norway

{phoebe.parsons, torbjorn.svendsen, giampiero.salvi}@ntnu.no, knut.kvale@telenor.com

Abstract

We introduce “Match ‘em”: a new framework for aligning output from automatic speech recognition (ASR) with reference transcriptions. This allows a more detailed analysis of errors produced by end-to-end ASR systems compared to word error rate (WER). Match ‘em performs the alignment on both the word and character level; each relying on information from the other to provide the most meaningful global alignment. At the character level, we define a speech production motivated character similarity metric. At the word level, we rely on character similarities to define word similarity and, additionally, we reconcile compounding (insertion or deletion of spaces). We evaluated Match ‘em on transcripts of three European languages produced by wav2vec2 and Whisper. We show that Match ‘em results in more similar word substitution pairs and that compound reconciling can capture a broad range of spacing errors. We believe Match ‘em to be a valuable tool for ASR error analysis across many languages.

1 Introduction

Metrics like word error rate (WER) provide a simple, automated way of understanding how well an automatic speech recognition (ASR) system is performing. However, this simplicity fails to capture the nuance regarding the severity of transcription errors, both in terms of spellings and semantics. Efforts have been made to improve WER. These include adding new metrics around information lost by mistranscriptions (Morris et al., 2004) and weighting keywords more heavily in WER (Nanjo and Kawahara, 2005). Attempts to optimize the

alignment between transcriptions have utilized articulatory features (Cucchiarini, 1996) as well as semantic distances (Roy, 2021). Additionally, new metrics such as SemDist (Kim et al., 2021) and Aligned Semantic Distance (Rugayan et al., 2022) have been developed to utilize the embedding vector space, instead of aligning the words themselves, to calculate the severity of errors. However, as all these metrics only aim to summarize the quality or utility of an ASR output, they do not provide details on the types or severity of commonly made errors.

Understanding the types of errors that ASR systems make has been of interest for many years. The goals are both understanding how wrong a transcription really is, as well as identifying specific areas for improvement. In Goldwater et al. (2010), the authors create individual word error rate to determine which words are frequently missed and which factors account for misrecognitions. In Vasilescu et al. (2012), the authors compare the ability of humans and automatic transcriptions to disambiguate homophonic or near-homophonic words that are frequently missed by ASR. Words that are frequently missed in conversational speech for Dutch, English, and German are analyzed in Lopez et al. (2022). The authors in Wirth and Peinl (2022); Salimbajevs and Strigins (2015) manually classify ASR errors for both their severity and type to understand how ASR is performing on German and Latvian speech, respectively.

Despite the benefits of metrics and error analysis, there are several factors that can be limiting to these tools. For semantic metrics, knowledge of the language (semantic embeddings, word importance) is crucial. However, access to such resources is not readily available for certain languages. Similarly, analysis of ASR errors is often reliant on manual efforts to label the errors made, thus limited by the amount of human hours

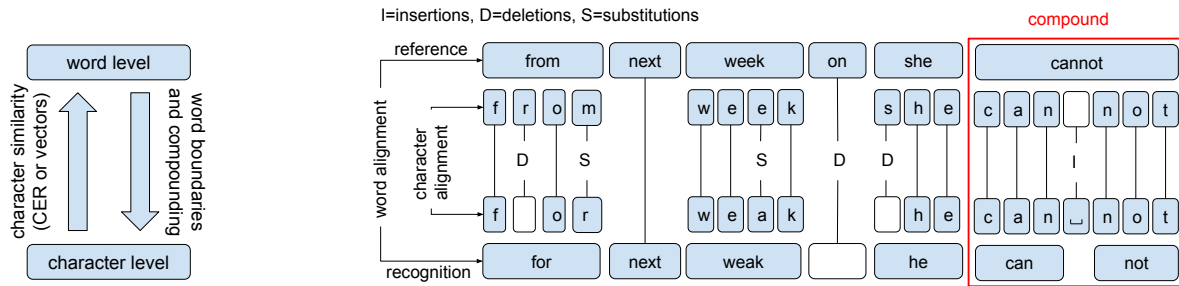


Figure 1: Left: interaction between word and character level in Match'em. Right: An example alignment showing the multi-tiered (word and character) approach as well as compounding.

available to contribute to the task. Lastly, many existing metric and evaluation paradigms are designed with the assumption that words operate as atomic units — an assumption challenged by end-to-end ASR systems where output is generated at the character or sub-word level.

In this paper, we propose a new framework, that we call Match 'em, for aligning ASR generated and reference text that operates both at the word and character level. The goal is to provide a better match between words and characters thus allowing for a detailed analysis of the common mistakes produced by ASR systems. Additionally, this method opens the possibility to use foundation ASR models trained on massive amounts of data to study phenomena related to variability in speech production by analyzing the ASR errors in detail; such phenomena include dialectal variation or pronunciation variation in second language learners or in speakers with speech sound disorder.

The contributions of the paper can be summarized as:

- We introduce a new framework for ASR output and reference alignment that operates on the word and character level. Each level influences the other level with the goal of obtaining an optimal global alignment.
- We introduce a character dissimilarity metric based on speech production to guide the within-word character alignments.
- At the word level, we define a word dissimilarity metric that inherits similarities from the character level. We also implement an algorithm for reconciling compounding (insertion or deletion of spaces)
- We evaluate the method on transcripts of three European languages obtained by two

state-of-the-art ASR models (wave2vec2, and Whisper), showing that Match 'em produces more meaningful alignments both in terms of word similarities and character similarities.

- We make all the code available.

2 The Match 'em framework

The standard Levenshtein alignment considers three edit operations (insertion, deletion, and substitution) when transforming the hypothesis text into the reference text (Levenshtein, 1965). The edit costs (that is, the penalty for any of the three edits) are also fixed before alignment occurs. This method is traditionally used to separately compute either word error rate (WER) at the word level or character error rate (CER) at the character level. The Match 'em framework we propose operates both at the word and the character level *simultaneously*. The alignment at each level is influenced by information coming from the other, as illustrated by Figure 1. In the figure, we can see that words that are spelled similarly are aligned, the characters within the words are aligned, and the breaking up of a compound word is accounted for. Details on how each of these components was achieved follows in the subsections below.

2.1 Character- and Word-Level Metrics

The first step in defining the Match 'em algorithm is to define metrics both at the character and word level. At the character level, we introduce a dissimilarity metric based on speech production, similar to the method in (Cucchiarini, 1996). We define a set of vectors of articulatory features for each letter in the target language's alphabet. To accommodate the different parameters by which vowels and consonants are defined, separate vec-

Vowels				Consonants					
value	height	front/back	rounding	value	voice	class	nasal	place	lip rounding
0	high	back	false	0	false	stop	false	bilabial	false
1	mid	mid	true	1	true	affricate	true	labio-dental	true
2	low	front		2		trill		alveolar	
				3		fricative		retroflex	
				4		approximate		palatal	
				5				velar	
				6				uvular	
				7				glottal	

Table 1: Articulatory features used to define the character-level metric for vowels (left) and consonants (right). For example, the vector [0, 2, 1] would be interpreted to mean a high, front, rounded vowel /y/, whereas the vector [1, 0, 1, 2, 0] would represent a voiced, nasal, alveolar stop /n/. The currently defined vector system does not account for every word sound and would need to be adjusted or expanded as new languages were used.

tor definitions are used for each class. Examples of the vector spaces are provided in Table 1. Each character is then assigned to one or more vectors depending on its typical pronunciation(s). Doing this, the method account for characters that might be commonly realized as two distinct phones (e.g., the Norwegian "r" is a dialect marker and can be realized as either an alveolar tap or uvular trill (Kvale and Foldvik, 1992)).

The distance (dissimilarity) between two characters (either vowel-to-vowel or consonant-to-consonant) is computed as the normalized Euclidean distance between the corresponding vectors. Comparing vowels and consonants in this articulatory space is not meaningful. Instead, the cost is set at 1.0 for most vowel-consonant substitutions, the same cost as a substitution of two completely different characters. With vowel-approximants, the cost is lowered to 0.9 to allow for the gestural and perceptual similarities. This value was chosen through experimentation and visual inspection of the resulting alignments. The cost is also set at 1.0 for any character-punctuation substitution. If multiple definitions character vectors are provided (e.g. in accounting for two realizations of "r"), the vector with the lowest resulting dissimilarity is used.

As these vectors' purpose is merely to support a character distance score, not to offer linguist truth, there are known simplifications and omissions in the vector definitions. As an example, di- or tri-graphs are not captured in the letter vectors.

In practice, we find that defining these character vectors to be straight-forward for languages

				Standard costs	Match 'em costs
cats	run	very	quickly		
	cat	runs	quick	4	3.286
cat		runs	quick	4	2.555
cat	runs		quick	4	1.869
cat	runs	quick		4	2.583

Table 2: Potential alignments for the two phrases *cats run very quickly* and *cat runs quick*. The cumulative costs for each alignment is given for the standard and Match 'em approaches.

with available orthographic to phonetic mappings. Even for languages which the authors were unfamiliar, vector definition was quick.

At the word level, the dissimilarity between two words is computed by performing an alignment between the within-word characters of the two words in question (see Figure 1 (right) for an example). This alignment is guided either by the character dissimilarity defined previously, or by the simpler, character-naïve CER. This dissimilarity is then used as the substitution cost when aligning words. Insertion and deletion costs at the word level are left at 1.0.

2.2 Multi-tier Alignment

The Match 'em alignment makes use of the dissimilarity metrics defined in Section 2.1 to perform multi-tier alignment at the word and character level. Both levels use dynamic programming similarly to the Levenshtein method. However, at the word level, character-based word dissimilarity is used as cost for substitutions. Similarly, at the character level articulatory character dissimi-

	edit costs						cumulative costs				
		cats	run	very	quickly			cats	run	very	quickly
cat runs quick	0	1 ←	1 ←	1 ←	1 ←	cat runs quick	0	1 ←	2 ←	3 ←	4 ←
	1 ↑	1 ↖	1 ↖ ←	1 ↖ ←	1 ↖ ←		1 ↑	1 ↖	2 ↖ ←	3 ↖ ←	4 ↖ ←
	1 ↑	1 ↖ ↑	1 ↖	1 ↖ ←	1 ↖ ←		2 ↑	2 ↖ ↑	2 ↖	3 ↖ ←	4 ↖ ←
	1 ↑	1 ↖ ↑	1 ↖ ↑	1 ↖	1 ↖ ←		3 ↑	3 ↖ ↑	3 ↖ ↑	3 ↖	4 ↖ ←

Table 3: Standard approach: step-by-step edit costs (left) and cumulative costs (right) for aligning the two phrases *cats run very quickly* and *cat runs quick* using the standard approach. Backtrace arrows indicate from which cell the cost is computed.

	edit costs						cumulative costs				
		cats	run	very	quickly			cats	run	very	quickly
cat runs quick	0	1 ←	1 ←	1 ←	1 ←	cat runs quick	0	1 ←	2 ←	3 ←	4 ←
	1 ↑	1/4 ↖	1 ←	1 ←	1 ←		1 ↑	0.25 ↖	1.25 ←	2.25 ←	3.25 ←
	1 ↑	1 ↑	1/3 ↖	1 ←	1 ←		2 ↑	1.25 ↑	0.583 ↖	1.583 ←	2.583 ←
	1 ↑	1 ↑	1 ↑	1 ↖	2/7 ↖		3 ↑	2.25 ↑	1.583 ↑	1.583 ↖	1.869 ↖

Table 4: Match ‘em approach: step-by-step edit costs (left) and cumulative costs (right) for aligning the two phrases *cats run very quickly* and *cat runs quick* using the Match ‘em approach. Backtracing arrows indicate from which cell the cost is computed.

larities are used to align characters within words.

As a demonstration of the benefit of Match ‘em, let us consider the examples provided in Table 2. Here we have four different potential alignments between the reference text *cats run very quickly* and the hypothesis text *cat runs quick*. All words between the reference text and the hypothesis are different (“cats” and “cat” are, for example, made different by the addition of the “s”). This means that with the similarity naïve standard approach used in WER with edit costs fixed at 1.0, any and all alignments are equally valid and the resulting alignment will be chosen at random. The local edit costs and the cumulative costs for the standard alignment can be found in Table 3.

Unlike with the standard alignment, Match ‘em discounts the costs of substituting similar words. Thus, although “cats” and “cat” are different words the cost for substituting them is only 1/4 (the CER between them). Thus, as shown in Table 4 (left), the costs for substitutions of the words “cats”, “run”, and “quickly” are less than one and when incorporated into the full costs (Table 4 (right)) an obvious best path is presented: one which results in the third alignment in Table 2.

2.3 Compounding

After the preliminary word-level alignment described in Section 2.2, Match ‘em accounts for errors around compound words or, equivalently, it accounts for errors created by adding or delet-

ing one or more space characters. With the standard Levenshtein alignment, the breaking up or creation of a compound word inflicts two edits: a deletion or insertion, as well as a substitution. For example, in Figure 1 (right), there would be a substitution between the words “cannot” (reference) and “can” (ASR) as well as an insertion of the word “not” (ASR). However, the difference really is the insertion or deletion of a space (a character). As exemplified by the figure, Match ‘em allows to classify this as a single word substitution at the word level, and as a single character insertion (the space) at the character level. It accomplishes this by iteratively checking the neighbouring words to every edit (substitutions, insertions or deletions). For every iteration, the neighbouring word is attached to the current word if the operation results in a lower character level cost. In the example, “not” is attached to “can” because this results in a reduction of word dissimilarity from 3 (“cannot” vs “can”) to 1 (“cannot” vs “can not”). This process is repeated as long as the cost decreases, allowing for compounds of several words.

3 Experiments

To evaluate the impact of this new alignment method, audio in three different European languages was transcribed using two state-of-the-art ASR model architectures. The three languages (Norwegian, Italian, and English) were chosen for a variety of reasons. Firstly, Match ‘em re-

quires languages with alphabets for which character articulatory vectors can be defined—thus excluding languages that use syllabaries or logographies, such as Japanese or Chinese, respectively. Also, these three languages cover multiple language families (Germanic and Romance), orthographic depths (Norwegian and Italian spellings being largely phonetically written as opposed to English being irregular (Seymour et al., 2003)), and dialectal variations (both Norwegian and Italian contain a large amount of dialectal variation compared to English (Kinder and Savini, 2004; Skjækkeland, 1997)). Additionally, Norwegian Bokmål allows for multiple legal spellings of words (e.g., *vet* and *veit* both being legal spelling for the present tense of *å vite* (“to know”). Lastly, Norwegian utilizes compounding of words (again with common, but perhaps less legal, variations to spellings) to a higher degree than English or Italian, which gives us an opportunity to test how Match ‘em performs on this aspect.

3.1 Datasets

For both Italian and English data, we used the VoxPopuli corpus (Wang et al., 2021), which consists of recordings from the European Parliament. As parliamentary recordings, the speech style is largely spontaneous with a good distribution of speakers. In the Italian corpus, we removed a number of utterances where there was a significant mis-alignment between the audio and human-generated transcriptions.

As Norway is not part of the European Parliament, the NB Tale dataset (National Library of Norway, 2015) was used instead of VoxPopuli. NB Tale is publicly available through the Norwegian National Library’s Language Bank and contains a good variety of speakers. In our experiments, we only used the subsection of NB Tale consisting of spontaneous speech recordings produced by native speakers, to better align with the speech style for Italian and English. All of the speech in NB Tale is human-transcribed using the Bokmål written standard.

3.2 Models

To generate transcriptions for our alignment analysis, we employed two end-to-end model architectures, wav2vec 2.0 (Baevski et al., 2020) and Whisper (Radford et al., 2022). The transcriptions are either generated as characters for wav2vec 2.0, or as byte pair encodings (effectively word-level

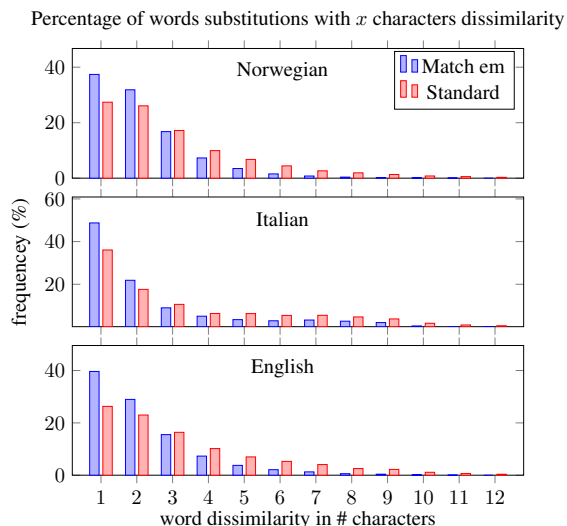


Figure 2: Percentage of word substitutions as a function of word dissimilarity in number of characters. Results are accrued over both wav2vec 2.0 and Whisper model outputs.

and character-level output (Radford et al., 2019)) for Whisper. These flexible outputs allow for potentially novel spellings and therefore constitute a good test bed for Match ‘em. Finally, both wav2vec 2.0 and Whisper have reported impressive accuracies, making them ideal candidates to generate reasonable transcriptions to evaluate.

For the wav2vec 2.0 architecture, we used different models depending on the language. For Italian and English, we used the VoxPopuli multilingual model (Wang et al., 2021) without a language model (LM). This model contains approximately 300 million parameters. However, the VoxPopuli model does not contain Norwegian. Thus, for Norwegian, we used the 300 million parameter wav2vec 2.0 model created by the Norwegian National Library AI Lab (De La Rosa et al., 2023) run with a LM. For the Whisper architecture, we used the same multi-lingual model (large-v2) for all languages. This model, unlike the wav2vec 2.0 counterparts, was trained to perform multiple tasks, including ASR in English and other languages, any-to-English translation, and non-speech detection. The Whisper model contains 1550 million parameters and was trained on 680,000 hours of loosely-supervised Internet audio (117,000 of those hours being in languages other than English). This model was run with a LM.

3.3 Implementation

We implemented the Match ‘em framework as a Python package¹. This implementation has been designed with a high degree of flexibility, allowing many features to be specified as runtime parameters. These include selecting which alignment to use (Levenshtein vs Match ‘em), whether compounding should be reconciled, and what kind of character dissimilarity to use (binary or vector based). The articulatory vectors, described in Section 2.1, are included in the Match ‘em repository. The vectors are defined in JSON format and can be easily expanded or edited for other letter-based orthographies.

4 Results

4.1 Word substitution similarity

Figure 2 considers word substitution pairs (excluding word insertion or deletion) from both the wav2vec 2.0 and Whisper text. In order to assess the quality of the alignment, we evaluate how many characters are different between the words in a substitution pair. In the figure, we can see that Match ‘em increases the frequency of word-pairs with a small orthographic distance. For example, consider all the pairs with only one character difference. For Match ‘em these account for 37.4% (Norwegian), 48.74% (Italian), and 36.64% (English) of all the substitution errors. These percentages are approximately ten percentage points higher than the corresponding values for the standard alignment (Norwegian: 27.39%, Italian: 36.07%, English: 26.31%).

As Match ‘em better aligns similar words, we can use it to analyze the types of character errors occurring within words. This is fundamentally different than analyzing character errors from standard CER alignment because it allows us to focus on errors in specific parts of the words that carry specific meaning. CER, as is typically computed, ignores word boundaries. Thus, while it may provide insight into which characters are frequently missed, it loses any information that might indicate what role those letters played. The value of character-aware error analysis can be illustrated by (Parsons et al., 2023)

As an example, we investigated word substitutions where only the final character changed.

¹<https://github.com/scribe-project/match-em>

Dataset	wav2vec 2.0		Whisper	
	Standard	Match ‘em	Standard	Match ‘em
English	3.92	4.67	8.43	10.22
Italian	11.48	13.52	12.04	14.63
Norwegian	5.66	7.62	5.17	6.69

Table 5: The percent of word substitutions produced by Match ‘em alignment where only the final character changed. The most common errors were considered (Norwegian: “e” or “r”, Italian: all vowels, English: “s”).

From there, we observed the most common character changes for each language. For Norwegian, these characters were “e” and “r”; while for English, it was the character “s”. For both of these languages, insertion or deletion of these characters will change the quantity of a noun or the tense of a verb. For Italian, the vast majority of words ends in a vowel, where the final vowel marks both gender and quantity of a word. Due to the frequency and similar semantic load, we considered all final vowels in Italian in our analysis. The percentage of all word substitutions containing just this final letter change are presented in Table 5. Through this we see that not only does Match ‘em align more instances of final letter change but that a sizeable amount of all substitution errors are just the final letter change. Such a final letter change might alter a word’s meaning slightly, but will rarely destroy the meaning of an entire sentence. Consequently, depending on the task at hand, those errors may be given higher or lower weight in ASR development.

4.2 Compounds

As described in Section 2.3, Match ‘em also attempts to recognize and rectify compounding errors. Although the majority of compounds include the concatenation of two words, in the Norwegian data we see that Match ‘em is able to account for cases where more than two words are combined, such as “to tusen og tolv” and “totusenogtolv”. Both these written forms are valid in Norwegian and have the same meaning (*two thousand and twelve*). In English, many of the compound pairs are contractions (e.g., “it is” vs. “it’s”, “we are” vs. “we’re”) where the difference is not only the space but also the substitution of character(s) for an apostrophe.

As this method works on the surface level of words, without any context of word meaning(s),

there is the potential that the compound word pairs while being similar in characters are actually semantically distinct. The most common pair found in our Norwegian data (“og så” - “også”) demonstrates this well because the two variants can be translated to English as *and so* and *also*, respectively. Most contractions that are seen in both English and Italian carry the same semantic content. As an exception, some Italian contractions should be considered as misspellings (like “un’Europa” instead of the correct “un’Europa” or “una Europa”). Regardless, as the meanings would still be interpretable by a human, the reduction in penalty for the compounding mistake is well justified. Given the success of the compounding analysis, we believe that more highly synthetic languages, such as Finnish, may be good candidates for Match ‘em analysis in future work.

Analyzing the difference in compounding errors between wav2vec 2.0 and Whisper gives some insights for the potentially different behaviour of these two models. For English, the top 10 most frequent compounding errors are nearly the same for both models and contain typical contractions (e.g. “it is” vs “it’s”). The numbers of errors are also comparable. For Italian, the Whisper model has a much lower number of compound errors compared to wav2vec 2.0 (see also Section 4.3). For Norwegian, the two models make a comparable number of compound errors, the most common of which is “og så” versus “også”. However, after “og så” and “også”, frequency of specific compound errors is different between the two models. Further analysis of these phenomena may give insights into the workings of these two architectures.

4.3 Standard versus Match ‘em WER

The goal of Match ‘em is to produce a better word alignment for detailed error analysis. It is, however, interesting to study how Match ‘em modifies the WER. If we exclude the compounding reconciliation, the better alignment does not change the total number of errors (insertions, deletions and substitutions), although it may change their relative distribution. Changes in WER are, therefore, an exclusive result of compounding reconciliation, where we keep a single substitution and reduce the number of insertions and deletions. Table 6 demonstrates this by showing the WERs computed with Levenshtein (standard) and Match ‘em alignment for the three test languages and two model

Dataset	wav2vec 2.0		Whisper	
	Standard	Match ‘em	Standard	Match ‘em
Norwegian	22.07	21.06	21.50	20.81
Italian	20.55	18.87	13.54	13.28
English	19.87	17.92	14.80	14.49

Table 6: The WER for each language, model, and alignment method.

architectures. As expected, by resolving compounding errors, Match ‘em results in a lower WER. The reduction is greater for wav2vec 2.0 which, as noted in Section 4.2, produces a higher number of compounding errors than Whisper. As mentioned in Section 2.3, however, it is not clear if lower is truly better here.

5 Conclusions

We propose the new Match ‘em framework for creating better alignment between reference and ASR-generated transcriptions both at the word and character level. We show that Match ‘em allows for a deeper understanding of ASR performance compared to WER, by supporting detailed analysis of common errors. By using word dissimilarity metrics and by reconciling compound errors, Match ‘em alignment results in word substitution pairs that are more similar compared to standard Levenshtein alignment. We show that analysis of these substitution pairs can yield insights into the potential semantic impacts of these errors. Our claims are verified across three European languages (English, Italian and Norwegian) and two state-of-the-art ASR architectures (wav2vec 2.0 and Whisper). We believe the Match ‘em framework to be a useful tool for other ASR researchers for gaining insights into their own models’ performances and, more generally, for speech researchers to gain linguistic insights by analyzing ASR errors on large annotated speech corpora.

Acknowledgments

This work has been done as part of the SCRIBE project as funded by the Norwegian Research Council, project number: 322964.

References

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Advances in Neural Information Pro-*

- cessing Systems, volume 33, pages 12449–12460. Curran Associates, Inc.
- Catia Cucchiaroni. 1996. Assessing transcription agreement: Methodological aspects. *Clinical Linguistics & Phonetics*, 10:131–155.
- Javier De La Rosa, Rolv-Arild Braaten, Per Kummer-vold, and Freddy Wetjen. 2023. Boosting Norwegian automatic speech recognition. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 555–564, Tórshavn, Faroe Islands. University of Tartu Library.
- Sharon Goldwater, Dan Jurafsky, and Christopher D Manning. 2010. Which words are hard to recognize? prosodic, lexical, and disfluency factors that increase speech recognition error rates. *Speech Communication*, 52(3):181–200.
- Suyoun Kim, Abhinav Arora, Duc Le, Ching-Feng Yeh, Christian Fuegen, Ozlem Kalinli, and Michael L. Seltzer. 2021. Semantic Distance: A New Metric for ASR Performance Analysis Towards Spoken Language Understanding. In *Proc. Interspeech 2021*, pages 1977–1981.
- J. J. Kinder and Vincenzo M. Savini. 2004. *Using Italian: a guide to contemporary usage*, chapter 1. Cambridge University Press.
- Knut Kvale and Arne Kjell Foldvik. 1992. The multifarious r-sound. In *Proc. International Conference on Spoken Language Processing (ICSLP-92)*, pages 1259–1262.
- Vladimir I. Levenshtein. 1965. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet physics. Doklady*, 10:707–710.
- Alianda Lopez, Andreas Liesenfeld, and Mark Dingemanse. 2022. Evaluation of automatic speech recognition for conversational speech in dutch, english and german: What goes missing? In *Proceedings of the 18th Conference on Natural Language Processing (KONVENS 2022)*, pages 135–143.
- Andrew Cameron Morris, Viktoria Maier, and Phil Green. 2004. From WER and RIL to MER and WIL: improved evaluation measures for connected speech recognition. In *Proc. Interspeech 2004*, pages 2765–2768.
- Hiroaki Nanjo and Tatsuya Kawahara. 2005. A new ASR evaluation measure and minimum Bayes-risk decoding for open-domain speech understanding. In *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 1, pages I/1053–I/1056 Vol. 1.
- National Library of Norway. 2015. NB Tale – Speech Database for Norwegian.
- Phoebe Parsons, Knut Kvale, Torbjørn Svendsen, and Giampiero Salvi. 2023. A character-based analysis of impacts of dialects on end-to-end Norwegian ASR. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 467–476, Tórshavn, Faroe Islands. University of Tartu Library.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust Speech Recognition via Large-Scale Weak Supervision. *arXiv*.
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. *arXiv*.
- Somnath Roy. 2021. Semantic-wer: A unified metric for the evaluation of ASR transcript for end usability. *CoRR*, abs/2106.02016.
- Janine Rugayan, Torbjørn Svendsen, and Giampiero Salvi. 2022. Semantically Meaningful Metrics for Norwegian ASR Systems. In *Proc. Interspeech 2022*, pages 2283–2287.
- Askars Salimbajevs and Jevgenijs Strigins. 2015. Error analysis and improving speech recognition for Latvian language. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 563–569, Hissar, Bulgaria. INCOMA Ltd. Shoumen, BULGARIA.
- Philip Seymour, Mikko Aro, and Jane Erskine. 2003. Foundation literacy acquisition in European orthographies. *British Journal of Psychology*, 94:143–174.
- Martin Skjækkeland. 1997. *Dei norske dialektane: tradisjonelle særdrag i jamføring med skriftmåla*. Høyskoleforl.
- Ioana Vasilescu, Martine Adda-Decker, and Lori Lamel. 2012. Cross-lingual studies of asr errors: paradigms for perceptual evaluations. In *LREC*, pages 3511–3518. Citeseer.
- Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. 2021. VoxPopuli: A Large-Scale Multilingual Speech Corpus for Representation Learning, Semi-Supervised Learning and Interpretation. 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 993–1003, Online. Association for Computational Linguistics.
- Johannes Wirth and Rene Peinl. 2022. Automatic speech recognition in german: A detailed error analysis. In *2022 IEEE International Conference on Omni-layer Intelligent Systems (COINS)*, pages 1–8.

Adding Metadata to Existing Parliamentary Speech Corpus

Phoebe Parsons¹

Per Erik Solberg²

Knut Kvale³

Torbjørn Svendsen¹

Giampiero Salvi¹

¹Department of Electronic Systems, NTNU, Trondheim, Norway

²National Library of Norway, Oslo, Norway

³Telenor Research and Innovation, Oslo, Norway

{phoebe.parsons, torbjorn.svendsen, giampiero.salvi}@ntnu.no,
per.solberg@nb.no, knut.kvale@telenor.com

Abstract

Parliamentary proceedings are convenient data sources for creating corpora for speech technology. Given its public nature, there is an abundance of extra information about the speakers that can be legally and ethically harvested to enrich this kind of corpora. This paper describes the methods we have used to add speaker metadata to the Stortinget Speech Corpus (SSC) containing over 5,000 hours of Norwegian speech with non-verbatim transcripts but without speaker metadata. The additional metadata for each speech segment includes speaker ID, gender, date of birth, municipality of birth, and counties represented. We also infer speaker dialect from their municipality of birth using a manually designed mapping between municipalities and Norwegian dialects. We provide observations on the SSC data and give suggestions for how it may be used for tasks other than speech recognition. Finally, we demonstrate the utility of this new metadata through a dialect identification task. The described methods can be adapted to add metadata information to parliamentary corpora in other languages.

1 Introduction

There has been, historically, a lack of high quality, freely available speech resources for machine learning tasks. Traditionally, these resources have been created to facilitate development of automatic speech recognition (ASR) models, and as such have been expensive to create, requiring human hours for both data collection and then careful, verbatim transcription. Even for “well resourced” languages like English, datasets rarely exceeded 1,000 hours. However,

as new ASR technologies loosen the requirements for transcription precision, this allows for even larger datasets that are created from less verbatim sources (Chen et al., 2021; Galvez et al., 2021). These new, more loosely supervised datasets often lack details found in older, more traditional speech resources and are therefore potentially limited in their application.

Many established speech resources are composed of relatively short duration segments with speech from only one speaker at a time. Additionally, this speaker is often known (even if only by an anonymized speaker identifier) and metadata, such as age and gender, is given about them. This richness of metadata allows for speech technology and machine learning tasks beyond ASR — such as language (or dialect) identification, speaker diarization, speaker identification or verification. Crucial to all these tasks is knowledge about who is speaking.

Recently, a number of speech corpora were created from public domain recordings of parliamentary proceedings; for instance, Iceland (Helgadóttir et al., 2017), Denmark (Kirkedal et al., 2020), Finland (Virkkunen et al., 2023), Croatia (Ljubešić et al., 2022) and the European Parliament (Wang et al., 2021). In all of these works it is known, at the very least, who is speaking in each segment (either by name or speaker ID), with most also including gender information. Virkkunen et al. explored their dataset using the rich metadata they were able to pull from an open API providing both distribution information and ASR results along age, gender, and educational background lines. However, it appears that this rich metadata was not released with the final dataset. Ljubešić et al. included name, gender, year of birth, party affiliation and party status for their speakers.

In 2023, the National Library of Norway (NB) developed the Stortinget Speech Corpus (SSC) (Solberg et al., 2023) using data from the Norwe-

gian parliament (called *Stortinget* in Norwegian). In early 2024, NB published the results of their analysis of several ASR systems for Norwegian (Solberg et al., 2024). In this report they showed that Whisper models (Radford et al., 2022), fine-tuned on the SSC and some additional smaller datasets, performed best on an unseen test set created from radio and TV program audio. This fine-tuned model outperformed both the base Whisper model as well as fine-tuned wav2vec (Baevski et al., 2020) models and commercial ASR systems from Google and Microsoft, thus demonstrating the importance of this speech corpus in combination with the Whisper architecture for ASR.

Despite the SSC’s obvious utility in training well-performing ASR models, it, as originally created, contains no metadata for each speech segment. We believe the effort to construct the missing metadata has merit as expanding the SSC into other speech technology domains would be a benefit for Norwegian speech research. To that end, we have undertaken the effort of ensuring that each segment in the SSC has been matched to a speaker identifier and that public speaker metadata has been added. As a result of this effort, this new metadata is now included with the SSC and made available by the Norwegian Language Bank at the National Library of Norway. Furthermore, we offer a recommendation for a subset of the SSC that more closely resembles traditional well annotated speech corpora and may be more applicable to other speech tasks. Finally, we believe that the efforts described in this paper can be easily extended and applied to similar corpora in other languages and countries.

2 The Pre-Existing SSC Dataset

The SSC contains more than 5,000 hours of natural Norwegian speech paired with non-verbatim transcripts created from the Norwegian parliament. The National Library of Norway created the SSC by following the technique described by (Ljubešić et al., 2022). They first broke the plenary meetings into segments using voice activity detection. Shorter segments were combined resulting in each SSC segment being roughly 30 seconds. In doing this, no concern was given to speaker boundaries. That is, the 30 second segments were created from files containing recordings of a whole day’s worth of parliamentary discussion, without awareness of who was speaking or whether there

was one or multiple speakers in the segment. Thus for each segment in the SSC, no speaker metadata is available.

After the audio had been segmented, an ASR system was then used to generate transcripts for these segments. The Levenshtein ratio¹ was then used to align the ASR output with the text of the official parliamentary proceedings sourced from the ParlaMint-NO corpus². The proceedings were human-transcribed at the utterance level with some light editing and omissions for standardization and legibility. Because the official proceedings are not a verbatim transcription of the spoken utterances, the ASR transcriptions may deviate considerably from the proceedings. Consequently, only segments where the score produced by the Levenshtein ratio between the proceedings text and the ASR text was above a threshold (0.5) were kept. For the selected 30 second segments, the proceedings text was taken as the transcription. The Levenshtein ratio score was also kept in the SSC. In this manner, the SSC was created.

3 Speaker Metadata

3.1 Recovering Speaker Information from ParlaMint

The first objective of this work was to recover who is speaking in each segment of the SSC. To do this, we turn our attention to the Norwegian ParlaMint-NO text corpus. As mentioned in Section 2, this corpus contains the proceedings text. Additionally, it is annotated with metadata on speaker identity, gender, date of birth, and which of the two written forms of Norwegian the transcript is in for every utterance.

The task of reconciling the SSC text and the speaker metadata was done using word offsets. When creating the SSC, ASR output was aligned with the proceedings from ParlaMint. Though the metadata available in ParlaMint was discarded during the original creation of the SSC, the word offsets — the index of the starting and ending words in the ParlaMint proceedings — were preserved for each approximately 30 second segment. We can then join the ParlaMint metadata and the SSC segments by reconciling the offsets.

To illustrate how this reconciling of word off-

¹<https://rapidfuzz.github.io/Levenshtein/levenshtein.html#ratio>

²<https://www.nb.no/sprakbanken/ressurskatalog/oai-nb-no-sbr-77/>

Utt ID	Speaker ID	Proceedings Text	Start	End
1	person.1	Good morning all	0	2
2	person.1	Today we will be starting with Representative Smith	3	10
3	person.2	As many of you know, moose in Norway are a common sight	11	22
4	person.2	Therefore we propose that	23	26

Table 1: A synthetic example of ParlaMint utterances. The starting and ending word indexes have been added for each utterance.

set and metadata occurs, we refer to the the fictional snippet of ParlaMint utterances presented in Table 1. Let us assume that we have an SSC segment where the text offsets with respect to that ParlaMint snippet are between 5 and 24. We first determine which utterance contains the word offset 5, in this case utterance 2 (the start index is smaller than 5 and the ending index is larger). We then determine which utterance contains the index 24, in this case utterance 4. As utterance 3 is between our starting and ending utterances we assume it too aligns with the current SSC segment.

By aligning the text in the SSC with the ParlaMint text, we have recovered the speaker information from each segment in the SSC. In addition to speaker identifiers (represented as person.1 and person.2 in the example), we also now have the date of birth, gender, and Norwegian written form (or forms) used in each segment. The speaker identifiers can be used to add further metadata, as described in the following sections.

3.2 Stortinget API

Beyond the metadata available from ParlaMint, we believed it to be useful to add publicly available information to the corpus, including the municipality and county where the speaker was born, as well as the county or counties represented by that speaker. This was accomplished by use of the Stortinget application programming interface (API)³. The Stortinget API provides a programmatic way to access data about the Norwegian parliament, including endpoints for bibliographic information on the speakers in the parliament. All metadata from the API is covered by a Norwegian Licence for Open Government Data⁴ which permits copying, using and distributing information from the API. The endpoint `kodetbiografi` contains information on the speaker’s municipality of birth, county of birth, and counties represented.

³<https://data.stortinget.no/>

⁴<https://data.norge.no/nlod/en/2.0>

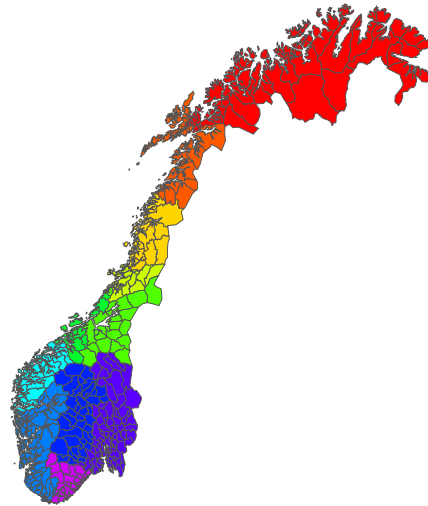


Figure 1: The municipalities of Norway mapped to dialect regions. The eastern dialect regions have been collapsed from (Skjekkeland, 1997).

We called this endpoint for each speaker using the speaker ID from the ParlaMint utterances. Not all speakers have information provided for each of the three fields that we were interested in. If one of these fields lacked information for a speaker, no further efforts were made to find this information, both on practicality and privacy grounds.

3.3 Municipality to Dialect

Our aim in gathering this municipality and county information was to enable an automated method of assigning presumed dialect. That is, given that Norwegian dialects are largely decided along geographic lines (Sandøy, 1987, p. 16), we hoped to use the municipality of birth to infer which dialect a person is likely to be speaking in.

The Norwegian language has no official standard speaking style (The Language Council of Norway). Hence, there is a large variety of dialectal realizations manifesting in pronunciation, lexical items, and grammar. Additionally, the culture encourages people to speak with their native

dialect in all situations from the least formal to the most. It is even common for speakers to retain their native dialects and to use dialectal lexical items when speaking in parliament. Thus, we find including dialect information to be both pertinent and, hopefully, useful to machine learning tasks related to speech.

To enable this automatic assignment of dialect, we created a mapping between all municipalities and counties in Norway and their assumed dialects. Using the dialect map created by Skjekkeland (Skjekkeland, 1997) as the ground truth, we manually analyzed maps of each county and their municipalities in order to align them with the boundaries drawn in Skjekkeland's map. Further, as we wished this municipality-to-dialect mapping to be useful with other existing Norwegian resources, historical municipalities and counties were included. As we found this mapping useful and was nontrivial to produce, it has been made available through the Norwegian Language Bank⁵.

This inference of dialect from birth municipality does not, of course, account for people who were born in one place then quickly moved to another. Nor does our inference take into account that speakers often tend to adapt their dialect, at least slightly, to the local or national "standard" dialect. Therefore, for speakers who represent the same county they were born in, one could assume that speaker is still, potentially, representative of the dialect label assigned. However, for the working going forward, we will be using dialect labels generated from the speaker's municipality of birth regardless of if they later moved to a new county.

4 Data observations

As stated earlier, the SSC was designed for loosely supervised ASR training, and has already been used for this aim⁶. However, other speech tasks require either a greater degree of faithfulness in transcription or audio with only one speaker per segment, or both. In order to understand which, if any, part of the SSC might be useful in these other tasks, an analysis of the data was performed.

⁵<https://www.nb.no/sprakbanken/en/resource-catalogue/oai-nb-no-sbr-92/>

⁶<https://huggingface.co/collections/NbAiLab/nb-whisper-65cb8322877f943912afcd9f>

4.1 Towards verbatim transcripts

While non-verbatim transcripts work well for weakly-supervised training of ASR models such as Whisper, other ASR frameworks (e.g. wav2vec 2.0 (Baevski et al., 2020)) still require transcripts that align more closely with the audio. Thus, we begin to look at the transcripts available to us to understand how often they align.

As described in Section 2, each SSC section has a score denoting the similarity between the proceedings text and the verbatim transcripts produced by ASR. It follows that when the proceedings text has a high similarity score to the ASR output the SSC text is presumably verbatim. However, these similarity scores are not infallible as ASR errors could lower the score regardless if the the SSC text was actually verbatim.

Despite the potential for ASR errors, we have observed that low scores are often a result of spoken information being omitted from the proceedings text. During proceedings, the Stortinget president often introduces the next speaker or provides other administrative information. Additionally, other speakers often recognize the president, have false starts in their sentences, or include other unnecessary words. As the proceedings are meant to be read, the transcribers tasked with creating the proceedings omit and lightly editorialize for readability. Thus, as can be seen in the example in Table 2, introductions of the next speaker (which would be obvious from the names associated with each utterance when reading the transcript) are not included in the proceedings.

We have found that, as a general rule of thumb, segments with Levenshtein ratios over 0.8 are highly accurate. While some segments achieve a perfect score of 1.0, they only account for 13.5 hours of the over 5,000 total hours in the SSC. Whereas, if all segments scoring over 0.8 are included, then over 3,300 hours of data is available.

4.2 One-speaker segments

As tasks such as speaker identification or dialect recognition generally require audio segments with only one speaker, identifying subsets of the SSC where there is only one speaker is beneficial.

This can be done by either finding segments in the SSC corpus that already contain a single speaker, according to the metadata, or by splitting multiple-speaker segments into a number of single-speaker sub-segments. To assess the impact

Speaker ID	person.LHH	person.DTA	person.TRJ
Proceedings	dette løftebruddet?		Nei, jeg tror
Transcription	dette løftebruddet	statsråd ris johansen	nei president jeg tror
English	this breach of promise	minister ris johansen	No, president I think

Table 2: An example of different transcription standards.

Speakers in Segment	Count of Segments
1	624337
2	88560
3	11808
4	78

Table 3: Counts of segments in the SSC by the number of speakers, according to the new SSC metadata

of those strategies, we aggregated SSC segments by the number of speakers in Table 3. We can see that one-speaker segments account for 86.14% of the SSC segments. Thus, it is feasible to simply discard the segments with multiple speakers and still have over 4,478 hours of audio.

However, as mentioned when discussing the proceeding transcriptions, there are many instances where brief speaker turns are not included in the transcriptions. It follows then, that though the new metadata in the SSC may only recognize one speaker, another speaker could have spoken and simply been omitted from the proceedings on the basis of readability.

4.3 Splitting multi-speaker segments

To fully use all data, segments with multiple speakers would ideally be split into one-speaker segments. We explored the simple approach of using forced alignment to align the text and the audio. The speaker utterance boundaries are known from the text and could be used to split the audio. However, in most instances the proceedings text is not verbatim enough for forced alignment. Forced alignment using the more verbatim ASR output could be feasible, however we then need to align the ASR output with the proceedings text (with a high degree of fidelity)—a non-trivial task. Ultimately, for future work, we see speaker diarization as a promising alternative for multi-speaker segments. Additionally, we have yet to explore how much, if any, of the speech in multi-speaker is overlapping, providing yet another avenue for

future work.

5 Comparison with the NPSC

The Norwegian Parliamentary Speech Corpus (NPSC) (Solberg and Ortiz, 2022) was created using data from 41 days of Norwegian parliament recordings where humans manually segmented and transcribed the data. Thus, the NPSC composes a small subset of the data available in the SSC. After listening to each speaker, the transcribers assigned each speaker in the NPSC a dialect. Five dialect regions were used for this task: Eastern Norway (from here on called East), Western Norway (West), Northern Norway (North), Trøndelag (Mid), and Southern Norway (South). Given this careful human supervision, the NPSC utterances may then serve as a “ground truth” for verbatim text, as well as speaker identities and dialects.

To reconcile the NPSC and the SSC, we could not use word offsets as the words are from fundamentally different sources (verbatim transcription versus official proceedings). However, the millisecond offset from the beginning of the day’s recording was preserved in both the NPSC and SSC segments. Therefore, we were able to use these millisecond offsets and the same approach as described with the word offsets to determine which NPSC utterances corresponded to each SSC segment.

5.1 One-speaker segments

As discussed in Section 4.2, there are potentially segments that the SSC metadata identifies as single-speaker, but in reality contains speech from multiple speakers. To understand the scope of this potential problem, we compare the speaker counts asserted by the SSC and the NPSC.

By doing this, we can see that when looking at utterances where the SSC metadata claims that only one speaker is present, we find that the NPSC believes there are more speakers 10.6% of the time. On the whole, we find that the NPSC and

SSC disagree on speaker counts 20.8% of the time. This implies that we should remain skeptical about the speaker counts given by the SSC, especially for tasks where it is crucial to have one, and only one, speaker in a segment.

5.2 Speaker dialect labels

As the NPSC also contains human prescribed dialect labels for each speaker, we can compare our inferred dialect labels with these ground truth labels.

There are 226 speakers in the NPSC, of which, metadata was available to assign the dialect to 164 of them. The dialect label from the SSC (as generated from municipality of birth) agreed with the NPSC human assigned label approximately 91.5% of the time.

Many of the speakers that we were unable to provide a dialect label for were speakers that spoke only a little or infrequently. Thus, the dialectally labeled speech accounts for 71.3% of the NPSC audio. Further, the duration of labeled audio in the SSC accounts for 78.6% of the audio, (over 4,000 hours), a similar percentage to the NPSC.

6 Automatic dialect identification

To demonstrate the utility of these new dialect labels, we have investigated the task of automatic dialect classification.

6.1 Model and fine-tuning

For the task of automatic dialect classification, we chose to fine-tune a model instead of creating a model from scratch. As a starting point, we took a model already fine-tuned for the language identification task⁷, itself fine-tuned from the Whisper-medium model⁸. The Whisper-medium model contains 769M parameters and was trained for ASR and speech translation on 680,000 hours of speech. The fine-tuning to language identification was done using the FLEURS dataset⁹ upon which the model achieved an accuracy of 0.88.

We then further fine-tuned the model from language to Norwegian dialect identification. Two models were trained, one using data from the

⁷<https://huggingface.co/sanchit-gandhi/whisper-medium-fleurs-lang-id>

⁸<https://huggingface.co/openai/whisper-medium>

⁹https://huggingface.co/datasets/google/xtreme_s#language-identification---fleurs-langid

NPSC, the other data from the SSC. This will allow us to understand the impact of the larger amount of data available in the SSC. For training, the first two convolutional layers in the encoder were fixed and each model was allowed to train for 3 epochs. The resulting model after these 3 epochs was used for the evaluation reported below.

6.2 Dataset splits

To prepare the NPSC and SSC for fine-tuning, the datasets were then divided into train, validation, and test sets by speakers. That is, a speaker (and all the utterances they said) would be assigned to one, and only one, of the three splits to ensure that the model was not simply learning the speaker's voice. As the NPSC is smaller, we utilized all of the NPSC where we had a dialect label for the speaker, resulting in a total of approximately 126 hours of speech.

We chose to use a subset of the SSC for the fine-tuning effort so as to have a dialectally balanced dataset. We determined which of the dialect regions contained the smallest amount of data (the South) and sampled data from each of the other regions to a similar size. This resulted in approximately 155 hours of data for each of the five dialect regions, or 774 hours of data in total. The size in both hours and number of speakers for both the NPSC and SSC training sets can be seen in Table 4.

To make a more direct comparison between the NPSC and SSC, we created a test set containing data from both. To do this, we removed speakers from the NPSC test set that appeared in the SSC training and removed speakers from the SSC test set that appeared in the NPSC training set. We then combined the remaining test data into a common NPSC+SSC test set.

6.3 Nordavinden og Sola

To evaluate how well these dialect identification models generalize beyond the parliamentary domain, we turned the Nordavinden og Sola (NVOS)¹⁰ (in English, *The North Wind and the Sun*) database. This database consists of speakers reading *The North Wind and the Sun* fable in Norwegian. Although the task was read speech, participants were allowed to alter the text, both in terms of lexical items and word order, to best fit their native dialects. The municipality for each

¹⁰<https://www.hf.ntnu.no/nos/>

	NPSC									SSC								
	train			validation			test			train			validation			test		
	seg	dur	spk	seg	dur	spk	seg	dur	spk	seg	dur	spk	seg	dur	spk	seg	dur	spk
east	21631	40	92	5826	4	12	2003	9	12	17690	127	92	2781	20	12	1192	8	11
west	12589	26	56	3288	5	7	2134	7	7	15481	111	60	3885	28	8	2191	15	7
mid	4560	9	22	208	2	3	906	0.5	3	17586	127	28	2298	16	4	1734	13	4
north	5230	11	26	549	3	3	1437	1	4	17092	123	27	2030	14	4	2361	16	3
south	3254	6	15	103	2	2	803	0.25	2	13245	95	11	3002	22	2	5315	36	1

Table 4: Amount of data available each data split for the NPSC and sub-sampled SSC. Quantities in number of segments, duration of speech in hours, and number of unique speakers.

speaker was recorded and from this we were able to assign one of the five cardinal dialects.

As the NPSC and SSC have different average utterance durations (NPSC utterances being an average of 7 seconds with a standard deviation of 5 seconds versus the SSC’s average of 25.8 seconds and standard deviation of 4.3 seconds), we created two test sets with the NVOS data with different utterance durations. In the first, the audio was left unaltered and the whole utterance (on average, about 32 seconds) was given to the model. In the second, we split each audio in half and then asked the model to classify these approximately 15 second audio clips.

6.4 Results and discussion

The accuracy, balanced accuracy, and weighted F1 from evaluating each of the two models (trained on NPSC or SSC) can be seen in Table 5. Metrics were calculated using scikit-learn 1.4.2. The model trained on the SSC data performs better than or equally well as the model trained on the NPSC for all test sets. When looking at the NPSC part of the combined test set, we can see that the SSC model performed as well as the in-domain NPSC model. However, the NPSC model performed very poorly when asked to predict using SSC audio.

Metrics for recall (macro and weighted) and F1 (micro and macro) were also calculated. However, as they follow the same general trend (where the model trained on SSC data performed as well or better than the model train on the NPSC, they are not included in this paper.

We can further observe from Table 5 that the length of the segment in the NVOS data has little impact. The SSC model did perform slightly better when presented with the full audio clips. This could be due to the fact that the segments in the SSC are approximately 30 seconds as well.

Confusion matrices for these tests are presented

in Figure 2. We find that both models often perform well on the East and West regions and poorly on the South. In fact, it is only in matrix (f) that a model predicts the South at all (of note as well, the only Southern speaker in the common test set is from the NPSC, meaning that the SSC model is robust enough to predict South for an out-of-domain speaker).

From these results, we can see that having more data even if not necessarily more speakers (211 speakers in the NPSC training set versus 218 in the SSC set) can positively impact model performance both in-domain and out.

While we are encouraged by the results presented here, there are several potentially confounding features. Our methodology for splitting the data along speaker lines does lead to imperfect datasets (for example, the South being represented by only one speaker in the test set, despite having the most hours of data 4). Further, no attention was paid to the content of the utterances. That is, within the parliamentary domain, it is conceivable that there are several set phrases that each speaker is apt to repeat. So, while there is no speaker overlap between the train, validation, and test sets, there is the potential for overlap of spoken content. Further, given the limited number of speakers, it is possible the that the model has learned some speaker-dependent features. Thus, we look forward to further exploring the impact of speaker and content on dialect identification in future works.

7 Conclusion

Through the efforts described in the paper, we enrich the SSC with speaker ID, gender, written form, age, dialect, municipality and county of birth and counties represented for each SSC segment.

Although the methods are developed for the Norwegian parliament, we believe they can rela-

	Trained on	NVOS half	NVOS full	Common test set		
				Total	NPSC	SSC
Accuracy	NPSC	0.75	0.75	0.60	0.74	0.45
	SSC	0.78	0.79	0.77	0.74	0.80
Balanced Accuracy	NPSC	0.63	0.63	0.48	0.52	0.58
	SSC	0.68	0.68	0.61	0.56	0.81
Weighted F1	NPSC	0.72	0.73	0.55	0.73	0.38
	SSC	0.77	0.78	0.76	0.73	0.81

Table 5: Accuracy, balanced accuracy, and weighted F1 of dialect identification models trained on either NPSC or SSC data. Models were evaluated against the NVOS dataset and the common dataset.

tively easily be adapted to parliamentary speech corpora in other languages.

The further aim of our work herein was to provide a subset of the large SSC that could be used for tasks beyond ASR. Thus, we provided observations on the corpus and suggested suitable subsets for different tasks in speech technology.

We demonstrated the utility of this new meta-data through a dialect identification task. The model trained using SSC outperformed the model trained with a smaller parliamentary corpus, thus showing an benefit of a corpus of the SSC’s size.

Finally, as a continuation of (Ljubešić et al., 2022) and (Solberg et al., 2023), this work provides a general template for how public datasets, such as parliamentary recordings, may be transformed into corpora for machine learning.

Acknowledgments

This work has been done as part of the SCRIBE project as funded by the Norwegian Research Council, project number: 322964.

References

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. [wav2vec 2.0: A framework for self-supervised learning of speech representations](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 12449–12460. Curran Associates, Inc.

Guoguo Chen, Shuzhou Chai, Guan-Bo Wang, Jiayu Du, Wei-Qiang Zhang, Chao Weng, Dan Su, Daniel Povey, Jan Trmal, Junbo Zhang, Mingjie Jin, Sanjeev Khudanpur, Shinji Watanabe, Shuaijiang Zhao, Wei Zou, Xiangang Li, Xuchen Yao, Yongqing Wang, Zhao You, and Zhiyong Yan. 2021. [GigaSpeech: An Evolving, Multi-Domain ASR Corpus with 10,000 Hours of Transcribed Audio](#). In *Proc. Interspeech 2021*, pages 3670–3674.

Daniel Galvez, Greg Diamos, Juan Ciro, Juan Felipe Cerón, Keith Achorn, Anjali Gopi, David Kanter, Maximilian Lam, Mark Mazumder, and Vijay Janapa Reddi. 2021. [The people’s speech: A large-scale diverse english speech recognition dataset for commercial usage](#). *CoRR*, abs/2111.09344.

Inga Rún Helgadóttir, Róbert Kjaran, Anna Björk Nikulásdóttir, and Jón Guðnason. 2017. [Building an ASR Corpus Using Althingi’s Parliamentary Speeches](#). In *Proc. Interspeech 2017*, pages 2163–2167.

Andreas Kirkedal, Marija Stepanović, and Barbara Plank. 2020. [FT Speech: Danish Parliament Speech Corpus](#). In *Proc. Interspeech 2020*.

Nikola Ljubešić, Danijel Koržinek, Peter Rupnik, and Ivo-Pavao Jazbec. 2022. [ParlaSpeech-HR - a freely available ASR dataset for Croatian bootstrapped from the ParlaMint corpus](#). In *Proceedings of the Workshop ParlaCLARIN III within the 13th Language Resources and Evaluation Conference*, pages 111–116, Marseille, France. European Language Resources Association.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. [Robust Speech Recognition via Large-Scale Weak Supervision](#). *arXiv*.

Helge Sandøy. 1987. *Norsk dialektkunnskap*. Novus Forlag.

Martin Skjækkeland. 1997. *Dei norske dialektane : tradisjonelle særdrag i jamføring med skriftmåla*. Høyskoleforl.

Per Erik Solberg, Pierre Beauguitte, Per Egil Kummer-vold, and Freddy Wetjen. 2023. [A large Norwegian dataset for weak supervision ASR](#). In *Proceedings of the Second Workshop on Resources and Representations for Under-Resourced Languages and Domains (RESOURCEFUL-2023)*, pages 48–52, Tórshavn, the Faroe Islands. Association for Computational Linguistics.

Per Erik Solberg and Pablo Ortiz. 2022. [The Norwegian parliamentary speech corpus](#). In *Proceedings of*

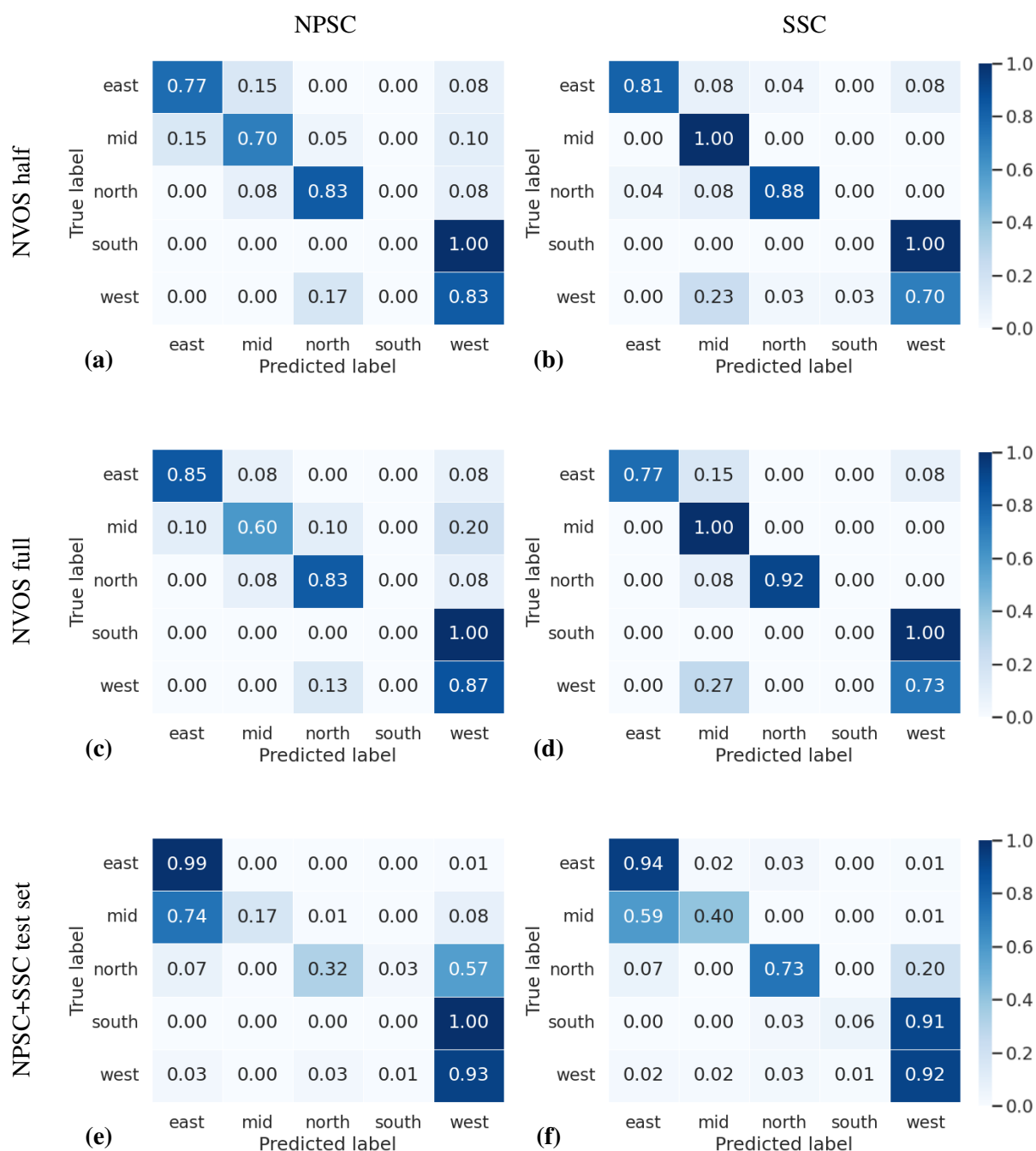


Figure 2: Normalized confusion matrices showing classifier performance on the three shared test sets. The first column (a, c, e) are from the model fine-tuned using NPSC data. The second column (b, d, f) are from the model fine-tuned using the SSC. The first row are the results when evaluated using the NVOS halves set, the second row the NVOS full set, and the third row the test set comprised of both NPSC and SSC data.

the Thirteenth Language Resources and Evaluation Conference, pages 1003–1008, Marseille, France. European Language Resources Association.

Per Erik Solberg, Marie Røsok, Ingerid Løyning Dale, and Arne Martinus Lindstad. 2024. [Status for norsk](#)

[talegenkjenning](#). Technical report, National Library of Norway.

The Language Council of Norway. Uttaleråd. <https://sprakradet.no/godt-og-korrekt-sprak/praktisk-sprakbruk/uttalerad/>. Accessed: 2025-01-07.

Anja Virkkunen, Aku Rouhe, Nhan Phan, and Mikko Kurimo. 2023. [Finnish parliament asr corpus: Analysis, benchmarks and statistics](#). *Lang. Resour. Eval.*, 57(4):1645–1670.

Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. 2021. [VoxPopuli: A Large-Scale Multilingual Speech Corpus for Representation Learning, Semi-Supervised Learning and Interpretation](#). 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 993–1003, Online. Association for Computational Linguistics.

Paragraph-Level Machine Translation for Low-Resource Finno-Ugric Languages

Dmytro Pashchenko and Lisa Yankovskaya and Mark Fishel

Institute of Computer Science

University of Tartu, Estonia

{dmytro.pashchenko,lisa.yankovskaya,mark.fisel}@ut.ee

Abstract

We develop paragraph-level machine translation for four low-resource Finno-Ugric languages: Proper Karelian, Livvi, Ludian, and Veps. The approach is based on sentence-level pre-trained translation models, which are fine-tuned with paragraph-parallel data. This allows the resulting model to develop a native ability to handle discourse-level phenomena correctly, in particular translating from grammatically gender-neutral input in Finno-Ugric languages. We collect monolingual and parallel paragraph-level corpora for these languages. Our experiments show that paragraph-level translation models can translate sentences no worse than sentence-level systems, while handling discourse-level phenomena better. For evaluation, we manually translate part of FLORES-200 into these four languages. All our results, data, and models are released openly.

1 Introduction

The existence of massively multilingual pre-trained translation models (e.g. m2m100, NLLB, and MADLAD-400: Fan et al., 2021; NLLB Team et al., 2022; Kudugunta et al., 2023) has made work on machine translation significantly easier by eliminating the need for training large models from zero. Nevertheless, even the largest of these models still leave many low-resource languages out—mainly due to lack of or difficulty to acquire textual data (monolingual or parallel) in those languages.

Moreover, these translation models approach translation by handling each sentence independently and thus do not handle discourse-level phenomena well¹. Ignoring the discourse-level phe-

nomena has been shown to pose problems for translation quality and its assessment (Bawden et al., 2018; Läubli et al., 2018). Even though decoder-only language models (e.g. GPT4, OpenAI et al., 2024) are an easy way to approach document-level translation, the availability of pre-trained open multilingual language models and their language coverage are even narrower than for translation models. Also, translation is more efficiently solved with sequence-to-sequence models when emergent abilities are not a requirement and the main purpose is to solve translation, not other tasks.

In this paper, we focus on developing machine translation for the Finno-Ugric family of languages, which is a good fit for addressing both aforementioned issues, namely support for low-resource languages and discourse-level phenomena ignorance:

- the majority of pre-trained models only support three languages from this family (Finnish, Estonian and Hungarian), with MADLAD-400 also including a few more, still leaving out dozens of languages, and
- Finno-Ugric languages have no grammatical category of gender and use gender-neutral pronouns. This increases their dependence on document-level context, see an example in Figure 1.

We narrow down our scope to four under-resourced members of the Finno-Ugric language family: Proper Karelian, Livvi, Ludian, and Veps. All four are low-resource languages and are not included in m2m100, NLLB, or MADLAD-400; they are also not supported by Google Translate² or DeepL³, as of January 2025.

but rather via monolingually denoising documents in several languages; translation is later taught to the model on sentence level.

²<https://translate.google.com>

³<https://deepl.com>

¹Although MADLAD-400 (Kudugunta et al., 2023) is pre-trained on full documents, this is done without cross-linguality

Text in Veps:	<u>Naine</u> tuli kodihe. Hänen mašin jäi garažas.
English translation:	<u>The woman</u> came home. Her car remained in the garage.

Figure 1: Example of translation challenges related to gender-neutral pronouns in Finno-Ugric languages: the Veps text includes the pronoun hänen, which can be translated both as “her” and “his”; resolving this ambiguity requires looking at the first sentence and the word naine (woman) as the antecedent.

With the issues listed above in mind, we collect paragraph-level corpora and develop paragraph-level machine translation models by simply fine-tuning sequence-to-sequence models on parallel paragraph pairs, comparing the results to sentence-level approach. In order to fit the paragraph into the context window of the model, we limit its length to five sentences at most—our experiments show that such a bounded context still allows the model to learn extrasentential dependencies.

Our key contributions are thus the following:

- We collect and release paragraph-level corpora for Proper Karelian, Livvi, Ludian, and Veps: monolingual, as well as parallel with Russian (Section 4).
- In order to evaluate the results, we extend part of the translation benchmark FLORES-200 by manually translating it into the new languages, as well as manually correct existing Russian translations for paragraph-level consistency (Section 4).
- We train both sentence-level and paragraph-level translation systems on the collected data and show that the latter has the same or better quality when applied to paragraphs as well as learns to translate discourse-level phenomena correctly (Sections 5 and 6).

The collected data⁴, trained models⁵, and created benchmarks⁶ are released openly.

Next, we outline the related work in Section 2 and present the methodology in Section 3.

⁴<https://huggingface.co/datasets/tartuNLP/pale-madlad-data>

⁵<https://huggingface.co/tartuNLP/pale-madlad-mt>

⁶<https://huggingface.co/datasets/tartuNLP/smugri-flores-testset>

2 Related Work

Document-level translation Elaborating on the importance of considering the extrasentential context in machine translation (MT), Bawden et al. (2018) describe major discourse-level phenomena that present problems for most MT systems: coreference, lexical cohesion, and lexical disambiguation. Taking into account the context beyond a single sentence is essential for correct translation. Throughout the history of MT, researchers tried to address this problem from different perspectives—from rule-based to statistical to corpus-based approaches—creating various document-level systems (Hardmeier, 2012; Hardmeier et al., 2013).

Currently, attempts have been made to incorporate context in the attention-based models’ scope by modifying their architecture. The researchers offered methods such as hierarchical attention (Miculicich et al., 2018) or memory networks (Maruf and Haffari, 2018) among others. However, the most straightforward strategies, like passing an entire text to the model, proved also the most effective. Sun et al. (2022) trained the Transformer model (Vaswani et al., 2017) on documents, repeatedly dividing them into parts to vary input lengths. Although this approach has shown a big leap in translation quality, it does not remedy another important problem: long processing times of large documents. The time and memory consumption of Transformer-based systems scales quadratically with the input length. We try to avoid this issue by splitting documents into small, fixed-size paragraphs rather than translating documents fully.

MT for low-resource Finno-Ugric languages

Machine translation for low-resource Finno-Ugric languages has been explored in a number of works. To name but a few, Tyers et al. (2009) examined rule-based and statistical MT systems when translating between North and Lule Sámi; Pirinen et al. (2017) employed rule-based MT in their North Sámi-Finnish system; Rikters et al. (2022) designed a neural MT system for Livonian. The languages studied in this work were presented in MT systems developed by Yankovskaya et al. (2023) and Purason et al. (2024), but unlike our approach, their systems do not take the document or paragraph context into account.

3 Methodology

In this chapter, we briefly describe our approach to dealing with paragraph-level data, ways to extract paragraphs from documents and evaluate paragraph-level translations. We chose MADLAD-400 as the basis for our experiments, since, in addition to being a small, powerful, and open-source model, it has the potential for paragraph-level translation as it was pre-trained with document-level monolingual data.

3.1 Splitting Documents into Paragraphs

With our primary task being to test whether including the extrasentential context improves the performance of MADLAD, we need to decide on how many sentences to use as the model’s input. On the one hand, the more sentences we take from a document, the more likely the model is to capture the necessary context for translating each sentence. On the other hand, passing the document as a whole as the model’s input may be impractical for two reasons:

- **Time and memory consumption.** The attention mechanism inside Transformers has quadratic computational complexity $\mathcal{O}(n^2)$, since the attention is calculated between each pair of tokens. Therefore, computation time and memory consumption increase quadratically with the input size. Shorter input sequences would ensure much faster model training.
- **Overfitting by length.** Varis and Bojar (2021) show that Transformers generalize badly to out-of-distribution input lengths. This means that loosening the restrictions on input length would require more training with diverse data (short and long) to avoid underfitting some lengths and overfitting the others. The stricter the restrictions—the easier the training.

We overcome the two aforementioned issues at once by splitting documents into smaller paragraphs of fixed, reasonable length. Since MADLAD was trained on sequences whose length did not exceed 256 tokens, we set a similar length limit. We abandoned the idea of forming paragraphs from as many sentences as possible to get close to the size limit, for this would have led to a low variance of data lengths. Instead, we combine a fixed number of sentences. If the paragraph length exceeds

256 tokens, we split the paragraph in two; if the paragraph is still too long but consists of a single sentence, we trim the paragraph to the maximum length.

Through experimentation, we have found that, on average, five sentences are enough to fit into the context window of 256 tokens on our training data without resorting to unnecessary splitting or truncation of paragraphs. Where the number of sentences is not divisible by 5, we take the remainder as a separate paragraph. We emphasize that there is no optimal choice of paragraph length and it should instead be chosen empirically or based on the model’s context length and the available data.

3.2 Evaluating Paragraph-Level Translations

The most popular surface-level metrics, BLEU (Papineni et al., 2002) and chrF++ (Popović, 2017), were designed to evaluate sentences. Applying them to paragraphs could compromise correlation with human judgments. Deutsch et al. (2023) have proved the opposite: BLEU scores for paragraphs not only align with those of humans but also become more accurate as paragraph size increases. This finding allows us to adopt BLEU as a paragraph-level metric without the need to train custom scoring models, which is problematic due to the resource-constrained setting.

We also use chrF++ as it is more suited for morphologically rich languages, such as the ones from the Finno-Ugric family. Drawing on the formal similarity and correlation of the BLEU and chrF++ metrics, we apply the latter directly to paragraphs as well.

3.3 Managing Language Tokens

MADLAD-400 requires a language token to be manually prepended to the user’s input sequence. These tokens take the form `<2xx>`, where `xx` stands for a target language code. For instance, the sequence `<2en> Mitä kuuluu?` indicates that the Finnish sentence “Mitä kuuluu?” needs to be translated into English. Thus, we prepend four language indicators to the input sequences: `<2kr1>` for Proper Karelian, `<2lud>` for Ludian, `<2olo>` for Livvi, and `<2vep>` for Veps. The codes are taken from the ISO 639-3⁷ code set. As for the Russian language, MADLAD encodes it as `<2ru>`.

However, in this work, we do not expand MADLAD’s vocabulary with new language tokens. In-

⁷https://iso639-3.sil.org/code_tables/639/data

stead, to save effort and time, we do nothing and expect the model to learn the tokens solely based on their textual representations, which we prepend to inputs. Pilot experiments showed us that this approach is as effective as specifying language tokens explicitly.

4 Data

In this paper, we focus on (i) two dialects of Karelian: Livvi (olo) and Proper Karelian (krl)⁸; (ii) Ludian (lud), which is closely related to Karelian, but is considered a language in its own right (Pahomov, 2017); and (iii) Veps (vep). These are all endangered Finnic languages, mainly spoken in Finland and Russia.

4.1 Data sources for training

The majority of the training data was parsed from the following resources: two media portals oma-media.ru⁹ and yle.fi¹⁰, open corpus of Veps and Karelian languages VepKar (Boyko et al., 2022), and Wikipedia. We were unable to utilize other published datasets, as they primarily comprised sentences rather than documents (although sentence-level data can still help improve the overall translation quality).

A preliminary analysis of translations revealed that the MT system was mixing Livvi and Proper Karelian. A possible reason for this mixing could be the incorrect assignment of language labels to the source data. After studying the sources and consulting linguists, we discovered that the texts from the media portal “Omamedia” were not only written in Livvi, as we previously thought, but also in other varieties of Karelian language, mainly Proper Karelian. Using the language identification tool GlotLID (Kargaran et al., 2023), we redistributed the texts according to the new language labels.

We did minor preprocessing steps aimed at normalizing characters and removing redundant elements (e.g., useless Wikipedia sections) to extract coherent texts from the sources.

Table 1 presents the composition of the final dataset.

⁸Proper Karelian comprises Northern (Viena) Karelian and Southern Karelian. In this study, we use both varieties to train our MT system, but we test the output only in Northern (Viena) Karelian

⁹<https://omamedia.ru/en>

¹⁰<https://yle.fi/t/18-44136/fi>

4.2 Benchmark dataset

The benchmark dataset of low-resource Finno-Ugric languages published by Yankovskaya et al. (2023) contains Livvi, our language of interest. We extended this dataset by adding three more languages: Proper Karelian (Viena), Ludian¹¹, and Veps. Like Yankovskaya et al. (2023), we translated the first 250 rows of the FLORES dataset (NLLB Team et al., 2022); the translations from Russian were done by native speakers of these languages who have extensive translation experience.

Another important step was to modify the existing FLORES-200 test set, transforming it from a sentence-level set into a paragraph-level one. Fortunately, the FLORES-200 benchmark (NLLB Team et al., 2022) is a collection of short excerpts from Wikipedia, where sentences are sequential. All we had to do was isolate these paragraphs. When their length exceeded the maximum allowable, we manually divided them into smaller paragraphs in such a way as to avoid incurring a significant loss of context. Thus, the original 250 rows transformed into 87 paragraphs. However, when verifying the consistency of paragraphs, we noticed that the sentences in the data set were probably translated separately, out of context. Therefore, we manually edited the paragraphs, ensuring the correct and consistent use of pronouns, names, terms, etc. in the Russian segment of FLORES.

We shall refer to these benchmarks sets as “Smugri FLORES benchmark.”

5 Experimental Setup

To investigate the effect of paragraphs on the quality of translation of Proper Karelian, Livvi, Ludian, and Veps, we fine-tune two MADLAD models: one on sentence-level data and the other on paragraph-level data. We translate the languages into Russian and vice versa. Russian was chosen as a translation objective (among other high-resource languages available in MADLAD) because most of the openly available parallel texts were aligned with the Russian language.

To further improve the model, we perform back-translation making use of our monolingual data. We back-translate in a single direction—from Finno-Ugric languages to Russian—and thus, enhance the quality of translation from Russian to Finno-Ugric languages (otherwise quite low). We

¹¹using the alphabet with ü instead of y

	krl		lud		olo		vep	
data source	mono	para	mono	para	mono	para	mono	para
vepkar-sent	45.4	32.3	5.9	7.9	36.0	22.2	38.3	20.4
vepkar-par	9.6	6.9	1.2	1.6	7.6	4.8	8.1	4.5
wikipedia-sent	-	-	-	-	28.4	-	99.8	-
wikipedia-par	-	-	-	-	7.7	-	24.0	-
omamedia-sent	8.3	-	-	-	3.5	-	6.5	-
omamedia-par	2.0	-	-	-	0.8	-	1.6	-
ylefi-sent	-	-	-	-	14.2	-	-	-
ylefi-par	-	-	-	-	3.2	-	-	-
total-sent	53.7	32.3	5.9	7.9	82.2	22.2	144.6	20.4
total-par	11.7	6.9	1.2	1.6	19.4	4.8	33.7	4.5

Table 1: The distribution of sentence-level (sent) and paragraph-level (par) parallel data (para) and monolingual data (mono) by language in the final dataset. Quantities are given in thousands, rounded to the nearest tenth.

avoid back-translation between the four selected languages because low-quality synthetic data can harm the resulting performance instead of improving it (Yankovskaya et al., 2023).

Using the HuggingFace framework¹², we fine-tune both the sentence-level and paragraph-level model for 10 epochs under equal conditions. We set the hyperparameters of Seq2SeqTrainingArguments to their default values with the following exceptions:

- We limit the generation length to 256 tokens.
- Following the MADLAD-400 paper, we set up an inverse square root scheduler with 300 warmup steps.
- We distribute fine-tuning across 8 GPUs. To approximately equalize the number of optimization steps for both models, we adjust the batch size depending on the total amount of data: 8 examples for paragraph-level data and 32 examples for sentence-level data.

We perform fine-tuning on the LUMI¹³ supercomputer with AMD Instinct MI250X GPUs.

We use both models to translate paragraphs from the modified Smugri FLORES benchmark. For generation, we set the standard beam size of 5. We evaluate translations with the BLEU and chrF++ metrics, of which we use the SacreBLEU (Post, 2018) implementations. When calculating chrF++,

we count only word bigrams. To measure statistical significance and confidence intervals, we do bootstrap resampling with 1000 resamples.

6 Results

In this section, we examine the obtained results, starting with a quantitative analysis that presents translations from Proper Karelian, Livvi, Ludian, and Veps into Russian, as well as from Russian to these four languages. Next, we conduct a brief qualitative analysis. After this, we compare our results with those generated by the online machine translation engine Tartu NLP Neurotõlge¹⁴. Finally, we explore how well translation abilities transfer to the unseen case of English translation.

6.1 Quantitative analysis

We begin our analysis by comparing the translation quality of two MADLAD models—one trained with sentences (SL model) and the other trained with paragraphs (PL model)—as measured by the automatic metrics of BLEU and chrF++ (see Section 3.2). To translate paragraphs with the sentence-level system, we process them sentence by sentence and then merge back into a paragraph. Otherwise, when given a full paragraph, the SL system tends to translate it into a single complex sentence with multiple subordinate clauses, thus decreasing the scores.

The results are presented in Table 2, in which we also provide the scores of the base MADLAD

¹²<https://huggingface.co/>

¹³<https://lumi-supercomputer.eu/>

¹⁴<https://translate.ut.ee/>

	base	SL	PL	<i>p</i> -value
krl-ru	14.6 ± 1.7 / 40.2 ± 2.1	21.1 ± 1.9 / 48.7 ± 1.6	21.9 ± 2.0 / 49.5 ± 1.6	0.060 / 0.036
lud-ru	8.8 ± 1.6 / 31.1 ± 2.2	18.0 ± 1.8 / 45.2 ± 1.6	19.5 ± 2.0 / 46.1 ± 1.6	0.004 / 0.034
olo-ru	9.5 ± 1.6 / 31.9 ± 2.3	22.0 ± 2.0 / 48.2 ± 1.7	22.4 ± 2.2 / 48.9 ± 1.7	0.217 / 0.076
vep-ru	8.6 ± 1.5 / 30.8 ± 2.1	21.1 ± 1.8 / 46.5 ± 1.7	21.1 ± 1.8 / 47.0 ± 1.7	0.392 / 0.090
ru-krl	0.4 ± 0.2 / 3.0 ± 0.8	13.5 ± 1.5 / 46.8 ± 1.3	13.3 ± 1.6 / 46.9 ± 1.2	0.221 / 0.385
ru-lud	0.3 ± 0.1 / 2.5 ± 0.5	4.3 ± 1.1 / 34.1 ± 1.1	3.9 ± 1.1 / 33.8 ± 1.0	0.143 / 0.127
ru-olo	0.6 ± 0.4 / 2.9 ± 0.7	8.7 ± 1.4 / 40.7 ± 1.2	8.5 ± 1.4 / 40.2 ± 1.7	0.193 / 0.185
ru-vep	0.3 ± 0.1 / 3.0 ± 0.7	12.0 ± 1.4 / 43.1 ± 1.5	12.1 ± 1.6 / 42.4 ± 1.9	0.409 / 0.138

Table 2: Translation metrics for translation directions from/into Russian, BLEU and chrF++ scores (separated by slash) of base MADLAD, sentence-level (SL) MADLAD, and paragraph-level (PL) MADLAD evaluated on the paragraph-level Smugri FLORES benchmark. *p*-value is the probability that SL and PL models are the same with respect to each metric; *p*-value less than 0.05 indicates that the difference between the models is statistically significant (highlighted in bold).

model. In the last column, we provide a *p*-value for each translation direction. Our null hypothesis is that the two models, sentence- and paragraph-level, are the same model. In cases where the *p*-value is less than 0.05, we reject the hypothesis and conclude that the difference between the models is statistically significant, with one clearly outperforming the other.

First, we observe that the base MADLAD-400 model, with no fine-tuning, is able to translate Proper Karelian, Livvi, Ludian, and Veps into Russian with good initial quality. The Proper Karelian→Russian translation score goes as high as 14.6 BLEU or 40.2 chrF++. This probably indicates that the model’s knowledge of related languages (Finnish, Estonian, Russian) was successfully transferred to this case.

After fine-tuning, the results improved considerably. The paragraph-level (PL) model is significantly better than the sentence-level (SL) one in the case of Ludian→Russian translation. The difference is notable, reaching 1.5 BLEU points and 0.9 chrF++ points. The chrF++ scores further confirm the superiority of the PL model in the Proper Karelian→Russian direction. At the same time, the BLEU metric shows no significant difference. Finally, in all other cases, both metrics indicate that the SL and PL models, on average, perform equally well.

Thus, the paragraph-level model is no worse and, at times, strongly better than the sentence-level model. The difference is the most pronounced in the case of translation *into* Russian, giving us reason to believe that the PL model successfully re-

solves some discourse-level phenomena inherent in Finno-Ugric languages, such as gender-neutral pronouns. These phenomena occur rarely (yet they are important for high-quality coherent translation), and automated metrics do not necessarily reflect the extent to which they have been handled. To further investigate the issue, we qualitatively analyze translated texts.

6.2 Qualitative analysis

Next we present the results of manual qualitative analysis of paragraphs translations from the FLORES-200 benchmark. Although the number of discourse-level phenomena in the test set is quite limited, we managed to discover cases where (i) lexical cohesion must be preserved to translate terminology and proper nouns and (ii) where pronouns in different sentences must be aligned via coreference resolution. A detailed descriptions of errors presented in Table 3 and a summary is presented below.

The first part of the qualitative analysis addresses lexical cohesion, which refers to the consistent translation of terminology. The PL model translates terminology and names more consistently than the SL model across all languages and directions (*from* Russian and *to* Russian). While the PL model occasionally produces incorrect translations of names and terms, it typically does so consistently. In contrast, the SL model is inconsistent, translating a term or name correctly in one sentence but incorrectly in another, or generating incorrect translations with slight variations (“Simonioff” and “Simoninov”).

→	SL	PL	Comments
krl-ru	<p>Ранее, генеральный директор <u>Ring</u>, Джейми <u>Симинофф</u>, отметил, что компания получила своё начало от того, что он не услышал, как в его гараже зазвонил звонок из магазина. Он рассказал, что сделал Wi-Fi звонок. <u>Симинофф</u> рассказала, что продажи выросли после того, как она появилась в 2013 году в шоу «Shark Tank», где судьи отказались финансировать её выступление. В конце 2017 года <u>Симинов</u> появился на покупательном канале QVC. Кроме того, <u>Ring</u> заключил соглашение с конкурирующей компанией по обеспечению безопасности ADT Corporation.</p>	<p>Ранее генеральный директор компании <u>Ring</u> Джейми <u>Симинофф</u> отметил, что компания получила своё начало, когда он не услышал звонок в дверь из магазина в своем гараже. Он рассказал, что сделал Wi-Fi дверной звонок. <u>Симинофф</u> сказал, что признание выросло после того, как он появился в 2013 году в шоу Shark Tank, где судьи отказались финансировать его инициативу. В конце 2017 года <u>Симинофф</u> появился на торговом канале QVC. Кроме того, <u>Ring</u> достиг соглашения с конкурирующей компанией безопасности ADT Corporation в судебном процессе.</p>	<p>The PL system preserves the company name and the person's surname across the paragraph. The SL system preserves the company name but translates the person's surname differently; moreover, it changes the person's gender from sentence to sentence.</p>
lud-ru	<p>Раньше генеральный директор <u>Ringo</u>, Джейми <u>Симинов</u>, заметил, что эта фирма ушла, потому что он не услышал звон дверей в своей гаражной мастерской. Он рассказал, как сделал дверной колокольчик с варежкой. <u>Симинов</u> рассказал, что продажи выросли после его выступления в программе «Шаркий танк» в 2013 году, где судьи не стали признавать его стартап. В конце 2017 года <u>Симинов</u> появился на покупном TV-канале QVC. <u>Кинг</u> также договорился о судебном процессе с компанией по охране прав конкурентов ADT Corporation.</p>	<p>Ранее генеральный директор <u>Ring</u> Джейми <u>Симинов</u> заметил, что эта фирма пошла по тому пути, что он не слышал дверные колокольчики в своей гаражной мастерской. Он сказал, что сделал колокольчик с помощью скатерти. <u>Симинов</u> сказал, что продажи выросли после его появления в программе Shark Tank в 2013 году, где судьи не стали понимать его стартап. В конце 2017 года <u>Симинов</u> проиграл дело на QVC-телеканале. <u>Ring</u> договорился также о судебном процессе с компанией по охране конкурентов ADT Corporation.</p>	<p>The PL system preserves the company name and the person's surname across the paragraph. The SL system translates the company name first in Latin, changing it, then in Cyrillic, getting it wrong once again. The person's surname is translated consistently.</p>
vep-ru	<p>Раньше начальник компании «Круг» <u>Зами Симинов</u> заметил, что эта кампания началась, когда он не услышал стукающихся дверей на своем дворе. Он сказал, что сделал Wi-Fi-установку. <u>Симинов</u> сказал, что продажи улучшились, когда он появился в 2013 году в телепередаче «Shark Tank», в которой члены жюри согласились выделить деньги на его проект. В конце 2017 года <u>Симинов</u> появился на передаче QVC. Кроме того, компания «Ринг» подала в суд на своего конкурента – подпольную компанию «ADT Corporation».</p>	<p>Ранее глава компании «<u>Ring</u>» <u>Жами Симинов</u> заметил, что эта кампания началась, когда он не услышал дверной замк на своем автосалоне. Он сказал, что сделал Wi-Fi замк. <u>Симинов</u> сказал, что продажи улучшились, когда он появился в 2013 году в телепередаче «Shark Tank», в которой единогласное жюри решило дать деньги его проекту. В конце 2017 года <u>Симинов</u> появился на телепередаче QVC. Кроме того, компания «<u>Ring</u>» устроила судебные разбирательства со своей конкуренткой – компанией-покровителем «ADT Corporation».</p>	<p>The PL system preserves the company name and the person's surname across the paragraph. The SL system translates the company name in two different ways: first, it is a literal translation (Ring—Круг), then it is a transliteration of the English title (Ring—Ринг). The surname is translated in the same fashion across the paragraph.</p>

Table 3: Translations of the same paragraph from FLORES-200 performed by the sentence-level (SL) MADLAD and paragraph-level (PL) MADLAD in three translation directions, demonstrating the preservation of proper nouns. Underlined with a straight line comes a company name (Ring), underlined with a wavy line comes a person's surname (Siminoff).

	PL	Neurotölge
krl-ru	22.0 \pm 2.0 / 49.1 \pm 1.8	23.4 \pm 2.0 / 50.4 \pm 1.6
lud-ru	19.3 \pm 1.8 / 46.2 \pm 1.5	21.7 \pm 2.0 / 48.2 \pm 1.4
olo-ru	22.1 \pm 2.2 / 48.5 \pm 1.7	25.9 \pm 2.4 / 51.4 \pm 1.8
vep-ru	20.7 \pm 1.8 / 46.3 \pm 1.8	26.5 \pm 2.5 / 51.2 \pm 1.8
ru-krl	13.4 \pm 1.5 / 46.8 \pm 1.3	10.6 \pm 1.3 / 43.5 \pm 1.2
ru-lud	4.0 \pm 1.0 / 33.3 \pm 1.0	3.6 \pm 1.0 / 31.6 \pm 1.2
ru-olo	8.4 \pm 1.4 / 40.6 \pm 1.2	7.0 \pm 1.3 / 36.2 \pm 1.2
ru-vep	12.0 \pm 1.7 / 43.0 \pm 1.5	12.1 \pm 1.5 / 42.9 \pm 1.4

Table 4: Comparison between our paragraph-level (PL) translation system and Neurotölge for translation directions from/into Russian. BLEU and chrF++ scores (separated by slash) of Neurotölge and paragraph-level (PL) MADLAD as evaluated on the paragraph-level Smugri FLORES benchmark.

We also identified several types of errors specific to translations into Russian. For instance, the same word may appear in translation in its original form (“Ring”), as a literal translation into Russian from English (“Крыт”), or as a transliteration into Cyrillic script (“Ринг”). The SL model more frequently combines these three forms inconsistently within the same text compared to the PL model.

The second part of the analysis focuses on coreference resolution, specifically examining the use of pronouns. While many paragraphs in the benchmark dataset mention people, most of them are about men. Both the SL and PL models translated gender-related structures correctly in most cases, typically defaulting to the male gender. However, we found examples where both models struggled with gender, although the PL model made fewer mistakes overall.

To illustrate our findings, we present a paragraph containing examples of lexical cohesion and coreference resolution. Table 3 provides translations of this paragraph generated by the SL and PL systems. It is translated into Russian from Proper Karelian, Ludian, and Veps, with translations from Livvi omitted to save space. English reference of the paragraph is provided below:

Previously, Ring’s CEO, Jamie Siminoff, remarked the company started when his doorbell wasn’t audible from his shop in his garage. He built a WiFi door bell, he said. Siminoff said sales boosted after his 2013 appearance in a Shark Tank episode where the show panel declined funding the startup. In late 2017, Siminoff appeared on shopping television channel QVC. Ring also settled a lawsuit with competing security company, the ADT Corporation.

A detailed explanation of the mistakes made by the systems is presented in Table 3. As we can see, the results highlight the PL model’s ability to effectively handle discourse-level phenomena.

6.3 Comparison with previous results

We compared the results of our paragraph-level model with translations generated by the online machine translation engine Tartu NLP Neurotölge. The online system demonstrates significantly better performance when translating *into* Russian (Table 4). However, our model outperforms Tartu NLP Neurotölge when translating *from* Russian to Proper Karelian and Livvi and shows comparable results for Ludian and Veps. For example, in the Russian→Proper Karelian direction, the PL model beats Neurotölge by 2.8 BLEU or 3.3 chrF++.

6.4 Zero-shot English translation

In this final experiment, we investigated how well the translation abilities of the models transferred to unseen pairs of languages in the example of English. We translated the FLORES-200 benchmark from Proper Karelian, Livvi, Ludian, and Veps to English and back. The results are shown in Table 5.

First, we notice that the original model without fine-tuning already has high scores for translation *into* English. This probably means that the model transferred its knowledge of Finnish and Estonian to their low-resource relatives. Besides, MADLAD-400 has seen much more data in English than in any other language, which may account for the scores being bigger than for zero-shot translation into Russian.

Next, we observe the boost in accuracy after fine-tuning, which tells us that the knowledge has

	base	SL	PL	<i>p</i> -value
krl-en	20.8 ± 2.1 / 49.4 ± 1.5	25.5 ± 1.8 / 53.7 ± 1.3	27.1 ± 2.1 / 55.0 ± 1.4	0.025 / 0.003
lud-en	11.4 ± 1.9 / 37.5 ± 1.8	19.2 ± 1.8 / 47.6 ± 1.4	20.1 ± 2.1 / 48.8 ± 1.5	0.037 / 0.005
olo-en	10.3 ± 1.6 / 36.7 ± 1.7	17.9 ± 1.6 / 45.9 ± 1.4	18.1 ± 1.7 / 46.8 ± 1.4	0.240 / 0.007
vep-en	6.4 ± 1.5 / 31.6 ± 1.8	14.8 ± 1.8 / 43.4 ± 1.5	13.4 ± 1.6 / 42.9 ± 1.4	0.003 / 0.071
en-krl	0.9 ± 0.5 / 4.7 ± 0.8	15.3 ± 1.8 / 48.0 ± 1.4	13.5 ± 1.6 / 46.5 ± 1.3	0.002 / 0.001
en-lud	0.3 ± 0.1 / 2.8 ± 0.4	3.2 ± 1.2 / 31.6 ± 0.9	3.7 ± 1.2 / 32.1 ± 0.9	0.051 / 0.031
en-olo	0.6 ± 0.4 / 3.1 ± 0.5	7.3 ± 1.3 / 37.3 ± 1.1	6.8 ± 1.1 / 36.7 ± 1.1	0.101 / 0.012
en-vep	0.5 ± 0.3 / 3.6 ± 0.5	7.7 ± 1.3 / 37.6 ± 1.3	7.9 ± 1.2 / 37.7 ± 1.3	0.259 / 0.184

Table 5: Zero-shot performance for translation from/into English. BLEU and chrF++ scores (separated by slash) of base MADLAD, sentence-level (SL) MADLAD, and paragraph-level (PL) MADLAD evaluated on the paragraph-level Smugri FLORES benchmark. *p*-value is the probability that SL and PL models are the same with respect to each metric; *p*-value less than 0.05 indicates that the difference between the models is statistically significant (highlighted in bold).

been successfully transferred to the unseen case of English translation. The scores for translation *into* English exceed those for translation *into* Russian and go up to 27.1 BLEU and 55.0 chrF++ in the case of Proper Karelian→English translation. As for the translation *from* English, the scores remain nearly equal to those for translation *from* Russian.

The ratio of capabilities of the sentence-level and paragraph-level models changes from case to case, with both BLEU and chrF++ metrics sometimes indicating the significant superiority of the PL system (Proper Karelian→English, Ludian→English) and sometimes the superiority of the SL system (English→Proper Karelian). As no direct fine-tuning, there is no wonder that the results oscillated so much.

However, the key indicator for us is the ability of the models to handle discourse-level phenomena. As all the languages in question have Latin script, the issue with translating proper names becomes less pronounced. Yet, the distinction between the models is apparent when it comes to gender consistency. For the example explored in Subsection 6.2, the SL model inconsistently shifts gender when translating sentences from any studied language into English. The PL model, unlike the SL, consistently and accurately translates gender across the paragraph for all languages.

7 Conclusion

In this paper, we developed a machine translation system for four low-resource Finno-Ugric languages: Proper Karelian, Livvi, Ludian, and Veps. Unlike previous MT systems that cover the same

languages, ours is paragraph context-aware. The analysis showed that the model consistently translates names and terminology, though, it still encounters difficulties with coreference resolution.

The developed system has been trained only on parallel corpora with Russian. Nevertheless, the system is also capable of translating to and from English, despite not being trained to do so, with paragraph-level abilities being successfully transferred to this case.

Additionally, we presented a FLORES-based benchmark dataset for Proper Karelian (Viena), Ludian, and Veps. The collected paragraph-level corpora are released as HuggingFace scripts that will allow one to re-collect the data.

We leave for future work experiments with more Finno-Ugric languages, including creating a paragraph-level benchmark that enables a more thorough evaluation of discourse-level phenomena handling. It would also be interesting to compare our results with multilingual decoder-only models, as many of these are starting to emerge.

Acknowledgments

This work was partially supported by the Estonian Research Council grant PRG2006 as well as the National Programme of Estonian Language Technology grant EKT67. All computations were performed on the LUMI Supercomputer through the University of Tartu HPC center. The authors also thank the University of Eastern Finland and the Karelian language revival project for their valuable consultation and support.

References

- Rachel Bawden, Rico Sennrich, Alexandra Birch, and Barry Haddow. 2018. Evaluating discourse phenomena in neural machine translation. 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 1304–1313, New Orleans, Louisiana. Association for Computational Linguistics.
- Tatyana Boyko, Nina Zaitseva, Natalia Krizhanovskaya, Andrew Krizhanovsky, Irina Novak, Nataliya Pellinen, and Aleksandra Rodionova. 2022. The open corpus of the veps and karelian languages: Overview and applications. *KnE Social Sciences*, 7(3):29–40.
- Daniel Deutsch, Juraj Juraska, Mara Finkelstein, and Markus Freitag. 2023. Training and meta-evaluating machine translation evaluation metrics at the paragraph level. In *Proceedings of the Eighth Conference on Machine Translation*, pages 996–1013, Singapore. Association for Computational Linguistics.
- Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. 2021. Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48.
- Christian Hardmeier. 2012. Discourse in statistical machine translation: A survey and a case study. *Discours*, (11).
- Christian Hardmeier, Sara Stymne, Jörg Tiedemann, and Joakim Nivre. 2013. Docent: A document-level decoder for phrase-based statistical machine translation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 193–198, Sofia, Bulgaria. Association for Computational Linguistics.
- Amir Hossein Kargaran, Ayyoob Imani, François Yvon, and Hinrich Schuetze. 2023. GlotLID: Language identification for low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6155–6218, Singapore. Association for Computational Linguistics.
- Sneha Kudugunta, Isaac Caswell, Biao Zhang, Xavier Garcia, Derrick Xin, Aditya Kusupati, Romi Stella, Ankur Bapna, and Orhan Firat. 2023. Madlad-400: A multilingual and document-level large audited dataset. In *Advances in Neural Information Processing Systems*, volume 36, pages 67284–67296. Curran Associates, Inc.
- Samuel Läubli, Rico Sennrich, and Martin Volk. 2018. Has machine translation achieved human parity? a case for document-level evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4791–4796, Brussels, Belgium. Association for Computational Linguistics.
- Sameen Maruf and Gholamreza Haffari. 2018. Document context neural machine translation with memory networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1275–1284, Melbourne, Australia. Association for Computational Linguistics.
- Lesly Miculicich, Dhananjay Ram, Nikolaos Pappas, and James Henderson. 2018. Document-level neural machine translation with hierarchical attention networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2947–2954, Brussels, Belgium. Association for Computational Linguistics.
- NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Hefernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia-Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation.
- OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, 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, Hee-

- woo 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 Kopic, 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, Rameesh 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.
- Miikul Pahomov. 2017. Lyydiläiskysymys: kansa vai heimo, kieli vai murre? Ph.D. thesis, Humanistinen tiedekunta, Suomi.
- 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.
- Tommi Pirinen, Francis M. Tyers, Trond Trosterud, Ryan Johnson, Kevin Unhammer, and Tiina Puolakainen. 2017. North-Sámi to Finnish rule-based machine translation system. In Proceedings of the 21st Nordic Conference on Computational Linguistics, pages 115–122, Gothenburg, Sweden. Association for Computational Linguistics.
- Maja Popović. 2017. chrF++: words helping character n-grams. In Proceedings of the Second Conference on Machine Translation, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.
- Matt Post. 2018. A call for clarity in reporting BLEU scores. In Proceedings of the Third Conference on Machine Translation: Research Papers, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.
- Taido Purason, Aleksei Ivanov, Lisa Yankovskaya, and Mark Fishel. 2024. SMUGRI-MT - machine translation system for low-resource finno-ugric languages. In EAMT 2024 Products and Projects track.
- Matiss Rikters, Marili Tomingas, Tuuli Tuisk, Valts Ernštreits, and Mark Fishel. 2022. Machine translation for Livonian: Catering to 20 speakers. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 508–514, Dublin, Ireland. Association for Computational Linguistics.
- Zewei Sun, Mingxuan Wang, Hao Zhou, Chengqi Zhao, Shujian Huang, Jiajun Chen, and Lei Li. 2022. Rethinking document-level neural machine translation. In Findings of the Association for Computational Linguistics: ACL 2022, pages 3537–3548, Dublin, Ireland. Association for Computational Linguistics.
- Francis M. Tyers, Linda Wiecheteck, and Trond Trosterud. 2009. Developing prototypes for machine translation between two Sami languages. In Proceedings of the 13th Annual Conference of the European Association for Machine Translation, Barcelona, Spain. European Association for Machine Translation.
- Dusan Varis and Ondřej Bojar. 2021. Sequence length is a domain: Length-based overfitting in transformer models. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics.
- 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.

Lisa Yankovskaya, Maali Tars, Andre Tättar, and Mark Fishel. 2023. Machine translation for low-resource Finno-Ugric languages. In Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa), pages 762–771, Tórshavn, Faroe Islands. University of Tartu Library.

Evaluating LLM-Generated Explanations of Metaphors – A Culture-Sensitive Study of Danish

Bolette S. Pedersen¹, Nathalie Sørensen³, Sanni Nimb³,
Dorte Haltrup Hansen¹, Sussi Olsen¹, and Ali Al-Laith^{1,2}

Department of Nordic Studies and Linguistics, University of Copenhagen, Denmark¹

Department of Computer Science, University of Copenhagen, Denmark²

The Danish Language and Literature Society, Denmark³

bspedersen@hum.ku.dk, nats@dsl.dk, sn@dsl.dk

dorteh@hum.ku.dk, saolsen@hum.ku.dk, alal@di.ku.dk

Abstract

We examine how well Danish culture-specific metaphors are explained by two of the best performing language models for Danish, ChatGPT and Llama. For comparison, the explanations are measured against how well cross-lingual (or ‘universal’) metaphors are explained by the models; referring here to metaphors that exist in Danish as well as *across* cultures and languages and in particular in English. To perform our study, we compile a pilot dataset of 150 Danish metaphors and idioms divided tentatively by culture specificity. We prompt the two models and perform a careful qualitative evaluation of the explanations against a four-graded scale. Our studies show that both models are heavily biased towards English since they have much more success in explaining the metaphors that also exist in English than the culture-specific ones, relying presumably on erroneous transfer from English when dealing with the latter. In particular, the *sentiment* of the culture-specific metaphors seems to be often ‘lost in translation’. We further claim that this strong colouring towards English poses a serious problem in the era of LLMs with regards to developing and maintaining cultural and linguistic diversity in other languages.

1 Introduction

Metaphorical expressions are an essential part of language and offer considerable cognitive benefits in both oral and written communication by making the content much more personal and engaging (Noveck et al., 2001; Citron and Goldberg, 2014; Prabhakaran et al., 2021). Metaphorical language is therefore also highly frequent and occur

with reference to both universal, culture-specific and personal aspects of life. In other words, metaphors and idiomatic expressions provide an advanced tool for humans to express themselves in abstract and complex situations with reference to highly culture-specific, personal, and opinion-oriented values (Lakoff and Johnson, 1980).

With the recent advancements of large language models (LLMs), however, using metaphors in communication is no longer exclusive to humans. Chatbots like ChatGPT produce and interpret metaphors when they communicate, and they do so with apparent fluency and equilibrium, in particular for English. A more careful look into the use of metaphors in language models, however, exposes quite a lot of serious problems and cultural biases, even if it is hard to pinpoint exactly from where these problems arise. Some may be due to unbalanced training data where some languages are prioritised over others and thereby causing erroneous language transfer and cultural hallucinations (Zhang et al., 2023; Cao et al., 2023) and (Myung et al., 2024). Others may derive from a general lack of grounding of the language models with respect to physical objects and spacial conditions, and therefore a lack of ability to ‘see’ which features from a concrete sense are transferred to the metaphorical meaning; a deficit that may decrease in future with language models becoming increasingly more multi-modal (Szot et al., 2024).

Under all circumstances, cultural biases in the use of and interpretation of metaphors become particularly evident and problematic when working with the models on medium-resourced languages like the Scandinavian ones. Standard techniques for evaluating the language models in terms of large-scale benchmarks that are often both rigid and simplistic in nature do not reveal a fully nuanced picture of how this complex figure of speech is dealt with by the models, as mentioned for Dan-

ish in (Pedersen et al., 2024).

In order to gain better knowledge and understanding of the models’ treatment of metaphor in our own language, we therefore aim at i) compiling a pilot metaphor dataset, which is culturally sensitive in that it is developed from Danish language resources and from the point of perspective of the Danish society, and ii) providing qualitative evaluations by Danish native speakers of the explanations given by the models on these metaphors.

For our study of model performance, we choose the currently two best performing models in Danish according to the Scandeval Benchmark, namely ChatGPT and Llama, as reported in (Nielsen, 2023, 2024). Both chatbots are based on high-performing multilingual transformer models that are well-suited for the kind of conversation on metaphors that we are interested in with our experiments. Where Llama is a partly open-source model, GPT is a proprietary model. We are however only exploring the models via prompting.

The paper is organised as follows: To position our work, we refer in Section 2 to related work on metaphors in linguistics, lexicography, and NLP. Further, in Section 3 we describe the creation of the culture-specific pilot dataset of Danish metaphors, looking into the typical source and target domains reflecting cultural aspects of the Danish society. Section 4 is devoted to our model experiments with ChatGPT and Llama and explains how we have prompted the models about Danish metaphors in both Danish and English and with and without a textual context. We describe in Section 5 our procedure for evaluating the LLM-generated explanations against a four-graded scale and discuss the annotation agreement results. In Section 6 we show and analyse the results and compare how the models deal with culture-specific vs. cross-cultural metaphors, and to which extent the two models differ in performance. All data are made freely available from github ¹. Finally, in Section 7 we conclude and sketch out how our experiments might be scaled up in future work and hopefully used for model improvement.

¹<https://github.com/kuhumcst/danish-semantic-reasoning-benchmark/tree/main/metaphors>

2 Related Work

Metaphors have been studied intensively in linguistic theory for decades and are considered an essential figure of speech that is closely related to our conceptual and cognitive system as well as to our culture. The work of Lakoff and Johnson (Lakoff and Johnson, 1980) constitutes a landmark in this line of research in stating that metaphors are fundamentally a basic means of understanding complex concepts of feelings and abstractions through mappings from more concrete and directly understandable domains. They further underline that the most fundamental values in a culture will be coherent with the metaphorical structure of the most fundamental concepts in the culture.

In recent linguistic studies, focus has further been into getting a deeper understanding of the underlying cognitive processes of metaphors (Bambini et al., 2019), as well as achieving consensus both monolingually and across languages of what constitutes a metaphor, often referred to as the Metaphor Identification Procedure (MIP) (Crisp et al., 2007; Nacey et al., 2019; Sanchez-Bayona and Agerri, 2024). Other works examine how metaphor relates to other figurative figures of speech such as irony, sarcasm, and hyperbole (Bathala et al., 2023), (Burgers et al., 2018).

In lexicography, conventionalised metaphors are typically described as specific word senses and most often also labelled explicitly in the dictionary as figurative/metaphor. In many cases the metaphor is also structurally related to its concrete sense in the form of a subsense/main sense relation. One example is the verb *to splice* in the Oxford English Dictionary (OED.com), where a subsense to the first sense of the verb is described as *In various transferred and figurative uses: To unite, combine, join, mend*. Also Svensk Ordbok (Swedish Dictionary, svenska.se) marks figurative subsenses (as in *fönster* (‘window’) with the label *äv. bildligt* (‘also figurative’)). In other cases, the dictionaries simply mention the figurative meaning as part of the concrete sense description. In the Danish Dictionary (Det Danske Sprog- og Litteraturselskab, 2024), however, metaphors are almost always described as subsenses labelled ‘metaphorical’ or ‘slang’ making them thereby easy to identify and extract for our present study.

Further, recent wordnet studies suggest a lexical metaphor representation, called ChainNet, where

the link to the concrete meaning is highly explicit and where features from the concrete sense that are transferred to the metaphorical sense are described in a systematic way in terms of so-called **feature transformations** (Maudslay et al., 2024).

In NLP, metaphors are also a topic of interest since understanding and representing them is one of the most challenging tasks to deal with in the field. In particular, it has been questioned to which extent LLMs generalise over the metaphorical meanings and represent the reference to the source domain, or whether they memorise them (Pedinotti et al., 2021; Aghazadeh et al., 2022; Wachowiak and Gromann, 2023). Knowledge graphs of metaphorical facts have further been studied as a means to represent the metaphor relations in the models in order to improve performance (Peng et al., 2021).

3 The Danish Pilot Metaphor Dataset

3.1 Single Word Metaphors and Metaphorical Idioms

For our study, we have compiled a pilot dataset comprising 150 Danish metaphors of which 75 are single word metaphors (as in *sejle* ('to sail') and 75 are metaphorical idioms (as in *høste frugterne* meaning 'reap the fruits'). All are extracted from The Danish Dictionary facilitated by the aforementioned main/subsense structure and by information on metaphorical use. In the editing process, most of the senses in the dictionary were assigned a (not published) value from a set of 152 different domain labels established as part of the dictionary project. This underlying information allows us to identify figurative senses within similar source domains such as agriculture and nautical terms. In the case of the metaphorical multiword expressions - which contain no information on domain, neither on the relation to a concrete sense - we rely on the domain information of the central lemmas in the expression.

3.2 Culture-specific vs. Cross-cultural Metaphors

A central aim of our experiment is to develop and test a culturally sensitive dataset of metaphors in Danish since we hypothesise that these may cause specific problems and expose specific weaknesses and bias in the language models. For each of the two types of metaphors (single-word or multi-word), 50 words/idioms were therefore se-

lected for being *culture-specific* to Danish (compared to English). In addition, a smaller set of 25 words/idioms that *do* exist correspondingly in English were selected for comparison. The datasets were validated by two informants who tested the (translated) metaphorical expressions in a network of English native speakers in order to confirm to which extent they are used also in their mother tongue. As commented on by our informants, the task of deciding whether an expression is culture-specific vs. cross-cultural was in fact not always truly binary since several grey-zone examples exist. In several cases approximate expressions do exist in English but not with the *exact* same selection of words from the source domain. In all such cases, however, we chose a restrictive approach and labeled the Danish expressions as culture-specific since there were no exact matches in English.

Such a grey-zone example is the metaphorical use of *studehandel* in Danish referring to a (political) agreement where two parties give a bargain on their overall ideological principles in order to each achieve short-term benefits. The concrete literal translation to English is 'stud trading'; however, a translation of the Danish metaphor into English would rather be 'horse trading' since 'horse' is the animal typically used in English to convey the same kind of agreement. Likewise, *myreflittig* in Danish has the literal translation 'ant diligent', but the corresponding metaphor in English would be 'busy as a bee', i.e. using another insect from the source domain to express a similar if not exactly the same meaning.

3.3 Typical Source and Target Domains of Danish Metaphors

We aim towards representing a selection of specific Danish traits of culture through a number of typical source and target domains of metaphors, i.e. domains which represent central aspects of the Danish society. As for the **source domains**, these include in particular the domain of farming and agriculture as found in examples like *håndplukning* (lit. 'handpicking', fig. 'carefully selecting a specific person for something, for instance a professional position'), *malkning* (lit. 'milking', fig. 'to achieve money or information in a reckless manner'), *gøde jorden*, (lit. 'to fertilise the soil', fig. 'to provide the prerequisites for something to happen'), and *tærskle langhalm*,

(lit. 'thresh long straw', lit. 'to speak too much about the same topic without providing new information') to give just a few. Also related to the old farming community are needle work metaphors as in *rendemaske* ('running stitch') referring in a derogative way to a 'roving person'.

The shipping domain is also central to the Danish self-understanding as represented by a long list of nautical metaphors such as *kæntre* (lit. 'cap-size'), *ballast* (lit. 'ballast'), and *sikker havn* (lit. 'safe harbour') just to mention a few. Interestingly enough, however, our informants made clear that these metaphors have many direct equivalents in English, probably due to the inherent cross-cultural nature of shipping. This goes for the ones mentioned above; an exception though is the term *splejse*, (lit. 'to splice (a rope)') which in Danish refers very specifically to sharing a bill.

Last but not least, a set of miscellaneous domains are represented in our dataset, referring to e.g. animals as in *haj* (lit. 'shark') referring to someone with good skills or *kylling* (lit. 'chicken') referring to someone with a cowardly behaviour. Many animal metaphors exist in a similar way in English, however, often with a slightly different connotation. More clearly culture-specific for Danish are different kinds of miscellaneous foods and artifacts, e.g. using *klejne* (lit. 'twisted cookie') as a reference to money or *koks* (lit. 'coke') as a reference to disorder and chaos.

Finally, it should be noted that in spite of their frequency, bodily anchored metaphors like *tage hånd om noget* (lit. 'take hand around something' meaning 'deal with something') and *få fod på* (meaning 'get a foothold on something') are not prioritised in our dataset since we overall consider them as being quite universal in nature and thus not particularly specific to Danish culture even if the specific lexical choices may differ in many cases.

Regarding the **target domains**, a majority of the selected metaphors are typical conceptual metaphors in the sense that concrete concepts map onto more abstract ones conveying an abstract or mentally complex meaning as seen in e.g. *hønsegård* (lit. 'chicken coop') which in its metaphorical sense reflects an environment characterized by indifferent talk, gossip, pecking order etc. corresponding approximately to the metaphorical meaning of 'barnyard' in English.

Several of the metaphors selected, however,

map an artifact to another artifact, often resulting in a *negative sentiment* of the concrete target as in *havelåge* (lit. 'garden gate') or *skærveknuser* (lit. 'shard crusher') both referring to old creaking bikes; relating indirectly to the fact that bikes are a very common means of transport in Denmark, and that they are not always in a good shape.

In fact, several of the selected metaphors convey a somewhat negative sentiment, presumably referring indirectly to the concept of the famous 'Law of Jante'². This 'law' refers to a strong cultural norm existing particularly in Denmark and Norway that emphasizes humility and collective equality. It basically states that no one should think they are better than others. Examples of metaphors referring to this norm and with a clear negative sentiment are *højbenet* and *højbandet* (lit. 'long-legged' and 'with a high brow') meaning 'being knowledgeable in an arrogant way', *tågehorn* (lit. 'fog horn') referring to somebody who talks a lot in an arrogant and unclear manner, and *flødebolle* ('chocolate candy with a filling of egg white whipped with sugar') referring to a person who is smug and has (too) high thoughts about himself, probably referring back to the fluffy egg-white foam with little substance. *Høj cigarføring* (lit: 'high holding of one's cigar') meaning being self-conscious and arrogant is another such expression.

4 Experiments

4.1 The Models Selected for Experiments

In the following, we describe the two models selected for the experiments.

ChatGPT: To represent ChatGPT, we use the ChatGPT-4o mini model which became available to the public in the ChatGPT web-interface³ in July 2024. The model is trained on data up to October 2023. In the experiments, we used a combination of the web-interface and the API to access the model.

Llama: Llama is represented by the Llama 3.1 405B model. This model has 405 billion parameters and like ChatGPT-4o mini, is trained on data

²The norm was formulated by the Danish-Norwegian author Aksel Sandemose in his novel "A Fugitive Crosses His Tracks" from 1933.

³<https://chatgpt.com>

	Danish	English
Isolated	Hvad er den overførte betydning af ordet/udtrykket X, og hvad har det med X i den almindelige betydning af gøre? Har man den samme metafor på engelsk?	What is the metaphorical meaning of the Danish word/expression X, and what has it got to do with the basic meaning of X? Does the same metaphor exist in English?
In context	Hvad betyder ordet/disse ord X i følgende eksempel: [CONTEXT]?	What does the Danish word/words X mean in the following example: [CONTEXT]?

Table 1: Examples of experimental prompts.

until late 2023. In the experiments, we access the model through the Llama API ⁴.

For both models it holds that information on the exact training data is not public knowledge. From information on previous model versions, we assume, however, that the vast majority of the training data stems from English texts⁵.

4.2 Prompting the Models

We prompt the models using templates. The same templates are used for prompting both models regardless of prompting through a web-interface or API.

We experiment with two types of templates: (1) prompting to explain the metaphor in isolation and (2) prompting to explain the metaphor within a context. Inspired by recent work (Myung et al., 2024), we hypothesise that prompts in English about Danish might provide higher quality explanations than prompts in a less-resourced language like Danish. Thus, we ask both types of prompts in both Danish and English, resulting in four different prompts. The templates for the four prompts are shown in Table 1.

In this setup, we collect a total of 600 responses per model, covering (1) culture-specific and cross-cultural metaphors, (2) metaphors in isolation and within context, (3) prompts (about Danish) in Danish and English, and finally (4) single-word and multi-word metaphors.

5 Qualitative Evaluation of LLM-Generated Explanations

5.1 Evaluation Procedure

All LLM-generated explanations of metaphors were carefully evaluated by a human expert. We used a four-graded scale of evaluation, where a score of '1' denotes a completely wrong explanation of the metaphor with hardly any relevant

points well explained, '2' denotes a wrong explanation but with some relevant points explained, '3' denotes a generally correct explanation but with some irrelevant points mentioned, and '4' is given for a correct explanation comprising a reference to all relevant aspects.

Consider below the example of an explanation of *sejle* ('to sail') given by ChatGPT to which the experts have assigned a score of 1:

The metaphorical use of "sejle" reflects its basic meaning of sailing. Just as sailing involves steering a vessel through different conditions, the metaphor suggests managing one's course through various life situations or challenges. It implies that, like a skilled sailor, a person is successfully navigating through complexities or adversities with a sense of control and confidence.

In fact, the metaphorical meaning in Danish is rather the complete opposite, meaning that everything is chaos as shown in the example *hele huset sejler* (lit. 'the entire house is sailing' meaning that it is a complete mess). The nautical idiom *tage bestik* (lit: 'calculate the course (of a ship)') is another example evaluated with 1, since the explanation is based on a completely wrong literal translation of *bestik*, which can also mean 'cutlery' in Danish.

An example of an explanation evaluated with a 2 is the following with the shipping verb *splejse* ('to splice'):

Metaphorically, "splejse" is used to describe the act of combining or merging different elements, ideas, or people in a way that creates a cohesive unit. For example, if someone talks about "at splejse ideer" (splicing ideas), they mean integrating or combining different ideas to form a more comprehensive or effective concept. Similarly, "at splejse folk" (splicing people) can refer to bringing people together, such as in networking or forming partnerships.

In this case, the merging aspect is correctly explained as a transfer from the concrete handicraft meaning, however, ChatGPT misses the actual meaning of *splejse* in Danish which is used when you pool your money together as a group to buy a gift together for instance, or to collectively pay for a dinner.

⁴<https://www.llama-api.com/>

⁵For ChatGPT 3.5 it was reported that only 0,5 % of the training material was in Danish and more than 50 % in English (www.lex.dk).

Further, an evaluation of 3 is given to the following explanation of *mursten* ('brick', referring in Danish to a big, heavy book):

Metaphorically, "mursten" (brick) can be used to describe something fundamental or essential, akin to how bricks are fundamental components in construction. For example, if someone refers to a concept or an idea as a "mursten," they are suggesting that it is a crucial building block or a key component in a larger structure or system. Additionally, "mursten" can be used in a more specific sense to describe something that is heavy or burdensome. For instance, referring to a lengthy report or a difficult problem as a "mursten" might imply that it is a substantial, challenging task to handle, much like lifting or working with bricks.

Where part of this explanation is completely off (and seems to be inspired by the metaphorical meaning of the English *cornerstone*), ChatGPT does in fact capture the essence of the metaphorical meaning in the last part of the explanation, referring to the example 'a lengthy report'.

Finally, a top score of 4 is given by the experts to the following explanation of metaphorical expression *brænde alle broer* (lit: 'burn all bridges'):

Brænde alle broer bag sig literally means "to burn all bridges behind oneself." The basic meaning is quite straightforward: it refers to the act of destroying or severing all connections, paths, or relationships behind you, making it impossible to return or go back. The metaphorical meaning of this expression is that someone is severing all ties or relationships with their past, making a drastic change, or taking a bold step forward without the possibility of turning back. This can be applied to various situations, such as leaving a job, ending a relationship, or making a significant life decision.

It should be noted that we do not as such evaluate the part of the explanation concerning whether or not the model finds that a similar metaphor exists in English. This part was rather used to get an impression of where erroneous transfer might come from, and also served as an extra check of whether the informants might have overlooked something when judging the universality of the metaphor.

5.2 Annotator Agreement

Five expert evaluators were involved in the evaluation of the LLM-generated explanations. All evaluators are computational linguists, three of them with specific expertise in computational lexicography. To calculate the inter-annotator agreement (IAA), three experts annotated 20% of the

explanations where the models were prompted with a metaphor in isolation (i.e. without a textual context), resulting in a Cohen's Kappa agreement score of 0.475 for the four-graded scale and 0.684 when collapsing the grading into a binary false/true task (i.e. collapsing score 1 and 2 as false and score 3 and 4 as true). Because of the discrepancies and spread in the grading during this first round, the annotation scheme was further discussed and exemplified among the annotators, and two experts subsequently annotated all explanations of metaphors in isolation, resulting this time in a Kappa score of 0.633 on the four-graded scale and 0.857 when seeing the task as binary (true/false). Overall, this can be considered substantial agreement and suggests that despite the inherent subjectivity of the semantic task (based on a relatively open annotation scheme), the annotators demonstrated a robust consensus on the evaluation. Disagreement cases were partly due to diverging assessments when faced with e.g. Danish misspellings or invented words (most typical for Llama), partly due to different opinions on how much to 'punish' wrong or missing bits of explanations.

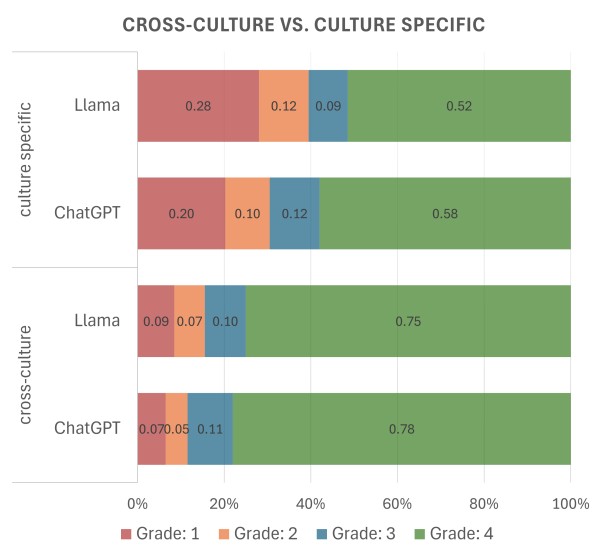


Figure 1: Explanations of Cross-cultural vs. culture-specific metaphors

6 Results and Discussion

As shown in Figure 1, both models have much more success in explaining the metaphors that also exist in English than the culture-specific ones. This indicates that they have too little information on Danish when dealing with the

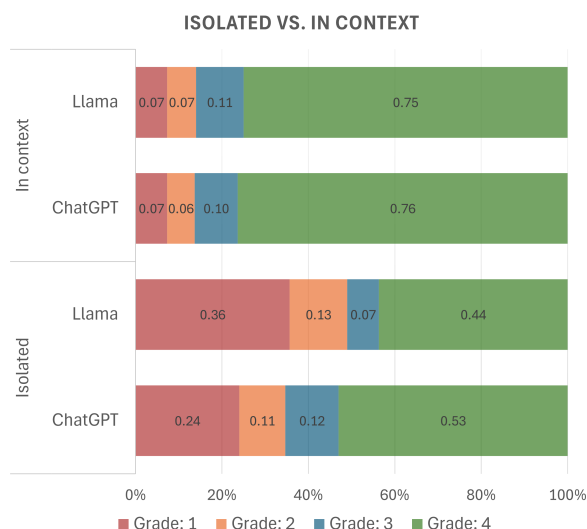


Figure 2: Isolated vs. in context prompts

culture-specific metaphors and therefore hallucinate wrongly (from an English source) in many cases. It can also be seen that the two models differ in quality, with ChatGPT outperforming Llama. For both models, it is clear that when prompted with an example context, they perform remarkably better (Figure 2) than when they are prompted without a context and just asked about the metaphorical meaning. Likewise Figure 4 shows that multiword metaphors are easier to grasp than single word metaphors, presumably due to the multiple words from the expressions themselves. Figure 3 shows that both models perform marginally better when prompted in English about Danish than when prompted in Danish.

Further, we note that metaphors that map from an artifact in the source domain to another artifact in the target domain are generally very hard for the models to predict, and this goes across all experiments. In particular, Llama has a tendency of (amusing) hallucinations, some of which seem to stem from quite simple morphological errors or misspellings. As in the following example where *fis* (lit. 'fart' meaning 'fuss' or 'craziness') from the expression *roterende fis i kasketten* meaning 'being crazy' is changed to *fisk* ('fish'). This error evokes a long story about fish swimming round and round in a cap and becoming stressed by it:

Udtrykkets oprindelse er usikker, men det er sandsynligt, at det kommer fra en gammel forestilling om, at fiskCould godt biputte sig i en kasket eller en anden lille, lukket beholder. Hvis fisken så begyndte at rotere rundt i kasketten, ville det være et tegn på, at den havde mis-

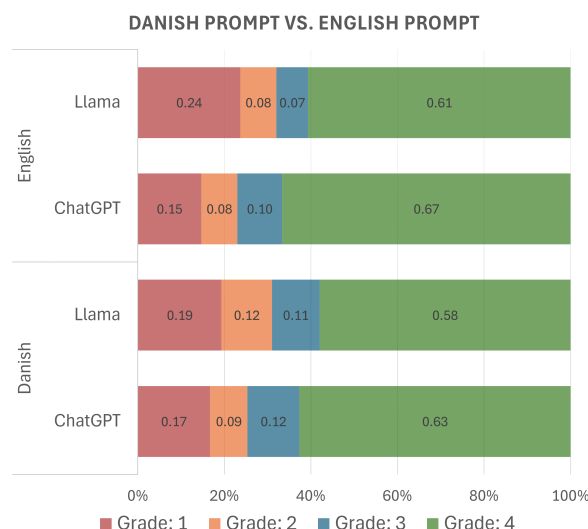


Figure 3: Danish prompts vs. English prompts about Danish metaphors

tet sin normale adfærd og nu var i en tilstand af stress eller panik.

'The origin of the term is uncertain, but it is likely that it comes from an old notion that fish could nestle well in a cap or other small, closed container. If the fish then began to rotate around the cap, this would be a sign that it had lost its normal behavior and was now in a state of stress or panic.'

Another example of hallucination regards the explanation of the concrete meaning of the (in fact cross-cultural) idiom *en nål i en høstak* ('a needle in a haystack') about a task which is really hard, in fact almost impossible. Here Llama hallucinates about women working too close to the haystacks with their needles and therefore missing them in the stack.

I den almindelige betydning af udtrykket handler det om at lede efter en nål, som er blevet tabt i en høstak, hvilket var et almindeligt problem førhen, da kvinder ofte syede eller reparerede tøj nær høstakke. Høstakke var meget store og hvis nålen faldt ud, kunne det være en stor opgave at finde den igen.

'In the common sense of the term, it is about looking for a needle that has been dropped in a haystack, which was a common problem in the past, as women often sewed or repaired clothes near haystacks. Haystacks were very large and if the needle fell out, it could be a big task to find it again'

Another thing that typically goes wrong in both models' explanations, and which is not so well captured by the grading, is the aspect of connotation or sentiment. In quite a lot of cases,

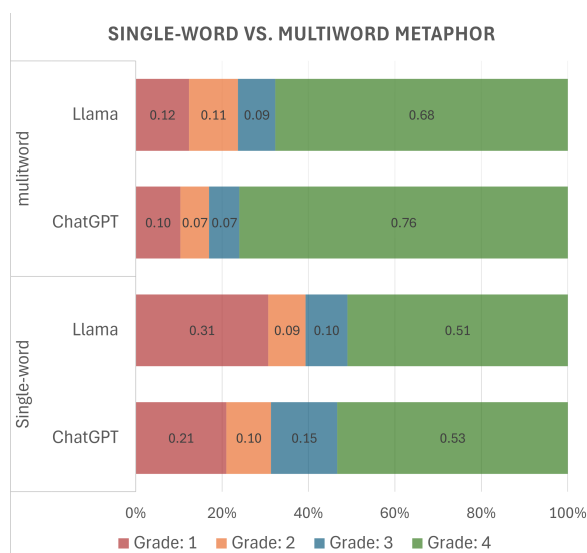


Figure 4: Single word vs. multiword metaphor

the models interpret the metaphors far too positively, missing completely the derogative or negative connotation of the concept and thereby the typical sarcastic Danish 'tone', which can in some contexts be rather harsh. Examples of such misunderstood metaphors, of which several have already been mentioned and explained above, are: *rendemaske* (lit. 'running stitch'), *'rågehorn*, ('foghorn'), *højbenet*, (lit. 'high-legged'), *højpanedet*, ('with a high brow') *sejle*, ('to sail'), *koks* (lit. 'coke'), *hønsegård*, (lit. 'chicken coop'), *havelåge*, (lit. 'garden gate'), and *skærveknuser*, (lit. 'shard crusher'). Likewise with the multiword metaphors *en sang fra de varme lande*, (lit. 'a song from the warm lands' meaning 'an evasive, bland explanation or reply') and *sejle sin egen sø* (lit. 'sailing your own sea' meaning 'be left to yourself; deal with your own problems (as a well-deserved punishment)'). Last but not least, similar things happen with metaphors of sexual connotations, which are completely overlooked or ignored by both models.

6.1 Limitations

Our dataset is relatively limited in size and would be improved by being scaled up. On the other hand, all 1,200 automatically generated explanations were carefully human-evaluated providing thereby an interesting set of nuanced observations regarding the performance of the models. Another limitation relates to the fact that we claim to explore culture-specific vs. cross-cultural metaphors in LLMs without going into the more ethnographic

discussion of what defines a culture and a language community. We have limited ourselves to look into Danish metaphors and compare them with English because we are aware that a majority of the training material used to train the models is in English. From there on we make a general assumption regarding lack of cultural diversity in current high-performing LLMs. Further, some of the cultural characteristics described for Danish may also count for the other Scandinavian communities, while others may not. Some are reflected also in other Northern European countries, whereas some are uniquely Danish. Furthermore, our informants have only involved British native speakers. This may also have caused some unintentional biases in our dataset where some metaphors may or may not exist in American English compared to British English.

7 Conclusions and Future Work

We have compiled a dataset of culture-specific Danish metaphors supplemented with metaphors that are also found cross-culturally, or more specifically between Danish and English. Our aim was to examine how well the two leading chatbots on Danish explain the metaphors and their reference to the source domain, and to which extent we could see a pattern of decline in quality of the explanations deriving from culture-specific expressions that do not have a parallel in English. Our experiments confirm our hypothesis quite strongly. Culture-specific metaphors are highly complex for the models to interpret, and the explanations indicate that erroneous language transfer from English takes place to a large extent, leading to strongly biased and/or hallucinated explanations. In particular, the models have problems in capturing the right sentiment of the metaphors, distorting thereby the specific Danish 'tone of voice'.

Chatbots like ChatGPT are currently rolled out throughout society, in particular through Co-pilot Enterprise, and people are using them for all kinds of tasks. In this context, the strong colouring towards English that we have documented in our work, indicates that the developing and maintaining of cultural and linguistic diversity is under strong pressure, and that the development might move very fast. This tendency is reinforced by the fact that the hallucinations are very well-formulated and on the surface convincing, meaning that only the highly experienced language user

can dismiss garbled output.

To extend our study, we would like to i) expand our Danish dataset, ii) include metaphor studies for the other Scandinavian language, and also iii) go deeper into the understanding of the inner wheels of the models with respect to where the tipping point is found between beneficial language transfer on the one hand and erroneous transfer that leads to cultural biases on the other. One way to proceed in improving the models (in addition to ensuring more Danish training material in the first place) is via fine-tuning or retrieval augmented generation with use of knowledge graphs or other structured information sources. Knowledge graphs can be compiled from culture-specific metaphor lists derived from existing dictionaries or corpora, or from wordnets enriched with feature transformations in a 'ChainNet'-like fashion. All in all enrichments that could potentially lead to better and more culturally diverse language interpretation and generation.

Acknowledgments

Thank you to Birte Pedersen and Guy Norman for acting as informants and for checking in their respective networks to which extent (direct translations of) Danish metaphors and idioms are commonly used in English.

References

- Ehsan Aghazadeh, Mohsen Fayyaz, and Yadollah Yaghoobzadeh. 2022. Metaphors in pre-trained language models: Probing and generalization across datasets and languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2037–2050, Dublin, Ireland. Association for Computational Linguistics.
- Naveen Badathala, Abisek Rajakumar Kalarani, Tejpalsingh Siledar, and Pushpak Bhattacharyya. 2023. A match made in heaven: A multi-task framework for hyperbole and metaphor detection. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 388–401, Toronto, Canada. Association for Computational Linguistics.
- Valentina Bambini, Paolo Canal, Donatella Resta, and Mirko Grimaldi. 2019. Time course and neurophysiological underpinnings of metaphor in literary context. *Discourse Processes*, 56(1):77–97.
- Christian Burgers, Kiki Y Renardel de Lavalette, and Gerard J Steen. 2018. Metaphor, hyperbole, and irony: Uses in isolation and in combination in written discourse. *Journal of Pragmatics*, 127:71–83.
- Yong Cao, Li Zhou, Seolhwa Lee, Laura Cabello, Min Chen, and Daniel Hershcovich. 2023. Assessing cross-cultural alignment between ChatGPT and human societies: An empirical study. In *Proceedings of the First Workshop on Cross-Cultural Considerations in NLP (C3NLP)*, pages 53–67, United States. Association for Computational Linguistics (ACL). Publisher Copyright: © 2023 Association for Computational Linguistics.; 1st Workshop on Cross-Cultural Considerations in NLP, C3NLP 2023; Conference date: 05-05-2023.
- Francesca MM Citron and Adele E Goldberg. 2014. Metaphorical sentences are more emotionally engaging than their literal counterparts. *Journal of Cognitive Neuroscience*, 26(11):2585–2595.
- Peter Crisp, Raymond Gibbs, Alice Deignan, Graham Low, Steen Gerard, Lynne Cameron, Elena Semino, Joe Grady, Alan Cienki, and Zoltán Kövecses. 2007. MIP: A method for identifying metaphorically used words in discourse. *Metaphor and Symbol*, 22(1):1–39.
- Det Danske Sprog- og Litteraturselskab. 2024. Den Danske Ordbog. <https://www.ordnet.dk/ddo>. (September 2024).
- George Lakoff and Mark Johnson. 1980. *Metaphors We Live By*. University of Chicago Press.
- Rowan Hall Maudslay, Simone Teufel, Francis Bond, and James Pustejovsky. 2024. ChainNet: Structured metaphor and metonymy in WordNet. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2984–2996, Torino, Italia. ELRA and ICCL.
- Junho Myung, Nayeon Lee, Yi Zhou, Jiho Jin, Rifki Afina Putri, Dimosthenis Antypas, Hsuvas Borkakoty, Eunsu Kim, Carla Perez-Almendros, Abinew Ali Ayele, Víctor Gutiérrez-Basulto, Yazmín Ibáñez-García, Hwaran Lee, Shamsuddeen Hassan Muhammad, Kiwoong Park, Anar Sabuhi Rzayev, Nina White, Seid Muhie Yimam, Mohammad Taher Pilehvar, Nedjma Ousidhoum, Jose Camacho-Collados, and Alice Oh. 2024. BLEnD: A benchmark for LLMs on everyday knowledge in diverse cultures and languages.
- Susan Nacey, W Gudrun Reijnierse, Tina Krennmayr, and Aletta G Dorst. 2019. *Metaphor Identification in Multiple Languages*. John Benjamins Publishing Company.
- Dan Saattrup Nielsen. 2023. Scandeval: A benchmark for Scandinavian natural language processing. *Proceedings of Nodalida 2023, The Faroe Islands*.
- Dan Saattrup Nielsen. 2024. Status på Scandinavian embedding benchmark (seb). *Slides from Benchmarkworkshop in The Danish Agency for Digital Government, September 20, 2024*.

- Ira A Noveck, Maryse Bianco, and Alain Castry. 2001. The costs and benefits of metaphor. *Metaphor and Symbol*, 16(1-2):109–121.
- Bolette Sandford Pedersen, Nathalie C Hau Sørensen, Sussi Olsen, and Sanni Nimb. 2024. Evaluering af sprogforståelsen i danske sprogmodeller - med udgangspunkt i semantiske ordbøger. *NyS, Ny-danske Sprogstudier*, pages 8–40.
- Paolo Pedinotti, Eliana Di Palma, Ludovica Cerini, and Alessandro Lenci. 2021. A howling success or a working sea? Testing what BERT knows about metaphors. In *Proceedings of the Fourth Black-boxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 192–204.
- Ciyuan Peng, Dang Thinh Vu, and Jason J Jung. 2021. Knowledge graph-based metaphor representation for literature understanding. *Digital Scholarship in the Humanities*, 36(3):698–711.
- Vinodkumar Prabhakaran, Marek Rei, and Ekaterina Shutova. 2021. How metaphors impact political discourse: A large-scale topic-agnostic study using neural metaphor detection. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 15, pages 503–512.
- Elisa Sanchez-Bayona and Rodrigo Agerri. 2024. Meta4xnli: A crosslingual parallel corpus for metaphor detection and interpretation. *arXiv preprint arXiv:2404.07053*.
- Andrew Szot, Bogdan Mazoure, Harsh Agrawal, R De-von Hjelm, Zsolt Kira, and Alexander T Toshev. 2024. Grounding multimodal large language models in actions. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- Lennart Wachowiak and Dagmar Gromann. 2023. Does GPT-3 grasp metaphors? Identifying metaphor mappings with generative language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1018–1032.
- Xiang Zhang, Senyu Li, Bradley Hauer, Ning Shi, and Grzegorz Kondrak. 2023. Don’t trust ChatGPT when your question is not in English: A study of multilingual abilities and types of LLMs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7915–7927, Singapore. Association for Computational Linguistics.

Tokenization on Trial: The Case of Kalaallisut–Danish Legal Machine Translation

Esther Ploeger¹ Paola Saucedo¹ Johannes Bjerva¹
Ross Deans Kristensen-McLachlan² Heather Lent¹

¹Department of Computer Science, Aalborg University

²Department for Linguistics, Cognitive Science, and Semiotics, Aarhus University
{espl, hc1e}@cs.aau.dk

Abstract

The strengths of subword tokenization have been widely demonstrated when applied to higher-resourced, morphologically simple languages. However, it is not self-evident that these results transfer to lower-resourced, morphologically complex languages. In this work, we investigate the influence of different subword segmentation techniques on machine translation between Danish and Kalaallisut, the official language of Greenland. We present the first semi-manually aligned parallel corpus for this language pair¹, and use it to compare subwords from unsupervised tokenizers and morphological segmenters. We find that Unigram-based segmentation both preserves morphological boundaries and handles out-of-vocabulary words adequately, but that this does not directly correspond to superior translation quality. We hope that our findings lay further groundwork for future efforts in neural machine translation for Kalaallisut.

1 Introduction

In contrast to many of the world’s indigenous languages facing challenges in revitalization as a result of colonialism (Meakins and O’Shannessy, 2016), Kalaallisut (West Greenlandic) has a vibrant linguistic ecosystem. Spoken as a first language by people of all ages (Grenoble and Whaley, 2021), Kalaallisut is used in all aspects of daily life by most of the population (Nielsen, 2021), from teenagers texting (Grenoble, 2011) to everyday communication (Ravn-Højgaard et al., 2018). It is also supported by language policies

¹<https://github.com/esther2000/tokenization-on-trial>

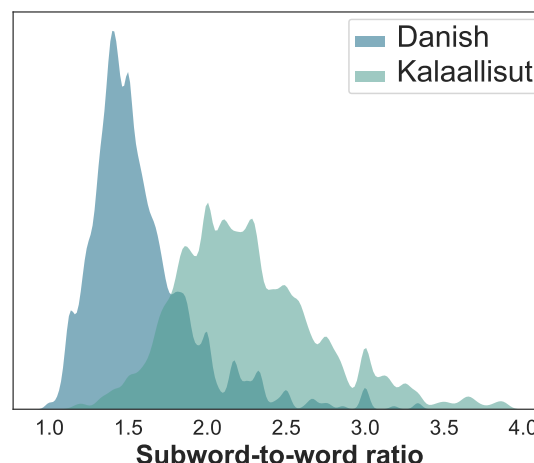


Figure 1: When applying BPE to our test dataset, Kalaallisut generally has higher subword-to-word ratios than Danish; *KDE plot, capped at 4*.

that prioritize its use in education and administration (Møller, 1988; Valijärvi and Kahn, 2020), and boasts a wide range of linguistic resources (e.g., word lists and dictionaries) and existing language technologies (e.g., a spell-checker and a grapheme-to-phoneme converter) from Oqaasileriffik, the Language Secretariat of Greenland.

Despite the vitality of the language, however, Kalaallisut – like most of the world’s languages – does not have sufficient resources for the data-intensive methods of contemporary NLP (Joshi et al., 2020). Specifically in the context of neural machine translation (NMT), Kalaallisut lacks the large-scale aligned parallel corpora required for contemporary machine learning methodologies, and is thus considered a low-resource language. Consequently, Kalaallisut trails behind higher-resourced languages in terms of NMT.

Beyond the limited availability of high-quality parallel corpora, Kalaallisut’s high degree of morphological inflection poses additional challenges for NMT. Commonplace tokenization methods often lead to large, sparse vocabularies for morpho-

logically rich languages (Vylomova et al., 2017; Gerz et al., 2018; Akın Özçift and Söylemez, 2021). To illustrate the difference with a morphologically simple language: Figure 1 shows that BPE tokenization yields many more subwords per word for Kalaallisut than for Danish. It is likely more difficult for models to learn systematic patterns in structure between source and target languages for morphologically rich languages than for morphologically-poor ones (Gutierrez-Vasques et al., 2023). While previous work has compared subword tokenization and segmentation strategies for polysynthetic languages of the Americas (Mager et al., 2022), this work found no single best solution across languages. Thus, NMT for Kalaallisut stands to benefit from a dedicated investigation.

Despite the desire for machine translation by speakers of Kalaallisut (Oqaasileriffik, 2023), however, there is a marked scarcity in research attention (Kristensen-Mclachlan and Nedergård, 2024). As a result, adequate benchmarking datasets for Kalaallisut NMT are extremely scarce.² This work aims to provide practical insights for improved NMT for Kalaallisut, by comparing the efficacy of different segmentation strategies. To this end, we provide the following contributions:

- We present the first semi-automatically aligned Danish–Kalaallisut parallel dataset, in the legal domain;
- We present the first open-science initiative to benchmark NMT from Danish *into* Kalaallisut;
- We compare subwords from four segmentation models and relate the insights to downstream NMT performance;
- We provide discussion and recommendations for future research on Kalaallisut NMT.

Ultimately, we hope that this work can be helpful for the development of open-science NMT systems for Kalaallisut going forward.

2 Background

Greenlandic Language Kalaallisut is the largest member of the Inuit-Yupik-Unangan family. Among the world’s languages, it is one of the more morphologically rich, described

as “*typologically extreme*” in the number and variety of suffixing morphemes available for marking nominal and verbal stems (Fortescue and Olsen, 2022). While effectively segmenting languages with greater morphological complexity is notoriously difficult in NLP (Klavans, 2018b), additional linguistic characteristics of Greenlandic may further complicate subword tokenization. Specifically, De Mol et al. (2020) point to three salient features of Greenlandic morphology: 1) some morphemes are polysemous (*e.g.*, no distinction between present and past tense); 2) unbound morphemes can sometimes be incorporated, resulting in ambiguous morpheme boundaries; and 3) some morphemes undergo phonological changes depending on the subsequent context, in order to avoid illegal morphophonemic sequences. The combination of these factors underscore the utility of a targeted investigation into optimal tokenization strategies for Kalaallisut NMT.

Polysynthesis in NLP In linguistic typology, polysynthesis is a high-level categorization for languages relying heavily on morphological inflection to convey meaning.³ As a result, individual utterances in polysynthetic languages tend to be relatively longer than their non-polysynthetic counterparts. In other words, where non-polysynthetic languages might add a pronoun or preposition, polysynthetic languages incorporate additional morphemes. This results in a kind of *holophrasis*, with a single word encoding both predicate and arguments of a clause within the verb itself (Mithun, 2017).

While polysynthetic languages can be found across the globe (*e.g.*, Quechua in South America and Ainu in Asia), many of them are endangered (Klavans, 2018a), and thus lack representation in NLP (Joshi et al., 2020). Indeed, the fact that most polysynthetic languages are low-resource has meant that the development of language technology for these languages continues to lag behind (Klavans, 2018b). At the same time, polysynthesis brings with it unique challenges for NLP (Es-kander et al., 2019). For example, in the context of NMT, Mager et al. (2018) observe marked information loss between polysynthetic and fusional languages, as a consequence of alignment. Specifically, the NMT systems omit the parts of

²The OPUS collection (Tiedemann, 2009) contains 291 parallel Danish ↔ Kalaallisut samples, most of which consist of a single word.

³It should be noted that, although widely-used across typology, polysynthesis as a proper typological categorization is contested by some linguists (Zúñiga, 2019).

the polysynthetic languages, where morpheme-to-morpheme alignment yielded no equivalent counterpart in the fusional language.

As MT has moved towards neural approaches, subword segmentation has become standard for leveraging large datasets. However, subword tokenizers like BPE have been met with skepticism for polysynthetic languages, as they do not accurately capture morpheme boundaries (Vyloмова et al., 2017; Gerz et al., 2018; Kann et al., 2018; Akın Özçift and Söylemez, 2021; Saleva and Lignos, 2021). In their study on tokenization for polysynthetic languages, Mager et al. (2022) compare BPE versus morphological segmentation across four polysynthetic languages (*i.e.*, Nahuatl, Raramuri, Shipibo-Konibo, and Wixarika). For three languages, morphological segmenters outperform BPE, except for Nahuatl, where BPE yields better results.

MT for Kalaallisut The rich media ecosystem surrounding Kalaallisut means that there exists a reasonable volume of data for the language, compared to many other indigenous languages. Nevertheless, much of this data is not immediately suitable for tasks such as training NMT systems. Taken along with the perceived challenges of working with the language outlined above, this means that MT systems for Kalaallisut have historically relied on rule-based (Oqaasileriffik, 2017) and hybrid approaches (Oqaasileriffik, 2023). Early works for NMT of Kalaallisut were developed in relation to Inuktitut, in attempts to benefit from cross-lingual transfer. Le and Sadat (2020) demonstrate that the use of (bi)character-based and word-based pre-trained embeddings can improve NMT performance for Inuktitut (an indigenous language of eastern Canada), suggesting similar possibilities for other Inuit languages. Nonetheless, the addition of Kalaallisut shows limited usefulness in transfer learning for Inuktitut-English MT thus far (Roest et al., 2020).

More recently, Kristensen-Mclachlan and Nedergård (2024) introduced the first benchmark for Kalaallisut-Danish NMT, containing over 1.2 million words of Kalaallisut and 2.1 million words of parallel Danish translations. However, the authors note limitations related to “crude” sentence level alignment, noting that future data collection efforts are still necessary. In experiments, they use a BiLSTM encoder-decoder architecture with BPE

tokenization (5k, 10k, 30k, and 50k vocabulary size), finding best results with 5k BPE. While the authors discuss potential concerns about subword tokenization for the morphologically rich Kalaallisut, their results demonstrate that BPE is reasonably amenable to the language. Still, they do not experiment with other tokenization strategies.

3 A Reliably Parallel Dataset

Aligning parallel datasets is non-trivial in the case of highly inflectional, low-resource languages. Popular alignment methods require pre-trained language embeddings (Thompson and Koehn, 2019), pre-suppose tokenized text (Varga et al., 2007), or assume that sequence lengths correspond directly across languages (Gale and Church, 1993). Kelly (2020) conducted extensive experiments on alignment of polysynthetic languages, but found that their result for Danish-Kalaallisut was too noisy and thus not useful downstream. Their data was sourced from magazines, however. We hypothesize that choosing a more structured domain (*e.g.*, legal) may make alignment more feasible.

Data Collection Oqaasileriffik referred us to the collection of parallel legal texts, hosted by the Greenlandic Government.⁴ Although Kalaallisut is the most widely spoken language (United Nations, 2023), Greenland’s legal system is bilingual, and laws and legal documents are often drafted in Danish. In this work, we use the *Law Collections*, which is an archive of the legislation of Greenland’s Self-Government, Danish legislation applicable to Greenland, and international regulations that are relevant to Greenland.⁵ In total, it consists of 2,545 publicly available documents in HTML format, originally written between 1908 and 2024, many of which are manually translated.

Alignment and Filtering Through scraping, we retrieve parallel documents, filtering out any non-translated documents. However, to obtain parallel *sentences*, we need to align the text. As mentioned, this assumes data or experimental consensus which is not available for Kalaallisut (*i.e.*, language embeddings and tokenized text). Fortunately, legal text is highly structured: our scraped data contains strict paragraph markers (*e.g.*, § 2)

⁴Available at <https://nalunaarutit.gl>

⁵<https://nalunaarutit.gl/om-nalunaarutit>

and clause enumerations (e.g., *a*)), equally across source and target. We leverage this structure by aligning through enumeration: for each document, we retrieve all enumerated text segments and align accordingly in case of 1:1 correspondence with enumeration markers. As an additional advantage, alignment on enumerated clauses furthermore serves as a filtering step. For example, less-structured introductory texts and sensitive information such as email addresses and full names are automatically filtered out. We strip the enumeration token from each line, apply deduplication and subsequently extract 1,000 lines for the validation set, and 1,000 other lines for the test set. We use the remaining lines as the training set. Importantly, we make the design choice to not remove near-duplicates. Legal texts can be highly formulaic, and since we perform an in-domain evaluation which cannot be expected to be widely generalizable regardless (see: Limitations), we decide to leave them in.

Dataset Size In Table 1, we show the size of the resulting corpus. The dataset consists of more than 40,000 parallel phrases. Unsurprisingly, due to Kalaallisut’s inflections, the number of separate words (whitespace delimited strings of characters, obtained with the `wc -l` command) is much higher for Danish than for Kalaallisut.

Split	# Lines		# Words	
	GL	DA	GL	DA
Training	39,936	39,936	663,734	929,904
Validation	1,000	1,000	16,594	23,021
Testing	1,000	1,000	16,665	23,846
Total	41,936	41,936	696,993	976,771

Table 1: Size of parallel legal text dataset.

While small compared to what is available for high-resource languages, the size of the dataset is larger than that used in a comparable low-resource neural MT study (Mager et al., 2022). It is smaller than the other open, parallel Danish-Kalaallisut dataset (Kristensen-Mclachlan and Nedergård, 2024), but as ours is aligned based on human alignments, we expect that ours includes considerably less noise. This leaves us with a small, but high-quality in-domain dataset for legal translation.

4 Experiments

Since we are interested in isolating the effects of subword segmentation on NMT performance, we train dedicated bilingual MT models from scratch. Our experimental set-up consists of three steps: subword segmentation, machine translation, and evaluation, each described in more detail below.

4.1 Subword Segmentation

We experiment with two types of unsupervised segmentation for Kalaallisut: traditional MT subword tokenizers, and morphological segmenters. Following Mager et al. (2022), we keep the Danish side of the parallel corpus consistent across experiments, as this allows us to isolate the effects of Kalaallisut segmentation. We apply BPE to the Danish text, trained on the Danish training set of our corpus. We use a vocabulary size of 5k, as this was found to be optimal in the three most similar research initiatives (Saleva and Lignos, 2021; Mager et al., 2022; Kristensen-Mclachlan and Nedergård, 2024).⁶

Traditional MT Tokenization Following Mager et al. (2022), we train and apply Byte-Pair Encoding (BPE; Sennrich et al., 2016). Originally introduced as a data compression algorithm (Gage, 1994), the segmenter is trained bottom-up by merging frequently co-occurring vocabulary items. In addition, we experiment with Unigram language modeling (Kudo, 2018). Rather than constructing the vocabulary bottom-up, it starts from the largest vocabulary, which is subsequently pruned. This method has been shown to preserve morphological segmentation better than BPE (Bostrom and Durrett, 2020), making it especially relevant for our study. We use both algorithms as implemented in SentencePiece (Kudo and Richardson, 2018).

Morphological Segmentation Segmenting text according to (predicted) morpheme boundaries may be particularly beneficial for low-resource MT, as a means to counter the data scarcity of co-occurring characters that inflections may introduce. As we do not have a large-scale in-domain annotated dataset of morphological segmentations for Kalaallisut, we are constrained to unsupervised segmenters. Specifically, we follow Saleva and Lignos (2021) in using Morfessor 2.0 (Smit et al.,

⁶For Kalaallisut, we also experimented with vocabulary sizes 1k, 3k, 7k, 9k and 11k, but found no improvement.

Method	Kalaallisut Segmentation		Machine Translation			
	Fertility	% Cont.	Danish→Kalaallisut		Kalaallisut→Danish	
			chrF2	BLEU	chrF2	BLEU
<i>None</i>	1.000	0.00	44.6	3.4	61.5	15.4
BPE	2.294	66.10	56.4	8.7	64.2	21.5
Unigram	2.290	66.27	61.4	10.1	58.9	17.1
Morfessor	1.925	70.72	56.7	7.9	58.3	17.2
FlatCat	1.870	69.30	63.2	9.6	57.0	15.1

Table 2: Comparison of segmentation and translation quality metrics on the Kalaallisut test set.

2014), henceforth simply *Morfessor*. In addition, we use FlatCat (Grönroos et al., 2014), which is an extension over Morfessor that uses a Hidden Markov model. After applying morphological segmentation, we post-process the data such that the SentencePiece output format is replicated (words separated by the underscore symbol, and subwords separated by spaces).

Each of the segmentation methods is trained on the training set and applied to all sets (training, validation, and test set) of the Kalaallisut part of the parallel corpus data only. As a baseline, we add the case of applying no segmentation whatsoever to the Kalaallisut side.

4.2 Machine Translation

We train bilingual NMT models for both translation directions separately, with the Transformer architecture (Vaswani et al., 2017). We use the Fairseq toolkit (Ott et al., 2019). Because of the limited data availability in our scenario, we tailor the hyperparameters to those typically found to be effective in low-resource translation, such as using a higher dropout rate (Sennrich and Zhang, 2019; Araabi et al., 2022). We use a learning rate of 0.0001, cross entropy as a criterion with label smoothing (0.2), and apply a dropout rate of 0.3. Each model is trained for a maximum of 100 epochs, with a patience setting of 5 epochs monitoring the validation loss. For generation we use the best checkpoint.

4.3 Evaluation

Subword Metrics To compare segmentation methods, we use two metrics proposed by Rust et al. (2021): *subword fertility* and *continued word proportion*. Subword fertility is the average number of subwords per word. This metric provides insight into “how aggressively a tokenizer splits”.

The proportion of continued words measures the percentage of words that are divided into more than one subword, indicating how *often* words are split. For “words”, we use the whitespace delimited character strings. Intuitively, lower scores are preferred, as high values signal weak compression efficacy, which could lead to oversegmentation.

Translation Quality For assessing the quality of the output translations, we report the chrF2⁷ (Popović, 2015) and BLEU⁸ (Papineni et al., 2002) scores, as implemented in SacreBLEU (Post, 2018). The ChrF2 metric is especially suitable to our scenario, as it is based on character n-grams. It has been previously been used in the context of low-resource NMT on diverse languages (e.g. Tiedemann, 2020). Due to the low-resourcedness, we do not include evaluation based on language embeddings, such as COMET (Rei et al., 2020), as there are indications that they are not reliable in low-resource scenarios (Falcão et al., 2024). While human evaluation would likely provide a better insight into the usefulness for speakers, the absolute number of native translation professionals is much lower than for many, higher-resourced, language pairs. At the same time, this highlights the need for research into reliable MT systems for Kalaallisut.

4.4 Main Results

Table 2 lists the subwords metrics and downstream MT performance for each of the segmentation methods. It should be noted that the results for Danish→Kalaallisut and Kalaallisut→Danish cannot be compared directly, because of the un-

⁷Signature: nrefs:1|case:mixed|eff:yes|nc:2|nw:0|space:no|version:2.4.3

⁸Signature: nrefs:1|case:mixed|eff:no|tok:13a|smooth:exp|version:2.4.3

even number of character and word n-grams. Instead, systems should be compared column-wise.

First, we observe that not using any segmentation method leads to suboptimal downstream MT results. Especially in the case of Danish→Kalaallisut, performance trails considerably behind that obtained with segmenters. BPE obtains the highest scores for translation into Danish, but this is not the case for translation into Kalaallisut, where both the Unigram and FlatCat approaches obtain higher chrF2 scores.

We do not observe clear patterns as to the subword metrics and MT performance. While the morphological segmenters, Morfessor and FlatCat, obtain the lowest fertility scores, this does seem to correspond directly to higher MT quality. While this corroborates earlier findings (Saleva and Lignos, 2021; Mager et al., 2018), more data points are needed to draw robust conclusions.

5 Analysis

To add more context to our findings, we perform additional analyses.

5.1 Subwords vs. Morphological Boundaries

To what extent do the subword segmenters preserve morphological boundaries? To analyze this, we apply each segmenter to a list of words, for which we have gold-standard annotations. We use the data from De Mol et al. (2020), who compiled a set of Kalaallisut words and phrases, and their morphological segmentations. These annotations originate from courses on Kalaallisut, and were corrected by a native speaker. Their data contains both short (e.g. “*he drinks*”) and long (e.g. “*it can be expected to have been eating jellyfish*”) general-domain examples. Since this is out-of-domain for the trained segmenters, it requires a degree of generalization. In total, we use 499 of these examples for our evaluation. We apply the segmenters to each of these examples, and evaluate the resulting subwords using precision (Eq. 1) and recall Eq. 2).⁹ The F1-score is then calculated as the harmonic mean between the average precision and recall.

$$P = \frac{|\{\text{gold morphemes}\} \cap \{\text{subwords}\}|}{|\{\text{subwords}\}|} \quad (1)$$

$$R = \frac{|\{\text{gold morphemes}\} \cap \{\text{subwords}\}|}{|\{\text{gold morphemes}\}|} \quad (2)$$

⁹Equations adapted from Nouri and Yangarber (2016).

Table 3 contains our results. For all segmenters, we find that morphological boundaries are only preserved modestly, with F1 scores all under 35 percent. The lowest score is found with BPE, with precision, recall and F1 only slightly above 10%. Relating this to the downstream results in Table 2, where best results for translation to Danish were obtained with BPE, it seems that preserving morphemes does not directly lead to optimal downstream NMT performance. This is in line with previous findings (Saleva and Lignos, 2021).

A second observation is that Unigram is (at least) on par with FlatCat and Morfessor when it comes to preserving morphological boundaries. This may be somewhat surprising, as Unigram is not a dedicated morphological segmenter. Yet, given its top-down pruning approach, morphemes are better preserved than with BPE’s bottom-up approach. This is in line with findings from Bostrom and Durrett (2020).

Method	Prec. (%)	Rec. (%)	F1 (%)
BPE	10.81	12.42	11.56
Unigram	30.88	37.68	33.94
Morfessor	31.08	31.61	31.34
FlatCat	29.58	29.40	29.49

Table 3: Comparison of morphological boundaries and subword segmentation.

5.2 Out-of-Vocabulary Words

One of the core motivations for subword segmentation, is that it enables better representations of out-of-vocabulary (OOV) words. This has been argued to improve downstream performance, for instance in the case of MT (Senrich et al., 2016). We explore how prominent OOV words are, when processed with varying segmentation techniques. We report the percentage of unknown items (UNKs) in the test portion of our parallel corpus, as shown in the logs of fairseq-preprocess. The results are listed in Table 4.

First, we observe that applying subword segmentation drastically reduces the number of UNKs. When not applying any segmentation, more than 14% of the words are OOV. This high number reflects Kalaallisut’s highly inflectional characteristics. Moreover, this observation may provide an explanation for why down-

Method	% UNK
None	14.30000
BPE	0.005080
Unigram	0.005050
Morfessor	0.196000
FlatCat	0.295000

Table 4: Proportion of OOV words in the Kalaallisut test set.

stream MT performance, specifically *into* Kalaallisut, lags when not applying any segmentation (Table 2). Secondly, we observe a difference between the morphological segmenters (Morfessor, FlatCat) and the traditional MT tokenizers (BPE, Unigram): using the latter results in fewer UNKs than the former. Notably, it is interesting that Unigram segmentation somewhat preserves morpheme boundaries (Table 3), while also resulting in relatively few UNKs.

6 Discussion and Recommendations

Given the lack of research for Kalaallisut NMT, we posit that collaboration between NLP researchers, Greenlandic language experts, and non-specialist native speakers of the language is crucial. In addition to a parallel dataset and experimental documentation, we aim to contribute to NMT for Kalaallisut by providing some high-level recommendations below.

Explore Additional Resources Beyond the legal domain, Kalaallisut boasts a wealth of traditional linguistic resources, like dictionaries (Berthelsen, 1997) and formal grammars (Fortescue, 1984; Sadock, 2003; Berge, 2011; Kahn and Valijärvi, 2021; Nielsen, 2022). Due to Greenland’s relationship with Denmark, national newspapers and official government resources are often available in both Kalaallisut and Danish, which allows “pseudoparallel” corpora to be compiled through webcrawling (Jones, 2022). Similar efforts could be applied to other domains. Additional digital resources for Kalaallisut include a spell-checker, text-to-speech system, and grapheme-to-phoneme converter (Oqaasileriffik), a hand tagged corpus (Per Langgård and VISL Team), and recent NMT benchmark dataset (Kristensen-Mclachlan and Nedergård, 2024). With the exception of the latter, no previous works make use of this wealth of resources, and thus

practitioners may benefit from their inclusion going forward.

Consider Other Dialects Even among low-resource languages, the majority of research attention is paid to standard language varieties, with the risk that non-standard dialects are left behind (Faisal et al., 2024). This holds true for Greenlandic, where works on non-standard dialects are far outnumbered by those for Kalaallisut. The Greenlandic language contains three main dialects: Kalaallisut (the western dialect, and the standard form), Tunumiisut (spoken in eastern Greenland), and Inuktitun (used in the northern region).¹⁰ While Kalaallisut is predominant, all dialects are vital to understanding Greenland’s linguistic diversity. Only a few grammar books are available for Tunumiisut (Robbe and Dorais, 1986; Menneccier, 1995; Tersis, 2008) and Inuktitun (Fortescue, 1986), however. No NLP datasets have as yet been published, despite their apparent presence on social media. This suggests that the language’s integration into advanced language technologies is still limited (Siminyu et al., 2020), and future works for Greenlandic NLP could thus benefit from curation of resources and experimentation across dialects.

Mind the Historical Context The colonization of Greenland involved Denmark’s efforts to “civilize” the Inuit population, primarily through educational programs aimed at reshaping their culture (Rud, 2009) and “modernization” efforts in the 1950s also prioritized the Danish language (Gad, 2017). These initiatives reflected broader colonial views that objectified Greenlanders based on race, gender, and class (Thisted, 2021). Even after World War II when decolonization began, they were often framed within medical and social research as “controllable subjects” (Rud, 2021). In spite of these pressures, the Greenlandic language remains widely spoken and serves as a symbol of national identity. In 2009, Greenlandic was declared the sole official language, but Danish remains prevalent in the public administration and essential for higher education (Faingold, 2023), and language policy is a recurring debate in Greenlandic politics (Gad, 2017). Despite this progress for the Greenlandic language, the legacy of colonialism still has consequences for indigenous lan-

¹⁰https://en.wikipedia.org/wiki/Greenlandic_language.

guages in NLP, which researchers must face. For in-depth conversations on this topic, we refer readers to Bird (2020) and Mager et al. (2023).

Avoid Extractivism Indigenous people have often been treated as research subjects rather than active participants in decision-making processes, particularly under colonial rule (Guillemin et al., 2016). While Greenland has made strides towards self-government (Kuokkanen, 2017), colonial legacies persist in imaginaries¹¹ of an “empty” Arctic whose resources can be readily exploited and its people trivialized (Hanrahan, 2017). In terms of research, this dynamic is enacted through the extraction of knowledge from marginalized communities for academic or bureaucratic consumption (Gaudry, 2011).

This historical context of exploitation raises ethical concerns about modern data collection practices in NLP. As in previous extractive practices, the potential misuse of data, along with issues surrounding privacy, consent, and bias, mirrors ongoing debates in the field of NLP regarding the ethical implications of data mining (Žliobaitė, 2017; Hassani et al., 2020; Watson and Payne, 2020; Singh, 2020; Rogers et al., 2021; Liu et al., 2023). For an in-depth conversation on extractivism in NLP, we refer readers to Bird (2024).

7 Conclusion

In this paper, we build upon the current state of Danish↔Kalaallisut NMT research, noting a sparsity of benchmarks and open-science experimental groundwork. We then introduce a new semi-manually aligned corpus of parallel legal texts for this language pair. Leveraging this, we conduct systematic experiments on subword segmentation, analyzing the impact of both traditional subword tokenizers (BPE, Unigram), and morphological segmentation (Morfessor 2.0, FlatCat) on downstream NMT performance. While segmentation techniques generally improve translation, we do not find one segmenter that beats the others in all aspects. Ideally, more data and evidence are needed to draw more robust conclusions.

Limitations

In this study, we do not examine any (massively) multilingual MT models. As a result, it is possi-

¹¹The concept of “imaginaries” refers to the collective symbols, ideas, and images that shape a society’s understanding (Taylor, 2004).

ble our work misses out on some of the benefits of transfer learning. However, the goal of this work not to create a new state-of-the-art, but rather investigate the isolated effects of subword solutions, relating to Kalaallisut. Accordingly, the findings in this paper can still serve as a starting point for those who continue this work in the future. Moreover, our work investigates isolated subword segmentation techniques, while segmentation methods are not necessarily mutually exclusive. For example, future work could look into applying BPE after morphological segmentation.

Another limitation of this work is its highly-specific legal domain. On the one hand, leveraging legal texts allows us to avoid extractivism, as these data are not taken from Greenlandic writings with deep cultural significance. On the other hand, the use of legal data can also be criticized as reinforcing colonial systems of authority. To avoid the latter, our work is exploratory in nature, and does not seek to create deployable, culturally-appropriate NMT systems for Greenlandic speakers. Instead, we aim to provide a methodology and results pertaining to segmentation, which can still transferable to works in NMT for Greenlandic, outside of the legal domain.

This work focuses solely on the dominant Kalaallisut dialect of Greenlandic. While the inclusion of more dialects is the subject of increasing awareness in NLP, text for other Greenlandic dialects is not supported by the platform through which we sourced our methodology.

Finally, future work in Kalaallisut MT would hugely benefit from human quality assessment. While we assume that automatic, reference-based metrics can give a decent basic estimate of translation quality, human annotations of translation errors would for example enable more fine-grained analysis.

Acknowledgements

We are thankful to Oqaasileriffik for answering our initial questions, and to the anonymous reviewers for their helpful feedback. EP, JB and HL are funded by the Carlsberg Foundation, under the Semper Ardens: Accelerate programme (project nr. CF21-0454).

References

Fatma Yumuk Akın Özçift, Kamil Akarsu and Cevhernur Söylemez. 2021. *Advancing natural lan-*

- guage processing (nlp) applications of morphologically rich languages with bidirectional encoder representations from transformers (bert): an empirical case study for turkish. *Automatika*, 62(2):226–238.
- Ali Araabi, Christof Monz, and Vlad Niculae. 2022. How effective is byte pair encoding for out-of-vocabulary words in neural machine translation? In *Proceedings of the 15th biennial conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 117–130, Orlando, USA. Association for Machine Translation in the Americas.
- Anna Berge. 2011. *Topic and discourse structure in West Greenlandic agreement constructions*. U of Nebraska Press.
- Christian Berthelsen. 1997. *Grønlandsk dansk ordbog*. Atuakkiorfik, Uddannelsesforl.
- Steven Bird. 2020. Decolonising speech and language technology. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3504–3519, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Steven Bird. 2024. Must NLP be extractive? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Bangkok, Thailand. Association for Computational Linguistics.
- Kaj Bostrom and Greg Durrett. 2020. Byte pair encoding is suboptimal for language model pretraining. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4617–4624, Online. Association for Computational Linguistics.
- Barbera De Mol et al. 2020. A comparison of data-driven morphological segmenters for low-resource polysynthetic languages: A case study of greenlandic.
- Ramy Eskander, Judith Klavans, and Smaranda Muresan. 2019. Unsupervised morphological segmentation for low-resource polysynthetic languages. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 189–195, Florence, Italy. Association for Computational Linguistics.
- Eduardo D Faingold. 2023. Language rights and the law in greenland. In *Language Rights and the Law in Scandinavia: Sweden, Denmark, Norway, Iceland, the Faroe Islands, and Greenland*, pages 241–264. Springer.
- Fahim Faisal, Orevaoghene Ahia, Aarohe Srivastava, Kabir Ahuja, David Chiang, Yulia Tsvetkov, and Antonios Anastasopoulos. 2024. DIALECT-BENCH: An NLP benchmark for dialects, varieties, and closely-related languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14412–14454, Bangkok, Thailand. Association for Computational Linguistics.
- Júlia Falcão, Claudia Borg, Nora Aranberri, and Kurt Abela. 2024. COMET for low-resource machine translation evaluation: A case study of English-Maltese and Spanish-Basque. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 3553–3565, Torino, Italia. ELRA and ICCL.
- Michael Fortescue. 1986. *Inuktun – An Introduction to the Language of Qaanaaq, Thule*. Institut for Eskimologi, University of Copenhagen, Copenhagen.
- Michael Fortescue and Lise Lennert Olsen. 2022. The acquisition of west greenlandic. In *The crosslinguistic study of language acquisition*, pages 111–219. Psychology Press.
- Michael D Fortescue. 1984. *West greenlandic*. Croom Helm London.
- Ulrik Pram Gad. 2017. What kind of nation state will greenland be? securitization theory as a strategy for analyzing identity politics. *Politik*, 20(3).
- Philip Gage. 1994. A new algorithm for data compression. *The C Users Journal*, 12(2):23–38.
- William A. Gale and Kenneth W. Church. 1993. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1):75–102.
- Adam James Patrick Gaudry. 2011. Insurgent research. *Wicazo Sa Review*, 26:113 – 136.
- Daniela Gerz, Ivan Vulić, Edoardo Ponti, Jason Naradowsky, Roi Reichart, and Anna Korhonen. 2018. Language modeling for morphologically rich languages: Character-aware modeling for word-level prediction. *Transactions of the Association for Computational Linguistics*, 6:451–465.
- Lenore A Grenoble. 2011. On thin ice: language, culture and environment in the arctic. *Language Documentation and Description*, 9.
- Lenore A Grenoble and Lindsay J Whaley. 2021. Toward a new conceptualisation of language revitalisation. *Journal of Multilingual and Multicultural Development*, 42(10):911–926.
- Stig-Arne Grönroos, Sami Virpioja, Peter Smit, and Mikko Kurimo. 2014. Morfessor flatcat: An hmm-based method for unsupervised and semi-supervised learning of morphology. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1177–1185.
- M. Guillemin, L. Gillam, E. Barnard, P. Stewart, H. Walker, and D. Rosenthal. 2016. “we’re checking them out”: indigenous and non-indigenous research participants’ accounts of deciding to be involved in research. *International Journal for Equity in Health*, 15.

- Ximena Gutierrez-Vasques, Christian Bentz, and Tanja Samardžić. 2023. [Languages Through the Looking Glass of BPE Compression](#). *Computational Linguistics*, 49(4):943–1001.
- Maura Hanrahan. 2017. Enduring polar explorers’ arctic imaginaries and the promotion of neoliberalism and colonialism in modern greenland. *Polar Geography*, 40(2):102–120.
- H. Hassani, C. Beneki, S. Unger, M. Mazinani, and M. Yeganegi. 2020. [Text mining in big data analytics](#). *Big Data and Cognitive Computing*, 4:1.
- Alex Jones. 2022. Finetuning a kalaallisut-english machine translation system using web-crawled data. *arXiv preprint arXiv:2206.02230*.
- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. [The state and fate of linguistic diversity and inclusion in the NLP world](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.
- Lily Kahn and Riitta-Liisa Valijärvi. 2021. *West Greenlandic: an essential grammar*. Routledge.
- Katharina Kann, Jesus Manuel Mager Hois, Ivan Vladimir Meza-Ruiz, and Hinrich Schütze. 2018. [Fortification of neural morphological segmentation models for polysynthetic minimal-resource languages](#). 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 47–57, New Orleans, Louisiana. Association for Computational Linguistics.
- Kevin Kelly. 2020. An evaluation of parallel text extraction and sentence alignment for low-resource polysynthetic languages.
- Judith L. Klavans. 2018a. [Computational challenges for polysynthetic languages](#). In *Proceedings of the Workshop on Computational Modeling of Polysynthetic Languages*, pages 1–11, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
- Judith L. Klavans, editor. 2018b. *Proceedings of the Workshop on Computational Modeling of Polysynthetic Languages*. Association for Computational Linguistics, Santa Fe, New Mexico, USA.
- Ross Kristensen-Mclachlan and Johanne Nedergård. 2024. [A new benchmark for Kalaallisut-Danish neural machine translation](#). In *Proceedings of the 4th Workshop on Natural Language Processing for Indigenous Languages of the Americas (Americas-NLP 2024)*, pages 50–55, Mexico City, Mexico. Association for Computational Linguistics.
- Taku Kudo. 2018. [Subword regularization: Improving neural network translation models with multiple subword candidates](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.
- Taku Kudo and John Richardson. 2018. [Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing](#). *CoRR*, abs/1808.06226.
- R. Kuokkanen. 2017. ‘to see what state we are in’: First years of the greenland self-government act and the pursuit of inuit sovereignty. *Ethnopolitics*, 16:179 – 195.
- Tan Ngoc Le and Fatiha Sadat. 2020. Revitalization of indigenous languages through pre-processing and neural machine translation: The case of inuktitut. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4661–4666.
- Yu Lu Liu, Meng Cao, Su Lin Blodgett, Jackie CK Cheung, Alexandra Olteanu, and Adam Trischler. 2023. [Responsible AI considerations in text summarization research: A review of current practices](#). In *The 2023 Conference on Empirical Methods in Natural Language Processing*.
- Manuel Mager, Elisabeth Mager, Katharina Kann, and Ngoc Thang Vu. 2023. Ethical considerations for machine translation of indigenous languages: Giving a voice to the speakers. *arXiv preprint arXiv:2305.19474*.
- Manuel Mager, Elisabeth Mager, Alfonso Medina-Urrea, Ivan Vladimir Meza Ruiz, and Katharina Kann. 2018. [Lost in translation: Analysis of information loss during machine translation between polysynthetic and fusional languages](#). In *Proceedings of the Workshop on Computational Modeling of Polysynthetic Languages*, pages 73–83, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
- Manuel Mager, Arturo Oncevay, Elisabeth Mager, Katharina Kann, and Thang Vu. 2022. [BPE vs. morphological segmentation: A case study on machine translation of four polysynthetic languages](#). In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 961–971, Dublin, Ireland. Association for Computational Linguistics.
- Felicity Meakins and Carmel O’Shannessy. 2016. *Loss and renewal: Australian languages since colonisation*, volume 13. Walter de Gruyter GmbH & Co KG.
- Philippe Menecier. 1995. *Le tunumiisut, dialecte inuit du Groenland oriental: Description et analyse*, volume 78. Peeters Publishers.

- Marianne Mithun. 2017. Argument marking in the polysynthetic verb. In *The Oxford Handbook of Polysynthesis*, pages 30–59. Oxford University Press.
- Aquigssiaq Møller. 1988. [Language policy and language planning after the establishment of the home rule in greenland](#). *Journal of Multilingual and Multicultural Development*, 9:177–179.
- Flemming AJ Nielsen. 2021. Literacy and christianity in greenland. In *The Inuit world*, pages 187–206. Routledge.
- Flemming AJ Nielsen. 2022. *Vestgrønlandsk grammatik*. BoD–Books on Demand.
- Javad Nouri and Roman Yangarber. 2016. [A novel evaluation method for morphological segmentation](#). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 3102–3109, Portorož, Slovenia. European Language Resources Association (ELRA).
- Oqaasileriffik. 2017. [Nutserut: The pre-2023 method](#). Accessed: 2024-10-14.
- Oqaasileriffik. 2023. [Nutserut: Hybrid artificial intelligence](#). Accessed: 2024-10-14.
- The Language Secretariat of Greenland Oqaasileriffik. [Resources - oqaasileriffik](#). Accessed: 2024-10-14.
- Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. [fairseq: A fast, extensible toolkit for sequence modeling](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. 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.
- Per Langgård and VISL Team. [Hand-tagged closed corpus for greenlandic - panola project](#). Accessed: 2024-10-14.
- Maja Popović. 2015. chrF: character n-gram f-score for automatic mt evaluation. In *Proceedings of the tenth workshop on statistical machine translation*, pages 392–395.
- Matt Post. 2018. [A call for clarity in reporting BLEU scores](#). In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.
- Signe Ravn-Højgaard, Ida Willig, Mariia Simonsen, Naja Paulsen, and Naimah Hussain. 2018. *Tusagas-siuitit 2018: en kortlægning af de grønlandske medier*. Ilisimatusarfik.
- Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. [COMET: A neural framework for MT evaluation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.
- Pierre Robbe and Louis-Jacques Dorais. 1986. *Tunumiit oraasiat = Tunumiut oqaasii = Det østgrønlandske sprog = The East Greenlandic Inuit language = La langue inuit du Groenland de l’Est*. Université Laval, Centre d’études nordiques, Québec.
- Christian Roest, Lukas Edman, Gosse Minnema, Kevin Kelly, Jennifer Spenader, and Antonio Toral. 2020. Machine translation for english–inuktitut with segmentation, data acquisition and pre-training. In *Fifth Conference on Machine Translation*, pages 274–281. Association for Computational Linguistics (ACL).
- Anna Rogers, Timothy Baldwin, and Kobi Leins. 2021. [‘just what do you think you’re doing, dave?’ a checklist for responsible data use in NLP](#). In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4821–4833, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Søren Rud. 2009. [A correct admixture: The ambiguous project of civilising in nineteenth-century greenland](#). *Itinerario*, 33:29 – 44.
- Søren Rud. 2021. [Governing sexual citizens: decolonization and venereal disease in greenland](#). *Scandinavian Journal of History*, 47:567 – 586.
- Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. [How good is your tokenizer? on the monolingual performance of multilingual 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 (Volume 1: Long Papers)*, pages 3118–3135, Online. Association for Computational Linguistics.
- Jerrold M. Sadock. 2003. *A Grammar of Kalaallisut (West Greenlandic Inuttut)*. LINCOM Europa. 21 cm.
- Jonne Saleva and Constantine Lignos. 2021. [The effectiveness of morphology-aware segmentation in low-resource neural machine translation](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 164–174, Online. Association for Computational Linguistics.

- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. [Neural machine translation of rare words with subword units](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.
- Rico Sennrich and Biao Zhang. 2019. [Revisiting low-resource neural machine translation: A case study](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 211–221, Florence, Italy. Association for Computational Linguistics.
- Kathleen Siminyu, Sackey Freshia, Jade Abbott, and Vukosi Marivate. 2020. [Ai4d–african language dataset challenge](#). *arXiv preprint arXiv:2007.11865*.
- J. Singh. 2020. [Natural language processing for studying consumer journey: a case study of sneaker shoppers](#). *International Journal of Research in Science and Technology*, 10:98–101.
- Peter Smit, Sami Virpioja, Stig-Arne Grönroos, and Mikko Kurimo. 2014. [Morfessor 2.0: Toolkit for statistical morphological segmentation](#). In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 21–24, Gothenburg, Sweden. Association for Computational Linguistics.
- Charles Taylor. 2004. *Modern Social Imaginaries*. Duke University Press, Durham, NC.
- Nicole Tersis. 2008. *Forme et sens des mots du tunumiisut: lexique inuit du Groenland oriental*. CNRS Editions, Paris.
- Kirsten Thisted. 2021. [Un-shaming the greenlandic female body](#). *On the Nude*.
- Brian Thompson and Philipp Koehn. 2019. [Vecalign: Improved sentence alignment in linear time and space](#). 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 1342–1348, Hong Kong, China. Association for Computational Linguistics.
- Jörg Tiedemann. 2009. *News from OPUS - A Collection of Multilingual Parallel Corpora with Tools and Interfaces*, volume V, pages 237–248.
- Jörg Tiedemann. 2020. [The tatoeba translation challenge – realistic data sets for low resource and multilingual MT](#). In *Proceedings of the Fifth Conference on Machine Translation*, pages 1174–1182, Online. Association for Computational Linguistics.
- United Nations. 2023. [Visit to denmark and greenland - report of the special rapporteur on the rights of indigenous peoples](#). United Nations General Assembly, Human Rights Council, A/HRC/54/42/Add.2.
- Riitta-Liisa Valijärvi and Lily Kahn. 2020. The linguistic landscape of nuuk, greenland. *Linguistic Landscape*, 6(3):265–296.
- Dániel Varga, Péter Halácsy, András Kornai, Viktor Nagy, László Németh, and Viktor Trón. 2007. Parallel corpora for medium density languages. *Amsterdam Studies In The Theory And History Of Linguistic Science Series 4*, 292:247.
- 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.
- Ekaterina Vylomova, Trevor Cohn, Xuanli He, and Gholamreza Haffari. 2017. [Word representation models for morphologically rich languages in neural machine translation](#). In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pages 103–108, Copenhagen, Denmark. Association for Computational Linguistics.
- K. Watson and D. Payne. 2020. [Ethical practice in sharing and mining medical data](#). *Journal of Information Communication and Ethics in Society*, 19:1–19.
- Indrè Žliobaitė. 2017. Measuring discrimination in algorithmic decision making. *Data Mining and Knowledge Discovery*, 31(4):1060–1089.
- Fernando Zúñiga. 2019. [Polysynthesis: A review](#). *Language and Linguistics Compass*, 13(4):e12326. E12326 LNCO-0774.

The Roles of English in Evaluating Multilingual Language Models

Wessel Poelman and Miryam de Lhoneux

Department of Computer Science

KU Leuven, Belgium

{wessel.poelman, miryam.delhoneux}@kuleuven.be

Abstract

Multilingual natural language processing is getting increased attention, with numerous models, benchmarks, and methods being released for many languages. English is often used in multilingual evaluation to prompt language models (LMs), mainly to overcome the lack of instruction tuning data in other languages. In this position paper, we lay out two roles of English in multilingual LM evaluations: as an *interface* and as a *natural language*. We argue that these roles have different goals: *task performance* versus *language understanding*. This discrepancy is highlighted with examples from datasets and evaluation setups. Numerous works explicitly use English as an interface to boost task performance. We recommend to move away from this imprecise method and instead focus on furthering language understanding.

1 Introduction

With the increase of in-context, prompt-based evaluation of auto-regressive languages models (LMs, Brown et al., 2020), choices have to be made on how prompts are created. Specifically in multilingual evaluation, a crucial choice is in which language(s) prompts are written. In practice, English tends to be mixed with a target language with the explicit goal of increasing *task performance*. We argue this goal is different from furthering *language understanding*. In this position paper, we outline two roles of English at the core of this discrepancy and their implications.

Several works have highlighted methodological issues in multilingual evaluation setups (Artetxe et al., 2020; Ploeger et al., 2024). The dominance of English in natural language processing (NLP) has also been discussed repeatedly (Joshi et al.,

2020; Ruder et al., 2022). With the increase of prompt-based evaluations of models, a new issue has appeared: English being used as an *interface*, rather than a *natural language*.

In recent work, Zhang et al. (2023) propose a taxonomy of prompt-based multilingual LM evaluations. They conclude that “[the model] achieves higher performance when the task is presented in English.” This finding is consistent among a large number of papers (Shi et al., 2022; Huang et al., 2022; Fu et al., 2022; Lin et al., 2022; Asai et al., 2024; Etxaniz et al., 2024, inter alia). Resorting to using English like this is hardly surprising given that instruction tuning datasets are expensive to create and not readily available for most languages. Less surprising still is the finding that English performs well, as it is included in virtually all LMs. It does bring into question: what is being evaluated and what do we learn from this?

To illustrate: MaLa-500 (Lin et al., 2024) is a Llama 2-based model (Touvron et al., 2023) that underwent continued pre-training in over 500 languages. It is partially evaluated on a news topic classification task using SIB-200 (Adelani et al., 2024a), a dataset of (*sentence*, *topic*) pairs in 205 languages. The model is prompted as follows:

The topic of the news {sentence} is {topic}

Using the prompt with a Turkish¹ example gives:

The topic of the news Bu oteller günün zenginlerinin ve ünlülerinin kalacağı yerlerdi ve çoğu zaman kaliteli yemeklere ve gece hayatına sahipti. is entertainment

This format is used across all 205 languages in few-shot setups from one to ten. This mixture of English and a target language is, arguably, not very ‘natural’. We refer to this role of English as an *interface*, rather than a *natural language*. In the next sections, we outline these roles and why they are important to consider in multilingual evaluation.

¹English translations of examples are in Appendix A.

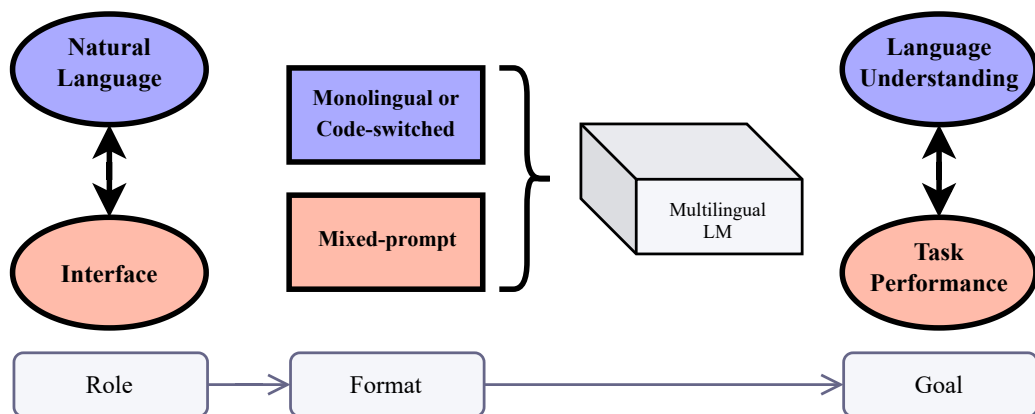


Figure 1 – Schematic overview of the different roles of English in multilingual LM evaluation.

2 Evaluation Goals

Language understanding. We take the common perspective that evaluation concerns a *task* which is used as a proxy for *understanding*. This is exemplified by the *natural language understanding* (NLU) label many datasets and models adhere to (including SIB-200). A news topic classification task shows that the model (arguably) ‘understands’ some of the differences between news categories. A model that rewrites, translates or summarizes ‘understands’ both task instructions and target passages. In a multilingual setting, the understanding of interest is *generalizability* across languages; a model performing a task in a target language supposedly *understands* something about that language. This is then applied to multiple languages. We refer to this as ‘multilingual natural language understanding’ (MLU). Specifically, we use MLU to mean ‘understanding a target language is part of multilingual natural language understanding.’²

Understanding English by itself and understanding a *natural* mix of English and another language are both part of MLU. The latter enters the domain of code-switching: the phenomenon where a speaker fluently switches between multiple different languages during the same conversational turn (Milroy and Muysken, 1995).³

The MaLa-500 prompt mixes English and a target language. However, it is hard to classify this as code-switching, as the switch is hardly natural, es-

pecially in a few-shot setup. Rather than a *natural language* that tells something about *language understanding*, English is used as an *interface* to the LM with the goal of increasing *task performance*. We refer to this mixing as a *mixed-prompt*.

Task performance. Another widespread perspective on evaluation in (multilingual) NLP considers performance on a task as an end in itself.⁴ If we want to classify news topics in a practical application operating in a multilingual setting, what a model supposedly understands or how well it models a particular language is of little value. What matters is the system performing its task adequately across languages. Without using English, the system might not even work at all. This is a common justification; mixing in English is arguably better than not having a system at all.

While practical, this perspective is seemingly at odds with the many tasks and datasets that present themselves under the aforementioned label of *language understanding*. Additionally, task performance as the sole goal introduces a usability issue. Auto-regressive LMs are increasingly meant to be directly interacted with (a *natural* language interface). If we have to resort to a mixed-prompt for the system to even function, it means the user has to be able to write English and get familiar with this unnatural mixing of languages.

Figure 1 summarizes our argument and terminology. Next, we provide more details regarding the discrepancies between using English as an interface versus using it as a natural language.

²We are aware this (ab)use of terminology is not standard.

³Some differentiate between code-switching and code-mixing, we do not make a distinction. For an overview of code-switching in NLP, we refer to Winata et al. (2023).

⁴We thank two reviewers for suggesting to put more emphasis on this perspective.

3 Evaluation Methods

As mentioned in §1, a large body of contemporary research in multilingual NLP focuses on prompting methods. Common evaluation setups range from (i) prompts fully in a target language, to (ii) English instructions with task-specific passages in the target language, to (iii) translating all text into English before presenting it to a model.⁵ None of these works refer to this mixture as being code-switched text. All conclude that a mixture of English and a target language (a mixed-prompt) generally results in the best task performance. In this section we show why a mixed-prompt is an inherently imprecise method to use in evaluation, even if maximizing task performance is the goal.

If we use a prompt fully in a target language, we are clearly evaluating part of MLU. A mixed-prompt introduces *additional factors* that are evaluated that are neither the task nor MLU. We illustrate this from two angles: the representation of the prompt and fortuitous issues from unnaturally mixing English and a target language.

Consider how to evaluate a multilingual masked language model on the news classification task. A classification layer is added to a pre-trained model to predict the topic labels; it sees label *indices* that are consistent across languages. The labels are language-agnostic for the model (i.e., detached from natural language). The evaluation method and goal are clear: mapping a target language sequence to one of these indices. There are no additional signals influencing this process.

In a prompting setup, the representation of the labels can either be language-agnostic (numbers, letters, symbols, etc.), or not (English words, target language words, etc.). These options result in any number of *tokens*, which will have different representations within the model, unless specifically accounted for. In many multilingual evaluation prompts, the classification labels are English words (such as in the MaLa-500 example). Without target language words or (to an extent) language-agnostic labels, the evaluation method and goal will be inherently imprecise.

In addition to the different representation, more than just the task is evaluated with a mixed-prompt setup. To illustrate this, consider the following setup from the AfriMMLU subtask of IrokoBench (Adelani et al., 2024b):

⁵We do not further discuss ‘translate everything’ as this resembles evaluating English as a *natural language*.

```
You are a highly knowledgeable and intelligent
artificial intelligence model answers multiple-choice
questions about {subject}
Question: {question}
Choices:
A: {choice1}
B: {choice2}
C: {choice3}
D: {choice4}
Answer:
```

The prompt and subject are always in English, the question and choices in the target language. With this setup, more is tested than just a task in a target language:

- Code-switching, if this is considered natural, or unnatural ‘mixed-prompt’ switching.
- Script-switching, if the target language uses a non-Latin script (which applies to Amharic in IrokoBench, using the Ge’ez script).
- Instruction following in English.
- Grammatical error correction in English.⁶
- Answering high-school level exam questions in the target language.

With these mixed-prompts, we arguably do not test MLU, as that would entail a native target language prompt. At the same time, we test more than just the task, even though that is the explicit goal of using English in this way.

While we only discussed classification tasks until now, our argument also applies to other types of tasks. Consider the following zero-shot machine translation prompt from Hendy et al. (2023):

```
Translate this sentence from {source} to {target}
Source: {source_sentence}
Target:
```

The prompt is always in English, the source and target are English words referring to the languages, and the source_sentence is in the target language. Filled in, it looks like this:

```
# DE → NL
Translate this sentence from German to Dutch
Source: Du gehst mir auf den Keks
Target:

# NL → DE
Translate this sentence from Dutch to German
Source: tijd voor een bakje koffie
Target:
```

⁶We have notified the AfriMMLU authors about this. The typo is in the prompt in the paper and in the *lm-evaluation-harness* (Biderman et al., 2024), which is used to obtain their results: https://github.com/EleutherAI/lm-evaluation-harness/blob/7882043b4ee1ef9577b829809c2f4970b0bdba91/lm_eval/tasks/afriMMLU/direct/utils.py.

The authors mention they “*explore prompt selection strategies along two dimensions: quality and relevance*”, but do not mention target language prompts. To underline the *interface* role of English: it is neither the translation source nor target here. Hendy et al. (2023) mention that “*keeping the prompt format the same allows us to potentially leverage the benefits of the underlying instruction finetuning protocol to the full extent.*” This makes explicit the goal of *task performance*. Prompting a model to translate a sentence is easily done in a manner that more closely aligns with the goal of MLU, does not use English, and is closer to natural code-switching:

```
# DE → NL (Dutch speaker)
Wat betekent “Du gehst mir auf den Keks” in het Nederlands?
```

```
# NL → DE (Dutch speaker)
Hoe zeg je “tijd voor een bakje koffie” in het Duits?
```

4 Why does this matter?

Interacting with computers in a natural manner is arguably the ultimate goal of numerous sub-fields of computer science. Work on natural language interfaces to information systems dates back decades (Winograd, 1972; Waltz, 1978). LMs bring us ever closer to this goal. However, in a multilingual setting, it is important to consider what *natural language* is, what is being evaluated, and what promises are sold. Next, we outline the implications of the *interface* versus *natural language* roles on evaluation practices.

Interface. Let us start with the role in which English is akin to a programming language.⁷ We need an interface to communicate with a system, in a way the system can understand. We have seen that mixed-prompts are used to get the system to perform better on a given task. Given the scarcity of instruction tuning datasets and the costs involved in creating these, it is understandable that this is a common (albeit sometimes implicit) perspective. English becomes the ‘programming’ language that glues target language passages together and makes the system perform a task. Programming languages also predominantly use English labels for their keywords. However, if the keyword for a `while` loop happens to be `mientras` or `kjsdfk` is irrelevant for its function. These

⁷Also reflected in this famous post: <https://x.com/karpathy/status/1617979122625712128>

are natural language-agnostic as the meaning (as interpreted by a compiler or interpreter) does not change. Variable names and keywords can be chosen arbitrarily.⁸ This is not the case with prompting, which is sensitive to slight changes, both in English (Sclar et al., 2023) and multilingual setups (Zhang et al., 2023; Asai et al., 2024).

Additionally, evaluation setups that use English as an interface introduce knowledge leakage from English to the target language. This is, again, with the explicit goal of improving task performance.⁹ Being able to understand English instructions is not the same as being able to understand target language instructions. If English truly was a programming language, this would not matter, as the meaning of the instructions would be separate from the meaning of the target language passages. Given that English is a natural language, this *de facto* means more is evaluated than just the task. Consequently, such evaluations are imprecise at best, as shown in §3.

Prompt-based evaluations should extend MLU to the *instruction* domain. A mixed-prompt setup claiming to test “*multilingual understanding*” might more accurately be described as “*understanding English instructions interleaved with passages from target language(s), albeit not in a natural code-switching setup.*”

Natural language. When we consider the other role of English in multilingual prompt-based evaluation, we should treat it the same as any other language. The ‘Multilingual Exemplars’ setup from Shi et al. (2022) is a creative interpretation of this perspective. In this few-shot setup, the model sees various examples, all in *different* languages. The final question is asked in the target language. A setup like this extends the definition of ‘multilingual language understanding’ to the extreme. It becomes harder to interpret what a multilingual model knows about any individual language in this context, but English is certainly not an interface, it is a natural language like all others.

A less extreme setup would simply use native, target language prompts or natural code-switched prompts. This is costly, but it aligns much bet-

⁸Within the specifications of the programming language.

⁹Knowledge leakage also explicitly happens in parameter sharing (Zeman and Resnik, 2008) or cross-lingual transfer (Philippy et al., 2023). However, these methods are fundamentally different from mixed-prompts as they (i) treat English as a natural language, and (ii) target knowledge sharing at the training or finetuning phase, not the evaluation phase.

ter with the goal of multilingual natural language understanding. Indeed, several works specifically explore this direction (Köpf et al., 2023; Singh et al., 2024). This approach clearly tests multilingual language understanding, including the instruction domain. If performance on a particular task in a particular language is lagging behind, or not working at all, it means focus should be put on addressing the core of these issues (e.g., data or modeling). Ideally, we should not resort to imprecise methods to boost task performance.

5 Conclusion

In this position paper we outline two roles of English in multilingual language model evaluation: as an *interface*, with the goal of *task performance*, and as a *natural language*, with the goal of *language understanding*. We (i) list works that incorporate English with the explicit goal of boosting task performance, even in tasks such as translation where it is neither the source nor target, underlining the *interface* role, (ii) show that mixing English with a target language in a *mixed-prompt* is unnatural (i.e., not code-switching), and (iii) outline why the interface role is an imprecise choice when evaluating multilingual language understanding of language models.

Additionally, we argue that using a mixed-prompt tests *more* than just performance on a certain task. Because English is a natural language and not a programming language, using it in a mixed prompt will inherently lead to fortuitous factors such as (un)natural switching between languages or scripts, grammatical error correction, and more. This all results in imprecise or misleading evaluations, even if the ultimate goal was to evaluate and improve task performance.

We finally contrast the implications of the two roles on evaluation practices. We recommend to move away from using English as an interface in multilingual evaluations and ultimately advocate for the goal of *language understanding*.

Acknowledgments

WP is funded by a KU Leuven Bijzonder Onderzoeksfonds C1 project with reference C14/23/096. We thank the LAGoM-NLP group at KU Leuven for valuable paper recommendations and Mahdi Dhaini for reviewing an early draft of this paper. We also thank the reviewers for their constructive comments.

References

- David Adelani, Hannah Liu, Xiaoyu Shen, Nikita Vassilyev, Jesujoba Alabi, Yanke Mao, Haonan Gao, and En-Shiun Lee. 2024a. [SIB-200: A Simple, Inclusive, and Big Evaluation Dataset for Topic Classification in 200+ Languages and Dialects](#). In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 226–245.
- David Ifeoluwa Adelani, Jessica Ojo, Israel Abebe Azime, Jian Yun Zhuang, Jesujoba O. Alabi, Xuanli He, Millicent Ochieng, Sara Hooker, Andiswa Bukula, En-Shiun Annie Lee, Chiamaka Chukwuneke, Happy Buzaaba, Blessing Sibanda, Godson Kalipe, Jonathan Mukiibi, Salomon Kabongo, Foutse Yuehgoh, Mmasibidi Setaka, Lolwethu Ndolela, Nkiruka Odu, Rooweither Mabuya, Shamsuddeen Hassan Muhammad, Salomey Osei, Sokhar Samb, Tadesse Kebede Guge, and Pontus Stenetorp. 2024b. [IrokoBench: A New Benchmark for African Languages in the Age of Large Language Models](#). *arXiv preprint, arXiv:2406.03368v1*.
- Mikel Artetxe, Sebastian Ruder, Dani Yogatama, Gorka Labaka, and Eneko Agirre. 2020. [A Call for More Rigor in Unsupervised Cross-lingual Learning](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7375–7388.
- Akari Asai, Sneha Kudugunta, Xinyan Yu, Terra Blevins, Hila Gonen, Machel Reid, Yulia Tsvetkov, Sebastian Ruder, and Hannaneh Hajishirzi. 2024. [BUFFET: Benchmarking Large Language Models for Few-shot Cross-lingual Transfer](#). 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 1771–1800.
- Stella Biderman, Hailey Schoelkopf, Lintang Sutawika, Leo Gao, Jonathan Tow, Baber Abbasi, Alham Fikri Aji, Pawan Sasanka Ammanamanchi, Sidney Black, Jordan Clive, Anthony DiPofi, Julen Etxaniz, Benjamin Fattori, Jessica Zosa Forde, Charles Foster, Jeffrey Hsu, Mimansa Jaiswal, Wilson Y. Lee, Haonan Li, Charles Lovering, Niklas Muennighoff, Ellie Pavlick, Jason Phang, Aviya Skowron, Samson Tan, Xiangru Tang, Kevin A. Wang, Genta Indra Winata, François Yvon, and Andy Zou. 2024. [Lessons from the Trenches on Reproducible Evaluation of Language Models](#). *arXiv preprint, arXiv:2405.14782v2*.
- 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.
- Julen Etxaniz, Gorka Azkune, Aitor Soroa, Oier Laccalle, and Mikel Artetxe. 2024. [Do Multilingual Language Models Think Better in English?](#) In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 550–564.
- Jinlan Fu, See-Kiong Ng, and Pengfei Liu. 2022. [Polyglot Prompt: Multilingual Multitask Prompt Training](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9919–9935.
- Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. [How Good Are GPT Models at Machine Translation? A Comprehensive Evaluation](#). *arXiv preprint, arXiv:2302.09210v1*.
- Lianzhe Huang, Shuming Ma, Dongdong Zhang, Furu Wei, and Houfeng Wang. 2022. [Zero-shot Cross-lingual Transfer of Prompt-based Tuning with a Unified Multilingual Prompt](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11488–11497.
- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. [The State and Fate of Linguistic Diversity and Inclusion in the NLP World](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293.
- Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Minh Nguyen, Oliver Stanley, Richárd Nagyfi, Shahul Es, Sameer Suri, David Alexandrovich Glushkov, Arnav Varma Dantuluri, Andrew Maguire, Christoph Schuhmann, Huu Nguyen, and Alexander Julian Mattick. 2023. [OpenAssistant Conversations - Democratizing Large Language Model Alignment](#). In *Thirty-Seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- Peiqin Lin, Shaoxiong Ji, Jörg Tiedemann, André F. T. Martins, and Hinrich Schütze. 2024. [MaLA-500: Massive Language Adaptation of Large Language Models](#). *arXiv preprint, arXiv:2401.13303v2*.
- Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O’Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. 2022. [Few-shot Learning with Multilingual Generative Language Models](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9019–9052.
- Lesley Milroy and Pieter Muysken, editors. 1995. *One Speaker, Two Languages: Cross-Disciplinary Perspectives on Code-Switching*. Cambridge University Press.
- Fred Philippy, Siwen Guo, and Shohreh Haddadan. 2023. [Towards a Common Understanding of Contributing Factors for Cross-Lingual Transfer in Multilingual Language Models: A Review](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5877–5891.
- Esther Ploeger, Wessel Poelman, Miryam de Lhoneux, and Johannes Bjerva. 2024. [What is “Typological Diversity” in NLP?](#) In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5681–5700.
- Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2022. [Square One Bias in NLP: Towards a Multi-Dimensional Exploration of the Research Manifold](#). In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2340–2354.
- 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](#). In *The Twelfth International Conference on Learning Representations*.
- Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2022. [Language models are multilingual chain-of-thought reasoners](#). In *The Eleventh International Conference on Learning Representations*.
- Shivalika Singh, Freddie Vargus, Daniel D’souza, Börje Karlsson, Abinaya Mahendiran, Wei-Yin Ko, Herumb Shandilya, Jay Patel, Deividas Mataciunas, Laura O’Mahony, Mike Zhang, Ramith Het-tiarachchi, Joseph Wilson, Marina Machado, Luisa Moura, Dominik Krzemiński, Hakimeh Fadaei, Irem Ergun, Ifeoma Okoh, Aisha Alaagib, Oshan Mudannayake, Zaid Alyafeai, Vu Chien, Sebastian Ruder, Surya Guthikonda, Emad Alghamdi, Sebastian Gehrmann, Niklas Muennighoff, Max Bartolo, Julia Kreutzer, Ahmet Üstün, Marzieh Fadaee, and Sara Hooker. 2024. [Aya Dataset: An Open-Access Collection for Multilingual Instruction Tuning](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11521–11567.
- 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, Wenying 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 preprint, arXiv:2307.09288v2*.

David L. Waltz. 1978. [An English language question answering system for a large relational database](#). *Communications of the ACM*, 21(7):526–539.

Genta Winata, Alham Fikri Aji, Zheng Xin Yong, and Thamar Solorio. 2023. [The Decades Progress on Code-Switching Research in NLP: A Systematic Survey on Trends and Challenges](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 2936–2978.

Terry Winograd. 1972. [Understanding natural language](#). *Cognitive Psychology*, 3(1):1–191.

Daniel Zeman and Philip Resnik. 2008. [Cross-Language Parser Adaptation between Related Languages](#). In *Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages*.

Xiang Zhang, Senyu Li, Bradley Hauer, Ning Shi, and Grzegorz Kondrak. 2023. [Don’t Trust ChatGPT when your Question is not in English: A Study of Multilingual Abilities and Types of LLMs](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7915–7927.

A Examples

The examples containing Turkish, Dutch or German are repeated here with English translations.

SIB-200 (sample 755):

The topic of the news Bu oteller günün zenginlerinin ve ünlülerinin kalacağı yerlerdi ve çoğu zaman kaliteli yemeklere ve gece hayatına sahipti. is entertainment

The topic of the news *These hotels were where the rich and the famous of the day would stay, and often had fine dining and nightlife.* is entertainment

Interface translation examples:

DE → NL
Translate this sentence from German to Dutch
Source: Du gehst mir auf den Keks
Target:

DE → NL
Translate this sentence from German to Dutch
Source: *You’re getting on my nerves*
Target:

NL → DE
Translate this sentence from Dutch to German
Source: tijd voor een bakje koffie
Target:

NL → DE
Translate this sentence from Dutch to German
Source: *time for a cup of coffee*
Target:

Natural translation examples:

DE → NL (Dutch speaker)
Wat betekent “Du gehst mir auf den Keks” in het Nederlands?

DE → NL (Dutch speaker)
What does “Du gehst mir auf den Keks” mean in Dutch?

NL → DE (Dutch speaker)
Hoe zeg je “tijd voor een bakje koffie” in het Duits?

NL → DE (Dutch speaker)
How would one say “tijd voor een bakje koffie” in German?

Revisiting Projection-based Data Transfer for Cross-Lingual Named Entity Recognition in Low-Resource Languages

Andrei Politov^{*†}, Oleh Shkalikov^{*}, René Jäkel, Michael Färber

Center for Scalable Data Analytics and Artificial Intelligence (ScaDS.AI),

TU Dresden, Dresden/Leipzig, Germany

{andrei.politov, oleh.shkalikov, rene.jaekel, michael.farber}@tu-dresden.de

Abstract

Cross-lingual Named Entity Recognition (NER) leverages knowledge transfer between languages to identify and classify named entities, making it particularly useful for low-resource languages. We show that the data-based cross-lingual transfer method is an effective technique for cross-lingual NER and can outperform multi-lingual language models for low-resource languages. This paper introduces two key enhancements to the annotation projection step in cross-lingual NER for low-resource languages. First, we explore refining word alignments using back-translation to improve accuracy. Second, we present a novel formalized projection approach of matching source entities with extracted target candidates. Through extensive experiments on two datasets spanning 57 languages, we demonstrated that our approach surpasses existing projection-based methods in low-resource settings. These findings highlight the robustness of projection-based data transfer as an alternative to model-based methods for cross-lingual named entity recognition in low-resource languages.

1 Introduction

Named Entity Recognition is well-studied in Natural Language Processing (NLP), but remains a challenge for low-resource languages due to the lack of manual annotation (Pakhale, 2023). Of the roughly 7,000 languages spoken worldwide, most are low-resource, with over 2,800 endangered (Eberhard et al., 2020). Cross-lingual approaches present a promising solution to address the scarcity of labelled data in these languages.

Cross-lingual NER methods can be categorized into *model transfer* and *data-based transfer* approaches (García-Ferrero et al., 2022). *Model transfer approaches* depend on the ability of multi-lingual models to convey task-specific knowledge across languages. *Data-based methods* automate labelling through translation and annotation projection processes while leveraging advancements in multi-lingual language models to enable zero-shot cross-lingual transfer. This approach allows models trained in high-resource languages to identify and classify named entities in other languages without additional annotated data. Additionally, categorization can be done through two approaches: *translate-test*, which labels original sentences in zero-shot settings, and *translate-train*, which generates labelled data to train a NER model.

Here we contribute to the field of cross-lingual NER by demonstrating the effectiveness of a data-based cross-lingual transfer method that achieves comparable and, in some cases, higher performance of multilingual language models in low- and extremely low-resource language scenarios.

Our work focuses on the projection phase of cross-lingual NER pipelines, introducing two improvements to projection-based methods. First, we propose a method specifically designed to improve word-to-word alignments. Second, we present a novel formalized projection approach of matching source entities with extracted target candidates. The proposed methods support *translate-train* and *translate-test* setups, achieving performance on par with model-based cross-lingual transfer techniques while offering greater flexibility. We evaluated our approach using the XTREME (Ruder et al., 2023) and MasakhaNER2 (Adelani et al., 2022) datasets comprising 57 languages in total in *translate-test* settings. The source code and the evaluation results are provided in the GitHub repository¹.

^{*}These authors contributed equally.

[†]Corresponding author.

¹<https://github.com/Cross-Lingual-NER/Projection-Data-Transfer-Cross-Lingual-NER>

2 Related Work

Model transfer methods leverage the ability of models to transfer task-specific knowledge across languages. For example, multilingual models like mBERT (Devlin et al., 2019) and XLM-RoBERTa (Conneau et al., 2020) are trained on high-resource languages and applied to low-resource languages without modification. Torge et al. (2023) demonstrated improved performance when models were fine-tuned on labelled data or pre-trained on a related language. However, low-resource languages often lack sufficient data, and transfer quality diminishes when applied to very different target languages.

Data-based methods employ labelled datasets, often available in high-resource languages, to perform labelling tasks in the target language. They include fully artificial data generation, like MulDA (Liu et al., 2021), and annotation projection methods. This paper focuses on the latter, which typically involves three steps: (i) translating the original sentence from the target (low-resource) to the source (high-resource) language, (ii) applying a NER model to the translated sentence, and (iii) projecting the labels back to the original sentence. While translation and NER use established models such as BERT (Devlin et al., 2019), many methods have been developed for the projection step.

The first major group (Yang et al., 2022; García-Ferrero et al., 2023; Parekh et al., 2024; Le et al., 2024) of projection methods is based on back-translation, where labelled source sentences or their parts are translated back to the target language, preserving the labels. EasyProject (Chen et al., 2023) is a translate-train method that employs the insertion of special markers, specifically square brackets, around source entities. The marked sentence is then passed to the translation model, which independently translates the entire sentence and each source entity. Afterwards, fuzzy string matching is used to project labels: for each substring in the back-translated sentence surrounded by markers, the method identifies the highest fuzzy match for the corresponding translation of the source entity and assigns the appropriate label.

Another type of projection method is based on word-to-word alignments (García-Ferrero et al., 2022; HWA et al., 2005; Tiedemann, 2015; Fei et al., 2020; Schäfer et al., 2022; Poncelas et al., 2023). The general idea is to compute word-to-word correspondence between words of a labelled

sentence in a source language and an original sentence in a target language. The entity’s label is projected onto target words that align with any of the entity’s words. García-Ferrero et al. (2022) have shown that using contextualized neural network-based aligners such as SimAlign (Jalili Sabet et al., 2020) or AWESoME (Dou and Neubig, 2021) is significantly more beneficial than statistical alignment tools like FastAlign (Dyer et al., 2013), but still can produce wrong alignments and therefore lead to projection errors.

3 Methodology

Our proposed approach focuses on projection-based methods that involve word-to-word alignments. We present two improvements (see Figure 1) to existing methods which are intended to be useful for languages that are under-presented in pre-trained language models. Firstly, we investigate an alternative alignment direction to address the known issue of word-to-word alignment quality. Secondly, we reformulate the annotation projection task as a bipartite matching problem between source entities and target candidates, using alignment-based matching scores to formalize the problem and eliminate reliance on heuristics, thereby facilitating method extension.

3.1 Alignment Direction

In projection-based pipelines, errors can arise at all three stages, diminishing the quality of resulting labels. Handling errors caused by forward translation and source NER models can be challenging. We aim to address projection errors caused by incorrect alignments.

Our approach involves computing word-to-word alignments between the original sentence and its back-translated labelled counterpart in the target language (i.e., target-to-target alignments see Figure 1 a). This method is motivated by the expectation that aligning words within the same language is easier than across different languages. This is particularly relevant for low-resource target languages, which often differ significantly from high-resource source languages.

Preserving entities during back-translation is crucial for projecting entities with the use of word alignments between original and back-translated sentences. To achieve this, we employed EasyProject (Chen et al., 2023) as outlined in the previous section.

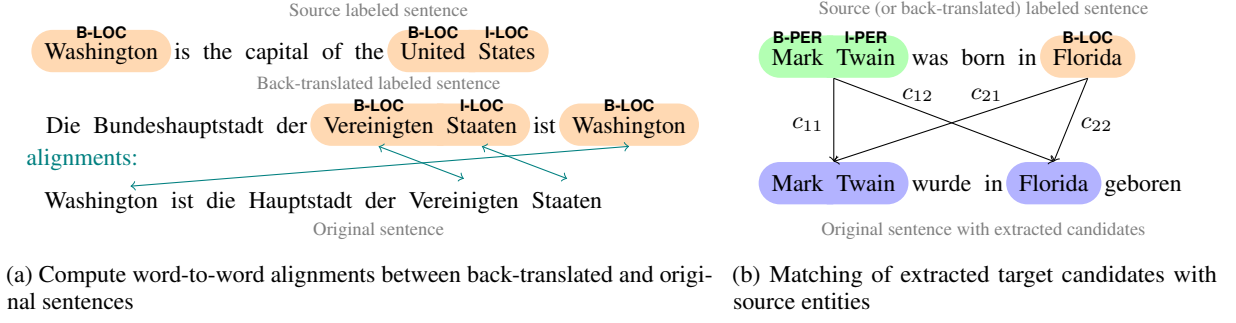


Figure 1: Proposed improvements to projection-based cross-lingual NER methods

3.2 Candidate Matching

The existing methods for addressing problems caused by incorrect alignments such as split annotation, annotation collision and wrong projection fully rely on heuristics (García-Ferrero et al., 2022). We consider that the main reason for these issues is a lack of information about any possible entity candidates in the original sentence in a target language. Instead, we propose to generate target entity candidates and match source entities with candidates by solving the weighted bipartite matching problem with additional constraints.

Let S be a set of source entity spans and T a set of target candidate spans. Then $x_{p^{src}, p^{tgt}}$ is a binary variable which represents whether a source entity $p^{src} \in S$ is being projected to a target candidate $p^{tgt} \in T$. Then the source entity-target candidate matching problem can be formulated as follows:

$$\begin{cases} \max_x \sum_{p^{src}, p^{tgt} \in S \times T} c_{p^{src}, p^{tgt}} x_{p^{src}, p^{tgt}} \\ x_{p_1} + x_{p_2} \leq 1, \quad [i_{p_1}^{tgt}, j_{p_1}^{tgt}] \cap [i_{p_2}^{tgt}, j_{p_2}^{tgt}] \neq \emptyset \\ \sum_{p^{tgt} \in T} x_{p^{src}, p^{tgt}} \leq 1, \quad \forall p^{src} \in S \\ x_{p^{src}, p^{tgt}} \in \{0, 1\}, \quad \forall (p^{src}, p^{tgt}) \in S \times T \end{cases} \quad (1)$$

where $p^{tgt} = (i_p^{tgt}, j_p^{tgt}) \in T$ is a candidate span represented as an index of the starting and the ending word, c is a score of matching. The first set of constraints represents that it is prohibited to project one or several different source entities to the overlapped candidates. The second ensures that all source entities will be projected.

The generation of target candidates is carried out with either N-grams- based or NER model-based candidate extraction. The former considers all continuous word sequences as candidates, while the latter predicts the candidate's spans using a multi-lingual NER model (ignoring predicted classes).

To calculate scores c from Equation 1 of matching between source entities and target candidates word-to-word alignments are being used:

$$c_{p^{src}, p^{tgt}} = \frac{a_{p^{src}, p^{tgt}}}{j_p^{src} - i_p^{src} + j_p^{tgt} - i_p^{tgt}} \quad (2)$$

where $a_{p^{src}, p^{tgt}}$ is a number of aligned words between a source entity and a target candidate. The motivation under this cost is to align entities and candidates based on the count of aligned words, considering source and target lengths to avoid matching with candidates with a lot of nonaligned words and handle single-word misalignments.

The complexity of the proposed problem remains an open question. Notably, it is not a straightforward instance of the maximum weight full bipartite matching problem, which can be solved in polynomial time, due to the first set of constraints that prevents projections onto overlapping candidates (i.e. some projections are mutually exclusive). In NER model-based candidate extraction, where no overlapping candidates exist, the problem reduces to a maximum weight bipartite matching.

To solve the problem in a general formulation, we propose a greedy approximate algorithm, which iteratively selects the projection with the maximum non-zero matching cost, performs this projection, and excludes all candidates that overlap with the projected candidate as well as the projected source entity.

The proposed concept of target candidate extraction and matching is structurally similar to T-Projection by García-Ferrero et al. (2023), with two key differences. T-Projection uses a fine-tuned T5 model, limiting target languages and producing candidates absent in the original sentence. For matching, T-Projection employs NMTScore by Vamvas and Sennrich (2022), while we use word-to-word alignments.

4 Experiments

We performed an intrinsic evaluation of the efficiency of our approaches across a total of 57 languages using the XTREME (Hu et al., 2020) (39 languages) and MasakhaNER2 datasets (excluding Ghomálá and Naijá languages due to limitations in translation model support - 18 languages in total). This evaluation encompasses the full pipeline, considering both translation and source NER model performance.

For a comparative analysis of existing and proposed approaches, we (re)implemented the aforementioned projection methods according to their original papers. In particular, we reimplemented the heuristic word-to-word alignment-based approach outlined by García-Ferrero et al. (2022). We enhanced this heuristic by introducing a word count ratio threshold of 0.8 to better handle misaligned unitary words. Additionally, we reimplemented the EasyProject method, which performs back-translation of labelled source sentences, using original, fine-tuned by authors, NLLB-200-3.3B² model. This back-translated output is then used for annotation projection, relying on word-to-word alignments computed between the original and labelled back-translated sentences in the same language (denoted as *tgt2tgt*).

NLLB200-3.3B³ (Costa-jussà et al., 2022) was employed as a translation model for all experiments. The XLM-R-Large model⁴, fine-tuned on the English split of the CONLL2003 (Tjong Kim Sang and De Meulder, 2003), served as both the source model and for target candidate extraction, as well as for model transfer experiments. We ignored MISC entities predicted by this model in the first set of experiments since this class does not exist in the MasakhaNER2 and XTREME datasets. For computing word-to-word alignments, we used the original implementations of SimAlign and non-finetuned AWESOME neural aligners with the default settings (with MBERT model).

As the evaluation involved full pipelines, the resulting metrics were influenced by both translation quality and the performance of the NER models. To ensure a fair and consistent comparison of the proposed methods, we employed the same models for translation and source labelling throughout all

experiments. For tasks involving the proposed integer linear programming (ILP) formulation of the projection problem, we utilized the previously described greedy approximation algorithm to derive solutions.

Evaluation results for the full pipelines are given in Table 1.

As shown in Table 1, candidate matching methods consistently deliver a strong performance. The proposed approach involving n-gram candidates extraction (*n-gram cand.*), compared to heuristics (since n-gram does not limit a set of candidates as NER cand. do), provide comparable or superior results while offering greater flexibility and avoiding hyperparameter optimization.

The NER model-based extraction (*NER cand.*) generally outperforms model transfer by effectively correcting labels for correctly predicted spans, resulting in greater accuracy particularly when model transfer mislabels these spans. It also surpasses the n-gram approach and achieves results comparable to model transfer because of more fine-grained candidates.

The model transfer generally performs better on the XTREME dataset, but candidate matching methods surpass heuristic approaches in most of the 36 languages, except for Bengali, Kazakh, and Swahili. The first may happen due to the model’s exposure to these languages or their partial representations during pretraining, despite being fine-tuned only on English data.

Although the average score for the MasakhaNER2 dataset is modest, the proposed method performs better than heuristics in 10 languages and worse in 8 out of 18 total languages. The full list can be found in the appendix. This discrepancy may be attributed to the simpler morphological structures in the first group (where proposed methods perform better), while the second group, especially languages like Xhosa and Zulu (Maho, 1999), presents greater morphological complexity, including noun class systems and agreement patterns.

The proposed method with target-to-target alignment direction generally does not outperform the source-to-target method, except for Japanese, due to errors introduced during back-translation, highlighting a potential area for future research.

Additional experiments, described in the appendix, evaluate the performance of the projection step independently. Table 2 shows projection per-

All models are from the HF Hub

²ychenNLP/nllb-200-3.3B-easyproject

³facebook/nllb-200-3.3B

⁴FacebookAI/xlm-roberta-large-finetuned-conll03-english

Approach	Align. dir.	XTREME					MasakhaNER2		
		yo	bn	et	fi	avg	bam	twi	avg
Model transfer	-	32.3	37.8	67.7	72.1	50.9	43.0	46.4	52.1
Heuristic SimAlign	src2tgt	32.8	36.9	56.6	59.1	41.8	49.3	71.8	66.6
	tgt2tgt	20.1	34.8	39.9	50.8	34.9	44.3	5.9	42.7
Heuristic AWESoME	src2tgt	33.3	38.8	56.0	58.6	41.5	49.7	74.6	67.3
	tgt2tgt	15.2	34.1	40.7	50.4	34.6	43.4	3.8	42.1
n-gram cand. SimAlign	src2tgt	29.7	36.5	60.9	62.5	43.3	48.9	69.5	66.4
	tgt2tgt	17.5	36.4	43.4	52.5	36.4	42.5	5.2	42.4
n-gram cand. AWESoME	src2tgt	28.8	36.5	60.0	61.7	42.3	48.3	70.9	66.7
	tgt2tgt	16.1	35.2	43.8	52.0	35.9	41.1	3.9	42.0
NER cand. SimAlign	src2tgt	52.0	38.3	58.6	61.2	46.4	55.3	69.3	63.0
	tgt2tgt	34.0	30.7	43.1	53.4	39.1	46.9	8.5	44.0
NER cand. AWESoME	src2tgt	50.2	38.2	58.0	60.5	45.9	55.0	69.1	62.5
	tgt2tgt	27.7	30.7	42.4	52.6	38.5	46.4	6.8	43.3

Table 1: F1 scores for various full pipelines and alignment directions on XTREME (first section) and MasakhaNER2 (second section). *Heuristic SimAlign* and *Heuristic AWESoME* are heuristic approaches, while *n-gram/NER cand. aligner.name* refers to the proposed candidate matching method with the specified aligner. The first columns show the language where the proposed method outperforms the heuristic the most, the seconds indicate where it underperforms the most, and the last columns provide the average results across all languages. **Bold** values are the overall best, and underlined values indicate the best projection-based approaches. Estonian (et) and Finnish (fi) are given as typical examples.

formance on pre-labelled Europarl parallel texts (Agerri et al., 2018), excluding translation and source NER labelling errors. It highlights that candidate matching methods yield results comparable to or better than prior approaches. The NER-based target candidates approach underperforms due to imperfect spans but surpasses plain model transfer by correcting mislabeled spans via source entity projection.

5 Conclusion

In this study, we presented novel annotation projection methods based on word-to-word alignments for cross-lingual NER.

The idea to compute word-to-word alignments between the original and back-translated labelled sentences in the same language, aimed at enhancing the quality of these alignments, did not produce the desired outcomes. This approach encountered significant challenges, primarily due to errors that occurred during the back-translation process.

In contrast, the proposed method of extracting candidates and matching them with source entities showed robust results. More specifically, the proposed formulation generally outperformed previous word-to-word alignment-based projection methods that relied on heuristics to deal with incorrect alignments.

By using the same NER model for candidate ex-

traction as in model transfer, the proposed approach can outperform model transfer. This is achieved by refining the labels for correctly predicted spans through projection from source entities.

Despite its advantages, the proposed approach remains heavily dependent on the quality of word-to-word alignments. However, the formulated ILP problem incorporates these alignments into matching scores that can be combined with other strategies using a weighted sum.

Our findings demonstrate that the projection-based data transfer approach can be a robust alternative to model-based methods for cross-lingual named entity recognition in low-resource languages.

Future research could aim to improve candidate extraction and explore alternative matching costs in addition to the alignment-based one. The proposed formulation, in contrast to heuristic approaches, facilitates the integration of various scoring mechanisms, allowing for the fusion of different scores to effectively address the limitations associated with each individual method.

Moreover, exploring the usage of LLMs for the projection step in cross-lingual NER pipelines shows potential, indicating that the development of multilingual LLMs could help enhance the performance of NER tasks across diverse languages, especially when working with limited labelled data.

Limitations

Translation Model Dependency: The performance of the proposed methods relies on the quality of the translation model used – in our case NLLB200-3.3B (Costa-jussà et al., 2022). Limitations in translation accuracy for certain languages may propagate errors through the pipeline, especially for morphologically complex or resource-scarce languages.

NER Model Dependency: Models used for extracting candidates or labelling translated sentences in the source language can be a source of errors. Incorrect predictions or omissions of entities by a model, coupled with the limited capability to correct such errors on the projection step, can adversely affect the quality of the resulting labelling of the original sentence. In our experiments, we rely on the XLM-R-Large model, fine-tuned on the English split of CONLL2003, although performance metrics may vary with different models.

Word-to-Word Alignment Model Dependency: The matching scores in the proposed ILP formulation for the projection step are computed based on word-to-word alignments. Therefore, the quality of the projection is inherently bounded by the quality of these alignments. In our study, we utilized state-of-the-art neural-based alignment models, specifically SimAlign and AWESoME. These models surpass previous statistically-based aligners as they incorporate the context of entire sentences. However, their performance remains limited. Furthermore, the quality of alignments varies between languages, which can be attributed to the representation of languages in the pretraining datasets of the models, as well as the inherent linguistic properties and structural differences among languages.

Dataset Variability: The proposed method demonstrates varying effectiveness across datasets, performing well on less complex languages but struggling with those that exhibit higher morphological complexity (e.g., Xhosa and Zulu). This indicates that additional adaptations may be needed for specific linguistic features.

Generalization Across Languages: The candidate matching method shows superior performance for most languages but underperforms in specific cases (e.g., Bengali, Kazakh, and Swahili), potentially due to inadequate representation in pre-training.

Optimization Heuristics: While the proposed optimization-based projection method reduces re-

liance on heuristics, the greedy algorithm used to solve the optimization problem may not achieve global optima in all scenarios.

Acknowledgements

The authors acknowledge the financial support by the Federal Ministry of Education and Research of Germany and by Sächsische Staatsministerium für Wissenschaft, Kultur und Tourismus in the programme Center of Excellence for AI-research "Center for Scalable Data Analytics and Artificial Intelligence Dresden/Leipzig", project identification number: SCADS24B.

The authors gratefully acknowledge the computing time made available to them on the high-performance computer at the NHR Center of TU Dresden. This center is jointly supported by the Federal Ministry of Education and Research and the state governments participating in the NHR.

Andrei Politov is deeply grateful to his late mother for her unwavering belief in him and the inspiration she provided throughout the development of this research.

References

- David Ifeoluwa Adelani, Graham Neubig, Sebastian Ruder, Shruti Rijhwani, Michael Beukman, Chester Palen-Michel, Constantine Lignos, Jesujoba O. Alabi, Shamsuddeen H. Muhammad, Peter Nabende, Cheikh M. Bamba Dione, Andiswa Bukula, Rooweither Mabuya, Bonaventure F. P. Dossou, Blessing Sibanda, Happy Buzaaba, Jonathan Mukiibi, Godson Kalipe, Derguene Mbaye, Amelia Taylor, Fatoumata Kabore, Chris Chinenye Emezue, Anuoluwapo Aremu, Perez Ogayo, Catherine Gitau, Edwin Munkoh-Buabeng, Victoire M. Koagne, Allahsera Auguste Tapo, Tebogo Macucwa, Vukosi Marivate, Elvis Mboning, Tajuddeen Gwadabe, Tosin Adewumi, Orevaoghene Ahia, Joyce Nakatumba-Nabende, Neo L. Mokono, Ignatius Ezeani, Chiamaka Chukwuneke, Mofetoluwa Adeyemi, Gilles Q. Hacheme, Idris Abdulmumin, Odunayo Ogundepo, Oreen Yousuf, Tatiana Moteu Ngoli, and Dietrich Klakow. 2022. Masakhaner 2.0: Africa-centric transfer learning for named entity recognition.
- Rodrigo Agerri, Yiling Chung, Itziar Aldabe, Nora Aranberri, Gorka Labaka, and German Rigau. 2018. Building named entity recognition taggers via parallel corpora. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Yang Chen, Chao Jiang, Alan Ritter, and Wei Xu. 2023. Frustratingly easy label projection for cross-lingual

- transfer. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5775–5796, Toronto, Canada. 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.
- Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.
- 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.
- Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2112–2128, Online. Association for Computational Linguistics.
- Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.
- David M. Eberhard, Gary F. Simons, and Charles D. Fennig. 2020. *Ethnologue: Languages of the World*, 23 edition. SIL International, Dallas.
- Hao Fei, Meishan Zhang, and Donghong Ji. 2020. Cross-lingual semantic role labeling with high-quality translated training corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7014–7026, Online. Association for Computational Linguistics.
- Iker García-Ferrero, Rodrigo Agerri, and German Rigau. 2022. Model and data transfer for cross-lingual sequence labelling in zero-resource settings. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6403–6416, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Iker García-Ferrero, Rodrigo Agerri, and German Rigau. 2023. T-projection: High quality annotation projection for sequence labeling tasks. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15203–15217, Singapore. Association for Computational Linguistics.
- Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR.
- REBECCA HWA, PHILIP RESNIK, AMY WEINBERG, CLARA CABEZAS, and OKAN KOLAK. 2005. Bootstrapping parsers via syntactic projection across parallel texts. *Natural Language Engineering*, 11(3):311–325.
- Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. 2020. SimAlign: High quality word alignments without parallel training data using static and contextualized embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1627–1643, Online. Association for Computational Linguistics.
- Duong Minh Le, Yang Chen, Alan Ritter, and Wei Xu. 2024. Constrained decoding for cross-lingual label projection. *ArXiv*, abs/2402.03131.
- Linlin Liu, Bosheng Ding, Lidong Bing, Shafiq Joty, Luo Si, and Chunyan Miao. 2021. MulDA: A multilingual data augmentation framework for low-resource cross-lingual NER. 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 5834–5846, Online. Association for Computational Linguistics.
- J.F. Maho. 1999. *A Comparative Study of Bantu Noun Classes*. Acta Universitatis Gothoburgensis: Orientalia et Africana Gothoburgensia. Acta Universitatis Gothoburgensis.
- Kalyani Pakhale. 2023. Comprehensive overview of named entity recognition: Models, domain-specific applications and challenges.
- Madhumangal Pal and G. P. Bhattacharjee. 1996. A sequential algorithm for finding a maximum weight K -independent set on interval graphs. *Int. J. Comput. Math.*, 60(3-4):205–214.
- Tanmay Parekh, I-Hung Hsu, Kuan-Hao Huang, Kai-Wei Chang, and Nanyun Peng. 2024. Contextual label projection for cross-lingual structured prediction. 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 5738–5757, Mexico City, Mexico. Association for Computational Linguistics.

Alberto Poncelas, Maksim Tkachenko, and Ohnmar Htun. 2023. Sakura at SemEval-2023 task 2: Data augmentation via translation. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 1718–1722, Toronto, Canada. Association for Computational Linguistics.

Sebastian Ruder, Jonathan Clark, Alexander Gutkin, Mihir Kale, Min Ma, Massimo Nicosia, Shruti Rijhwani, Parker Riley, Jean-Michel Sarr, Xinyi Wang, John Wieting, Nitish Gupta, Anna Katanova, Christo Kirov, Dana Dickinson, Brian Roark, Bidisha Samanta, Connie Tao, David Adelani, Vera Axelrod, Isaac Caswell, Colin Cherry, Dan Garrette, Reeve Ingle, Melvin Johnson, Dmitry Panteleev, and Partha Talukdar. 2023. XTREME-UP: A user-centric scarce-data benchmark for under-represented languages. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1856–1884, Singapore. Association for Computational Linguistics.

Henning Schäfer, Ahmad Idrissi-Yaghir, Peter Horn, and Christoph Friedrich. 2022. Cross-language transfer of high-quality annotations: Combining neural machine translation with cross-linguistic span alignment to apply NER to clinical texts in a low-resource language. In *Proceedings of the 4th Clinical Natural Language Processing Workshop*, pages 53–62, Seattle, WA. Association for Computational Linguistics.

Jörg Tiedemann. 2015. Improving the cross-lingual projection of syntactic dependencies. In *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*, pages 191–199, Vilnius, Lithuania. Linköping University Electronic Press, Sweden.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Sunna Torge, Andrei Politov, Christoph Lehmann, Bochra Saffar, and Ziyang Tao. 2023. Named entity recognition for low-resource languages - profiting from language families. In *Proceedings of the 9th Workshop on Slavic Natural Language Processing 2023 (SlavicNLP 2023)*, pages 1–10, Dubrovnik, Croatia. Association for Computational Linguistics.

Jannis Vamvas and Rico Sennrich. 2022. NMTScore: A multilingual analysis of translation-based text similarity measures. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 198–213, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jian Yang, Shaohan Huang, Shuming Ma, Yuwei Yin, Li Dong, Dongdong Zhang, Hongcheng Guo, Zhoujun Li, and Furu Wei. 2022. CROP: Zero-shot cross-lingual named entity recognition with multilingual labeled sequence translation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*,

pages 486–496, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Appendix

Isolated Evaluation of the Projection Step

Table 2 depicts performance only of the projection step, excluding translation and source NER labeling errors, on labelled parallel texts from the Europarl-based NER dataset⁵ (Agerri et al., 2018). Since our experiments with the tgt2tgt alignment direction yielded negative results, all pipelines presented in the table are only for the src2tgt case.

We can see that annotation projection methods that incorporate candidate matching can achieve results comparable to or better than previous approaches. Specifically, for the German language, the newly proposed method exhibits a significant performance improvement.

Projection method	de	es	it
Heuristic SimAlign	80.0	90.7	87.0
Heuristic AWESoME	81.9	90.3	87.3
n-gram SimAlign	89.8	89.2	87.8
n-gram AWESoME	92.0	88.6	87.2
Model transfer	67.5	74.1	69.6
NER SimAlign	74.5	79.8	72.3
NER AWESoME	74.7	80.0	72.0

Table 2: F1 scores resulting from the evaluation of only the projection step using the Europarl-based NER dataset with English as a source language.

The NER-based target candidates approach performs in this experiment worse due to imperfect spans predicted by the model. However, it still outperforms plain model transfer because it corrects wrongly predicted labels for spans using the projection from matched source entities.

In the case of the Spanish language, the heuristic word-to-word alignment-based algorithm slightly outperforms the proposed approach utilizing the n-gram candidate extraction strategy. This advantage arises from the algorithm’s ability to merge two continuous ranges of target words aligned with source entity words, when only one misaligned word exists between these ranges. In contrast, our approach exhibits this capability only in specific situations.

⁵ShkalikovOleh/europarl-ner

Notes on Complexity of the Problem

It can be demonstrated that the proposed problem, when excluding the second set of constraints that limit the number of projections for source entities, reduces to the maximum weight independent set problem on interval graphs, which is solvable in polynomial time (Pal and Bhattacharjee, 1996). Therefore, any potential complexity in the entire problem may be due to the combination of non-overlapping constraints and the constraints limiting the number of projections for each source entity. Although it is likely that the proposed ILP formulation could be solved in polynomial time, we cannot make a definitive claim since an appropriate algorithm has yet to be identified.

Insights from the MasakhaNER2 Dataset Experiments

Here, we provide further details on the results across different languages from the MasakhaNER2 dataset.

The set of 10 languages where the proposed method performs better than heuristics from the MasakhaNER2 dataset includes: Bambara ('bam'), Fon ('fon'), Hausa ('hau'), Igbo ('ibo'), Luganda ('lug'), Mossi ('mos'), Shona ('sna'), Swahili ('swa'), Wolof ('wol'), and Yoruba ('yor'). The second set of 8 languages where the proposed methods perform worst includes: Ewe ('ewe'), Kinyarwanda ('kin'), Luo ('luo'), Chichewa ('nya'), Tswana ('tsn'), Twi ('twi'), Xhosa ('xho'), and Zulu ('zul'). The exact metric values can be found in the provided GitHub repo.

Empathy vs Neutrality: Designing and Evaluating a Natural Chatbot for the Healthcare Domain

Cristina Reguera-Gómez

TNO

Utrecht University

c.regueragomez@uu.nl

Denis Paperno

Utrecht University

d.paperno@uu.nl

Maaïke H.T. de Boer

TNO

maaike.deboer@tno.nl

Abstract

As lifestyle-related diseases rise due to unhealthy habits such as smoking, poor diet, lack of exercise, and alcohol consumption, the role of Conversational AI in healthcare is increasingly significant. This study provides an empirical study on the design and evaluation of a natural and intuitive healthcare chatbot, specifically focusing on the impact of empathetic responses on user experience regarding lifestyle changes. Findings reveal a strong preference for the empathetic chatbot, with results showing statistical significance ($p < 0.001$), highlighting the importance of empathy in enhancing user interaction with healthcare chatbots.

1 Introduction

In our contemporary healthcare situation, lifestyle-related diseases are increasing, primarily influenced by unhealthy habits such as smoking, poor diet, lack of exercise, and alcohol consumption (Balwan and Kour, 2021). Simultaneously, conversational AI, or chatbots, have gained popularity and emerged as powerful tools, particularly in the healthcare sector (Amiri and Karahanna, 2022).

Nevertheless, the use of conversational agents in the healthcare domain is not too widespread, especially when compared to other industries such as travel and hospitality (Laranjo et al., 2018). Furthermore, very little is known about how the linguistic design of a medical conversational agent can impact the users' likelihood to employ it for their healthcare queries (Shan et al., 2022).

This paper focuses on the design and evaluate a natural and intuitive chatbot for the healthcare domain, including an empirical analysis of the results. More specifically, we investigate how

the use of empathy in generated messages can affect user experience during queries about lifestyle changes, hence influencing the likelihood to incorporate a healthcare conversational agent in their daily lives (de Boer et al., 2023). The two primary contributions of this study are:

1. To provide insight in the impact of empathetic versus neutral tones in messages in a LLM based chatbot.
2. To understand user expectations in human-computer interactions - using chatbots - in the healthcare domain, especially on lifestyle changes.

2 Related Work

2.1 Empathy and Language in Human-Computer Interaction (HCI)

Empathy plays a crucial role in making HCI more natural and intuitive. This paper draws on concepts of cognitive and affective empathy in human interaction.

Empathy is generally divided into two types: cognitive and affective empathy. Cognitive empathy is the ability to understand another person's emotional state without necessarily sharing it. Reniers et al. (2011) describe cognitive empathy as constructing a mental model of another's emotions. For example, someone with strong cognitive empathy can understand a friend's distress over a failure and offer appropriate advice. Cognitive empathy facilitates communication by enabling deeper understanding of others' experiences.

In contrast, affective empathy involves an emotional response to another's feelings. Affective empathy allows individuals to emotionally connect with others by vicariously experiencing their emotions. For instance, when a friend celebrates an achievement, a person with affective empathy would also feel joy. This type of empathy is essen-

tial for providing emotional support and fostering deeper connections.

Empathy is fundamental in social cognition (Iacoboni, 2005), allowing individuals to share experiences and goals. While empathy in humans involves complex cognitive and emotional mechanisms, chatbots can replicate empathetic communication by imitating patterns of human interaction. In practice, empathetic language in chatbots focuses on word choices that acknowledge the user's emotional state, effectively simulating empathy through language.

Human empathy, as standardly defined (Cuff et al., 2016), involves complex mental and emotional processes that chatbots do not possess. Instead, when discussing empathy in chatbots, we refer to their ability to produce responses that mimic human empathetic behaviours. Henceforth, a chatbot can be considered empathetic if its responses create the illusion of understanding and validating the user's feelings, even though it lacks real emotional experience.

2.2 Language Choices in Empathetic Communication

Empathetic communication in chatbots is achieved not only through understanding emotions but also through specific linguistic choices. Research by Yaden et al. (2023) identifies words associated with empathy, showing how language can create a sense of emotional support and connection. For example, the use of personal pronouns such as “I” and “you” helps create a more direct and personal interaction. Similarly, adjectives like “good” and “happy” convey positive emotional states, while verbs like “hope” and “need” can express concern or reassurance.

In addition to word choices, certain phrases play an essential role in empathetic communication. Lapointe (2014) found that common phrases like “I know” and “I understand” are often used to validate the user's feelings, while phrases like “it is” and “you are” are used to acknowledge the situation. These phrases help build emotional connection and foster a sense of understanding between the speaker and listener, which is crucial in emotionally sensitive interactions.

2.3 Research on Empathy in Chatbots

Research has increasingly focused on how empathetic language in chatbots can enhance user experience. Liu and Sundar (2018) explored whether

chatbots should offer both informational and emotional support when advising on personal issues. Their findings show that users generally prefer empathetic expressions over neutral advice, even when delivered by a chatbot, particularly when users are skeptical of machines' ability to show empathy.

Casas et al. (2021) further investigated empathetic chatbots by developing a system that generates emotionally attuned responses. Their chatbot outperformed both a standard chatbot and even some human responses in terms of perceived empathy. These studies demonstrate that empathetic language significantly improves user satisfaction with chatbots.

In the healthcare domain, the BabyTalk project (Mahamood and Reiter, 2011) examined parental preferences for emotionally sensitive medical reports about babies in neonatal care. Parents overwhelmingly preferred emotionally supportive, or affective, language over neutral descriptions. This shows that empathetic language is not only valued but essential in high-stress environments.

3 Conversational Agent Design

The decision to use a LLM-powered chatbot was driven by the need for a system capable of understanding and generating natural language with a high degree of fluency and contextual awareness. Unlike traditional rule-based or retrieval-based chatbots, which rely on predefined scripts or a database of responses, an LLM-powered chatbot can generate nuanced, contextually appropriate responses based on the specific needs of the user at any given moment.

One of the primary advantages of LLMs is their ability to process complex language inputs, making them well-suited for conversations that require deep contextual understanding, such as those in the healthcare domain. Given the nature of healthcare queries, which often involve detailed and sensitive information, it was essential to implement a system that could handle such complexities with a high degree of accuracy and flexibility.

The main objective of the chatbot is generating responses to user queries in a manner that is both informative and aligned with the specific version (empathetic or neutral) being tested.

The implementation of both the empathetic and neutral version is the same, except for the specific prompt used. In our first experiment, we evaluate

different LLMs to decide on the most suitable for our task.

The implementation of the chatbots involved using the AutoTokenizer from Hugging Face to preprocess and tokenise input data, ensuring compatibility with the model and efficient handling of user queries. The chatbot’s LLM was run locally using a GPU cluster, which was crucial for managing the computational demands of real-time text generation during user interactions. The web component of the chatbot was built using Flask, a lightweight web framework for Python, chosen for its simplicity and effectiveness in developing web-based applications.

3.1 Empathetic Chatbot Design

The empathetic version of the chatbot was designed with a specific focus on enhancing user experience through emotionally supportive communication. This required a detailed approach to ensure that the generated responses not only conveyed the necessary information but did so in a manner that validated and supported the user’s feelings. To implement empathy in the generated responses, the empathetic chatbot was programmed to follow a predefined prompt of empathetic communication, which diverges from that of the neutral one, and that was designed to shape its tone and language. The prompt explicitly instructs the model to generate responses that include empathetic expressions, focusing on word choices that reassure and validate the user’s experiences. This approach ensures that the chatbot’s interactions are not only informative but also emotionally supportive, thereby enhancing the overall user experience:

- **Neutral prompt:** “You are a chatbot who provides advice about lifestyle changes.”
- **Empathetic prompt:** “You are a friendly chatbot who provides advice about lifestyle changes. Your responses must be empathetic. A response is considered empathetic if it shows: 1) Comprehension towards the feelings of the other (i.e. ‘I understand that you are concerned about your health.’), and 2) Engagement in the feelings of the other (i.e. ‘I feel so happy that you have decided to live a healthier lifestyle.’). Remember, your generated advice should contain word choices that reassure and validate other people’s experi-

ences, according to the definition of an empathetic response.”

The empathetic prompt is characterised by two key elements in empathetic language that align with the literature, as they were mentioned in the previous section: comprehension of feelings (cognitive empathy) and engagement with feelings (affective empathy). The chatbot acknowledges and understands the user’s emotions, providing responses that appeal to the user’s emotional state. For example, it might say, “I understand that you are concerned about your health.” to validate the user’s concerns. Furthermore, the chatbot expresses positive reinforcement and encouragement, aiming to motivate the user. For instance, “I feel so happy that you have decided to live a healthier lifestyle!” is used to engage with and uplift the user. This prompt differs from that of the neutral chatbot, which only was instructed to provide advice about lifestyle changes, without any remark about the tone employed (see Figure 1). The prompts were chosen to meet the generative task requirements and to align with existing literature. They were also refined to ensure that the LLM could accurately understand the type of message it was asked to generate.

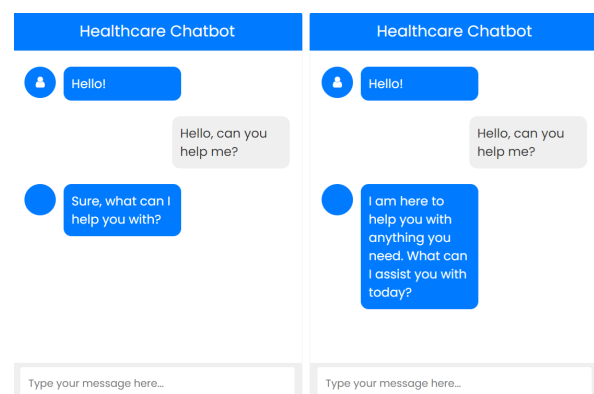


Figure 1: User interface and greetings generated by the neutral chatbot (left) and empathetic chatbot (right).

4 Experiment I: LLM Evaluation

The first experiment consisted of an evaluation of the responses generated by different LLMs, where the best-performing LLM was used as the basis of the chatbot in the user experiment (experiment 2).

4.1 Dataset

In order to do so, we asked the models to generate answers to questions obtained from the MASH-QA dataset (Zhu et al., 2020). This dataset was chosen because it is composed of consumer healthcare queries sourced from the popular health website WebMD, which features a wide range of articles covering various consumer healthcare topics. The answers to these queries are drawn from sentences or paragraphs within the articles related to the specific healthcare condition. These responses are curated by healthcare experts to ensure they accurately address the questions. We selected 100 questions, divided equally according to the following topics: exercise, food, smoking and alcohol. These topics are the same we used during the user experiment, since they are related to the most common causes of lifestyle diseases.

4.2 Models

We chose four LLMs, two of them being domain-specific—MedAlpaca (Han et al., 2023) and Meditron (Chen et al., 2023)—and the other two being general—GPT-4 (OpenAI, 2024) and Llama 3 (Meta, 2024). The motivation behind this choice is that it is crucial to experiment with a diverse set of LLMs, due to the lack of agreement in the literature over the superior performance of general or domain-specific models for medical tasks (Zhou et al., 2024; Nori et al., 2023). Henceforth, the two most suiting domain-specific LLMs were chosen, along with two general ones: one that had yielded good results for medical tasks (GPT-4), and a powerful, open-source one (Llama 3).

4.3 Evaluation

The evaluation was performed with G-Eval (Liu et al., 2023), a state-of-the-art NLG evaluation framework that uses a chain-of-thought (CoT) and a form-filling paradigm to assess the quality of texts generated by LLMs with GPT-4. The primary benefit of this evaluation framework is that it achieves a higher correlation (0.588) with human judgments compared to conventional metrics and previously established LLM-based evaluators, such as BLEU or ROUGE.

For this study, we tested G-Eval with the following metrics:

1. Fluency: “the quality of the answer in terms of grammar, spelling, punctuation, word

choice, and sentence structure” (Fabbri et al., 2021, as cited in (Liu et al., 2023)).

2. Coherence: “the collective quality of all sentences. We align this dimension with the DUC quality question of structure and coherence whereby ‘the answer should be well-structured and well-organized. The answer should not just be a heap of related information, but should build from sentence to a coherent body of information about a topic’” (Fabbri et al., 2021, as cited in (Liu et al., 2023)).
3. Groundedness: “the use of a fact in the answer, given the fact that this answer is conditioned by it” (Mehri & Eskenazi, 2020, as cited in (Zhong et al., 2022)).
4. Naturalness: “the quality of the answer in terms of being like something a person would naturally say” (Mehri & Eskenazi, 2020, as cited in (Zhong et al., 2022)).

These metrics were chosen because they encompass linguistic aspects related to human-likeness and user experience, so the scores associated with them can shed light on which models perform best on these aspects. In other words, this evaluation gives insights into how each LLM performs on the task of advice about lifestyle changes, from a linguistic point of view.

	GPT-4	Llama 3	MedAlpaca	Meditron
Fluency	0.865	0.864	0.844	0.846
Coherence	0.753	0.732	0.726	0.694
Groundedness	0.883	0.879	0.851	0.894
Naturalness	0.806	0.787	0.826	0.818
Avg. scores	0.827	0.816	0.812	0.813

Table 1: Results of the LLM evaluation.

4.4 Results and Discussion

As shown in Table 1, all the models had similar average scores across every metric, within the range 0.812 - 0.827, and with GPT-4 giving the highest score. Nevertheless, the results of the one-way ANOVA indicated that none of the differences between models in any metric were statistically significant ($p > 0.05$).

GPT-4 outperformed the other models in fluency (0.865) and coherence (0.753), which illustrates its linguistic abilities to generate dialogues.

The model's scores in fluency and coherence indicate its advanced linguistic capabilities, which are crucial for dialogue. Its fluency score (0.865) reflects the model's ability to produce smooth, easily readable text. GPT-4's coherence score (0.753) also surpasses the other models, suggesting that GPT-4 maintains logical consistency and context better throughout its responses. However, other individual metrics show slightly different results.

Meditron, one of the domain-specific models, received the highest score on groundedness (0.894), where the generated answers were compared about those from the golden standard in the MASH-QA dataset. Groundedness measures the factual accuracy and alignment of generated answers with a predefined gold standard, in this case, the MASH-QA dataset. Meditron's domain-specific training likely enhances its ability to produce accurate, relevant information within its specialised area. This specialisation illustrates the trade-off between general linguistic capabilities and domain-specific accuracy. While other models surpass Meditron in metrics concerning general dialogue quality, Meditron provides more precise and reliable information in the medical field.

The most surprising aspect of the data is in the results of naturalness, where MedAlpaca outperformed the other models (0.826). Naturalness evaluates how human-like the generated responses are, which is critical for creating engaging interactions. Despite MedAlpaca not leading in overall average scores or in fluency and coherence, its top performance in naturalness suggests that its generated messages are more intuitively aligned with human conversational patterns. Since naturalness was the most important metric, due to its relation to human-likeness, MedAlpaca was the chosen model to embed in the conversational agent of the main experiment.

5 Experiment II: User Experiment

We further compared the user experience with the neutral and empathetic conversational agents based on MedAlpaca.

5.1 Procedure

The experiment consisted of randomised controlled trials followed by cross-sectional surveys. A total of 25 participants were recruited, all of whom had completed university-level education. Of these participants, 68% identified as women,

and 32% identified as men. In terms of age distribution, 68% were between 25 and 34 years old, while 12% were either 18 to 24 years old or 35 to 44 years old. Additionally, 4% were aged 45 to 54 years, or 55 to 64 years. The participants did not necessarily search for lifestyle change. They interacted with the chatbot remotely and were instructed to complete the experiment in a quiet environment. The independent variables included factors such as the participant demographics, empathy condition and scenario.

An initial questionnaire was used to gather information on personal information such as age and gender, and a 5-point Likert scale questionnaire on the following topics: frequency of use with chatbots, feelings towards chatbot use, and feelings towards chatbot use in healthcare.

A within-subject design was used, where the same participant tested all conditions. During the experiment, they interacted with a chatbot and asked for lifestyle advice according to the following scenarios they enacted: eating healthier, exercising more, quitting smoking and reducing alcohol intake. After each of those four scenarios, they filled in a questionnaire.

5.2 Materials

The questionnaire used to test the participants' interaction was an adapted version of the Chatbot Usability Questionnaire (CUQ) (Holmes et al., 2019). The CUQ was selected due to its evaluation focus on conversational agents. Traditional metrics like the SUS (Brooke et al., 1996), though valuable, may not fully capture the nuanced aspects of chatbot interactions. The CUQ, in contrast, is designed to assess these aspects, making it a more suitable tool for evaluating the overall usability and effectiveness of chatbots.

While the original CUQ questions focus on the usability and evaluation of the chatbot's interface, they barely cover linguistic aspects. Henceforth, we modified the CUQ so that it could assess the chatbot's communication style, particularly the impact of empathetic versus neutral tones, which was one of the main objectives of this study. The adapted CUQ has two sets of questions: the first 8 of them evaluate the linguistic aspects of the interactions, and the other 8 focus on the usability aspect (see Table 2).

Question	
1	The chatbot's personality was realistic and engaging.
2	The chatbot seemed too robotic.
3	The chatbot was welcoming during initial setup.
4	The chatbot seemed very unfriendly.
5	The chatbot acknowledged my feelings appropriately.
6	The chatbot ignored my concerns.
7	The chatbot used language that was considerate and supportive.
8	The chatbot communicated in a cold and distant manner.
9	I trust the information provided by the chatbot.
10	I am skeptical of the advice the chatbot gave me.
11	Chatbot responses were useful, appropriate and informative.
12	Chatbot responses were irrelevant.
13	I am satisfied with my experience interacting with the chatbot.
14	My experience interacting with the chatbot was frustrating.
15	I would recommend this chatbot to others for lifestyle change advice.
16	I would advise others against using this chatbot for lifestyle change advice.

Table 2: Adapted Chatbot Usability Questionnaire used in our user experience study.

5.3 Data Collection and Analysis

Our data, collected anonymously and remotely, consist of the questionnaire's responses and chatlogs.

To investigate the impact of the empathy condition on the CUQ scores, we conducted a one-way ANOVA with blocking, using chatbot experience, chatbot opinion, and medical chatbot opinion as block variables. Before proceeding with the analysis, the dataset underwent a rigorous process to check for normality and homogeneity of variances. Normality tests, such as Q-Q plots and histograms, were conducted to visually inspect that the CUQ scores within each group (empathetic or neutral chatbot) followed a normal distribution. Additionally, the Kolmogorov–Smirnov test was applied to confirm that the distribution of the CUQ scores do not significantly differ from a normal distribution with equal mean and deviation.

Additionally, we conducted a qualitative analysis using the chatlogs and participants' answers to the open questions in the questionnaire.

5.4 Results and Discussion of the Quantitative Analysis

5.4.1 Overall CUQ Score

The overall CUQ score comprises the results from the complete questionnaire, without any distinction between the nature of the questions. The mean overall CUQ score for the empathetic chatbot was 66.3 ± 17.0 , and 49.4 ± 20.3 for the neutral one. Moreover, the empathetic chatbot consistently scored higher across all the scenarios, as it can be seen on Figure 2.

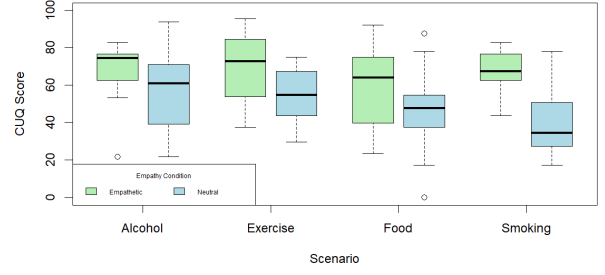


Figure 2: Overall CUQ scores per empathy condition and scenario.

The ANOVA results show that the empathy condition is highly statistically significant, with a p-value of 0.00002138 ($p < 0.001$), whereas the block variables (chatbot opinion, medical chatbot opinion, and chatbot use frequency) are not. The ANOVA coefficients illustrate how much changing each variable modifies the CUQ score. The intercept shows us the “base case” which, in this case, it is when the condition is empathetic, the chatbot frequency of use is yearly, and the opinion towards regular and medical chatbots is uncertain. In this base case, the average CUQ score was 64.9 ± 4.0 . Then, it showcases that, if from this base case we only change the condition to neutral, without modifying all the other variables, the average CUQ score will be reduced by -17.0 ± 3.8 . This effect is significant ($p < 0.001$). Other block variables are not statistically significant.

5.4.2 Linguistic CUQ Score

The linguistic CUQ score encompasses a subsection of scores about linguistic statements. These sentences evaluated if the chatbot's linguistic style while providing answers was perceived as welcoming, friendly and supportive by the participants. The empathetic chatbot had a mean linguistic CUQ score of 59.3 ± 9.7 , compared to 50.2 ± 9.2 for the neutral chatbot. Similarly to the previous section, the empathetic chatbot consistently outperformed the neutral one across all scenarios, as illustrated in Figure 3.

The ANOVA reveals that the empathy condition is also highly statistically significant, with a p-value of 0.000007095 ($p < 0.001$). Regarding the ANOVA coefficients, with the base case described in the previous section, the average CUQ score is 59.5 ± 2.1 . If the condition shifts from empathetic to neutral, without altering any other factors, the average CUQ score decreases by -9.1 ± 2.0 , a change that is statistically significant ($p < 0.001$).

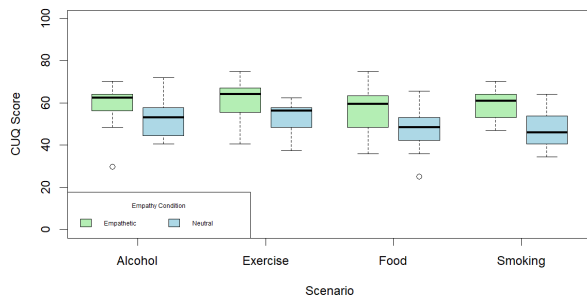


Figure 3: Linguistic CUQ scores per empathy condition and scenario.

The remaining block variables do not have a significant effect.

5.4.3 Usability CUQ Score

The usability CUQ score includes a subset of scores related to usability statements, assessing whether participants perceived the chatbot's answers as useful and relevant for lifestyle change advice. The empathetic chatbot had a mean usability CUQ score of 57.1 ± 8.5 , while the neutral chatbot scored 49.2 ± 12.3 . Consistent with previous findings, the empathetic chatbot consistently surpassed the neutral one in all scenarios, as shown in Figure 4.

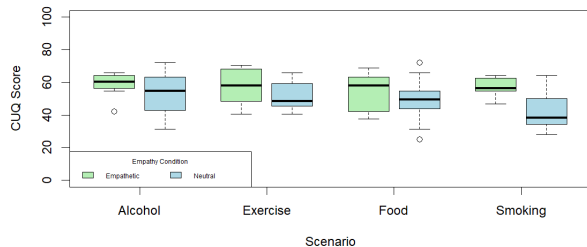


Figure 4: Usability CUQ scores per empathy condition and scenario.

The ANOVA demonstrates that the empathy condition has a highly statistically significant effect, with a p-value of 0.0003546 ($p < 0.001$). This indicates that the variation observed in the CUQ scores is unlikely to be due to chance. In the context of the base case described in the previous sections, the average CUQ score is 55.5 ± 2.3 . When the condition is shifted from empathetic to neutral, while keeping all other conditions constant, there is a notable decrease in the average CUQ score by -7.9 ± 2.1 . This decrease is statistically significant, with a p-value of less than 0.001, highlighting the impact of the empathy condition on the CUQ scores. Additionally, the analysis reveals that the remaining block variables do not have a

significant effect on the CUQ scores.

5.5 Results and Discussion of the Qualitative Analysis

5.5.1 Chatbot Dialogues

The dialogue excerpts obtained from the chatlogs highlight the differences in the way neutral and empathetic chatbots respond to user queries. In the interactions with the neutral chatbot, responses were direct and factual, with no additional commentary or expression of understanding. For example, when a participant asked about fruits low in sugar, the chatbot simply listed “apples, pears, and berries” without further elaboration. This pattern is consistent across all interactions with the neutral chatbot, where the focus was on delivering concise and straightforward information.

In contrast, the empathetic chatbot provided responses that not only addressed the participants' queries but also incorporated elements of empathetic communication. The responses often began with expressions of understanding or concern, followed by advice or information that was more detailed and personalised. For instance, when a participant mentioned feeling sluggish after meals, the empathetic chatbot acknowledged the participant's feelings and provided a comprehensive answer that included suggestions for dietary adjustments and a rationale behind those suggestions.

This approach aligns with the lexical and phrasal choices associated with empathetic communication as identified by Yaden et al. (2023) and Lapointe (2014), such as the use of first and second person pronouns (“I understand that you feel...”), modal verbs (“would”, “could”), and phrases that validate the user's experiences (“I hope this advice was helpful.”). Furthermore, an n-gram frequency analysis of the chatlogs reveals significant differences in word usage between the empathetic and neutral chatbots. Specifically, the words identified in Yaden et al. (2023) as characteristic of empathetic communication constitute 14.04% of all unigrams produced by the empathetic chatbot, compared to 6.80% in the neutral chatbot. Similarly, the two-word phrases listed in Lapointe (2014), account for 3.27% of all bi-grams generated by the empathetic chatbot, but only 0.86% in the neutral one. These differences are highly statistically significant ($p < 0.001$).

5.5.2 User Feedback

Participants' feedback further supports the contrast between the interactions with the neutral and empathetic chatbots. Users often described the responses of the neutral chatbot as "cold" and "robotic", noting the lack of empathetic engagement. One participant remarked that the chatbot's responses felt like "getting a list of Google results", which indicates that the interaction was perceived as impersonal and purely informational.

Conversely, feedback on the empathetic chatbot was generally positive, with participants appreciating the more engaging and supportive nature of its responses. Participants highlighted that the empathetic chatbot provided "useful" information and that the interaction felt "lively" and "holistic". One participant even mentioned that the chatbot's advice made them seriously consider changing their behaviour, such as reducing alcohol consumption. These comments and descriptions align with the conclusions from the previous subsection on the chatbot dialogues.

6 Conclusion

This paper aimed to investigate how the use of empathy in generated messages can affect user experience during queries about lifestyle changes.

The two primary contributions of this study are to provide insight in the impact of empathetic versus neutral tones in messages in a LLM based chatbot, and to understand user expectations in human-computer interactions - using chatbots - in the healthcare domain, especially on lifestyle changes.

The results of the first experiment show the differences between different LLMs, specifically two domain-specific and two general ones, on the different metrics fluency, coherence, groundedness and naturalness. These differences are not big, and the model with the most naturalness on the MASH-QA dataset concerning lifestyle questions - MedAlpaca - is chosen as the model to use in the second experiment.

The results of the second experiment show that empathy plays a crucial role in enhancing user satisfaction. The empathetic chatbot significantly outperformed the neutral chatbot across all dimensions measured by the Chatbot Usability Questionnaire, including overall user experience, linguistic perception, and usability ($p < 0.001$). This outcome highlights the importance of empathy in

chatbot communication, especially in healthcare settings where users are likely to seek comfort and understanding.

Beyond just evaluating chatbot performance, it was essential to analyse what users expect from these interactions and how these expectations shape their experience. Results revealed that users expect healthcare chatbots to offer more than just accurate and relevant information — they expect to participate in an interaction that mirrors human conversation. The high CUQ scores for the empathetic chatbot suggest that when these expectations are met, users are more satisfied and more likely to view the chatbot as a trustworthy and effective tool for asking about health advice.

Some of the limitations of this work include that the user experiment was specifically set to 4 scenarios and the participants recruited were not searching actively for lifestyle change. Although the participants were free to use their wording, the scenarios were quite restricted. In future work, it would be nice to conduct the experiment in a more realistic setting with more participants to verify our findings. Additionally, the sample size was relatively small and homogeneous, which hinders the generalisation of the results to a broader population. For example, individuals from different educational backgrounds or age groups might prioritise straightforwardness over empathy, which could yield slightly different results over the preferred tone in messages. Future work could replicate the experiment with a larger, more diverse sample to verify whether these preferences could be applied universally or are influenced by specific demographic factors.

In summary, it is evident that the most effective healthcare chatbots are those that balance generating accurate medical information with an empathetic dialogue style. While other general linguistic capabilities are important, the success of a healthcare chatbot heavily relies on its ability to communicate empathetically and align with human conversational patterns. This project has demonstrated that incorporating empathy into chatbot design can significantly improve user experience, making these tools more appealing and effective in supporting lifestyle changes and health-related decision-making.

Acknowledgments

This project has been supported by the Department of Data Science at TNO.

References

- Parham Amiri and Elena Karahanna. 2022. Chatbot use cases in the Covid-19 public health response. *Journal of the American Medical Informatics Association*, 29(5):1000–1010.
- Wahied Khawar Balwan and Sachdeep Kour. 2021. Lifestyle Diseases: The Link between Modern Lifestyle and Threat to Public Health. *Saudi Journal of Medical and Pharmaceutical Sciences*, 7(4):179–184.
- Maaike H. T. de Boer, Jasper van der Waa, Sophie van Gent, Quirine T. S. Smit, Wouter Korteling, Robin M. van Stokkum, and Mark Neerinx. 2023. A contextual Hybrid Intelligent System Design for Diabetes Lifestyle Management. In *Proceedings of the Fourteenth International Workshop on Modelling and Representing Context (MRC 2023)*.
- John Brooke et al. 1996. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7.
- Jacky Casas, Timo Spring, Karl Daher, Elena Mugellini, Omar Abou Khaled, and Philippe Cudré-Mauroux. 2021. Enhancing conversational agents with empathic abilities. In *Proceedings of the 21st ACM International Conference on Intelligent Virtual Agents*, pages 41–47.
- 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 arXiv:2311.16079*.
- Benjamin M.P. Cuff, Sarah J. Brown, Laura Taylor, and Douglas J. Howat. 2016. Empathy: A Review of the Concept. *Emotion Review*, 8(2):144–153.
- Tianyu Han, Lisa C. Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexander Löser, Daniel Truhn, and Keno K. Bresssem. 2023. MedAlpaca – An Open-Source Collection of Medical Conversational AI Models and Training Data. *arXiv preprint arXiv:2304.08247*.
- Samuel Holmes, Anne Moorhead, Raymond Bond, Huiyu Zheng, Vivien Coates, and Michael Mctear. 2019. <https://doi.org/10.1145/3335082.3335094> Usability testing of a healthcare chatbot: Can we use conventional methods to assess conversational user interfaces? In *Proceedings of the 31st European Conference on Cognitive Ergonomics*, page 207–214.
- Marco Iacoboni. 2005. Understanding Others: Imitation, Language, and Empathy. In Susan Hurley and Nick Chater, editors, *Perspectives on Imitation: Vol. 1. Mechanisms of Imitation and Imitation in Animals*, pages 77–99. The MIT Press, Cambridge, MA.
- Stephanie Lapointe. 2014. A Corpus Study of the Verbal Communication of Empathy/Sympathy by Anglophone Nurses in Quebec. Master’s thesis, University of Quebec at Chicoutimi, UQAC Repository.
- Liliana Laranjo, Adam G Dunn, Huong Ly Tong, Ahmet Baki Kocaballi, Jessica Chen, Rabia Bashir, Didi Surian, Blanca Gallego, Farah Magrabi, Annie Y S Lau, and Enrico Coiera. 2018. Conversational agents in healthcare: a systematic review. *Journal of the American Medical Informatics Association*, 25(9):1248–1258.
- Bingjie Liu and S Shyam Sundar. 2018. Should machines express sympathy and empathy? Experiments with a health advice chatbot. *Cyberpsychology, Behavior, and Social Networking*, 21(10):625–636.
- 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. *arXiv preprint arXiv:2303.16634*.
- Saad Mahamood and Ehud Reiter. 2011. Generating affective natural language for parents of neonatal infants. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 12–21.
- Meta. 2024. Introducing Meta Llama 3: The most capable openly available LLM to date. *AI at Meta Blog*.
- Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. 2023. Capabilities of GPT-4 on Medical Challenge Problems. *arXiv:2303.13375*.
- OpenAI. 2024. GPT-4 Technical Report. *arXiv:2303.08774*.
- Renate L. E. P. Reniers, Rhiannon Corcoran, Richard Drake, Nick M. Shryane, and Birgit A. Völlm. 2011. The QCAE: A Questionnaire of Cognitive and Affective Empathy. *Journal of Personality Assessment*, 93(1):84–95.
- Yi Shan, Meng Ji, Wenxiu Xie, Xiaobo Qian, Rongying Li, Xiaomin Zhang, and Tianyong Hao. 2022. Language Use in Conversational Agent-Based Health Communication: Systematic Review. *Journal of Medical Internet Research*, 24(7):e37403.

- David B. Yaden, Salvatore Giorgi, Matthew Jordan, Anneke Buffone, Johannes C. Eichstaedt, H. Andrew Schwartz, Lyle Ungar, and Paul Bloom. 2023. Characterizing Empathy and Compassion Using Computational Linguistic Analysis. *Emotion*, 24(1):106–115.
- 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. *arXiv preprint arXiv:2210.07197*.
- 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. *arXiv:2311.05112*.
- 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.

Assessed and Annotated Vowel Lengths in Spoken Icelandic Sentences for L1 and L2 Speakers: A Resource for Pronunciation Training

Caitlin Laura Richter
Reykjavik University
caitlinr@ru.is

Kolbrún Friðriksdóttir
University of Iceland
kolbrunf@hi.is

Kormákur Logi Bergsson
University of Iceland
klb16@hi.is

Erik Anders Maher
University of Iceland
eam16@hi.is

Ragnheiður María Benediktsdóttir
University of Iceland
rmb9@hi.is

Jon Gudnason
Reykjavik University
jg@ru.is

Abstract

We introduce a dataset of time-aligned phonetic transcriptions focusing on vowel length (quantity) in Icelandic. Ultimately, this aims to support computer assisted pronunciation training (CAPT) software, to automatically assess length and possible errors in Icelandic learners' pronunciations. The dataset contains a range of long and short vowel targets, including the first acoustic description of quantity in non-native Icelandic. Evaluations assess how manual annotations and automatic forced alignment characterise quantity contrasts. Initial analyses also imply partial acquisition of phonologically conditioned quantity alternations by non-native speakers.

1 Introduction

We present a corpus of Icelandic speech with manually corrected time-aligned phonetic transcriptions, targeted towards native and non-native Icelandic speakers' acoustic realisations of vowel quantity (length). Quantity is important in non-native (L2) Icelandic learning because it is contrastive, as in *vinur* [vɪːnʏr] 'friend', *vinnur* [vɪnːʏr] 'you, s/he work(s)', but challenging for many learners whose first languages do not use this cue. Computer assisted language learning (CALL) such as pronunciation training (CAPT) enables self-directed learning beyond traditional classrooms, and could provide opportunities to practice and internalise the Icelandic quantity system.

The acoustic implementation of Icelandic quantity has been studied only in small manually annotated native-speaker (L1) datasets. Addressing learners' needs requires (i) understanding quantity realisation in a broad sample of L1 and L2 speech, and (ii) developing scalable automated methods to describe a sufficient sample of the language and to

evaluate learners' speech relative to acoustic targets in autonomous interactive CAPT software.

We release time-aligned phonetic annotations for 2707 tokens of 72 Icelandic words,¹ greatly increasing the variety of contexts with available acoustic data on quantity, and including non-native speech for the first time. §4 uses this data to explore the realisation of quantity contrasts, comparing manual annotations and automated equivalents from the Montreal Forced Aligner (MFA), to address four **Research questions:**

RQ1 How do (subsets of) the annotated data relate to expectations from comparable studies?

RQ2 How strongly do quantity contrasts emerge in the annotated features, for L1 and L2 speakers?

RQ3 How accurate is Montreal Forced Aligner (MFA) timing, compared to gold annotations?

RQ4 How useful is MFA for issues in RQs 1-2?

2 Vowel Quantity in Icelandic

2.1 Language description

Stressed vowels in Icelandic, generally the first syllable of a word, have a quantity contrast conditioned by the vowel's environment (Einarsson, 1945; Kristinsson et al., 1985). A usual description of surface facts (Árnason, 1998; Gussmann, 2011) is that stressed vowels (including diphthongs) are long when followed by at most one consonant: *tré* 'tree', *hús* 'house', the first vowel *í* in *sími* 'telephone'. They are short when two or more consonants (geminate included) follow them before either the next vowel or the end of the word, e.g. *mjólk* 'milk', *a* in *pabbi* 'dad', except that specific clusters {p,t,k,s}+{j,v,r} are preceded by long vowels, e.g. long *i* in *sitja* 'sit'. In phonological terms it is conventional to say that vowels are long in open syllables and closed in short syllables, but it has proved challenging to complete this with an account of Icelandic syllable structure that does not

¹<https://github.com/catiR/length-contrast-data-isl>

circularly refer back to vowel length (see Árnason 2011; Craioveanu 2023; Fortuna 2016; Gussmann 2011; Þráinsson 1994; for issues bearing on phonological characterisation and the interface with morphosyntax). In practise, language teachers as well as linguists presenting the most thorough descriptions of Icelandic vowel length rarely complete formal phonological accounts of it (Árnason, 1998; Kristinsson, 1988; Craioveanu, 2023), so we continue the convenience of using the orthography as the simplest means to communicate.

2.2 Acoustic properties

The reader is referred to Pind (1999) for a review of acoustic research on Icelandic vowel quantity from Einarsson (1927) onwards, and subsequently Árnason (2011). In summary, absolute durations of long vs. short vowel segments overlap considerably, but there is a complementary relationship between vowels and the consonant(s) that follow them, such that these segments' combined duration in a word is relatively consistent: [a:l] in *gala* and [a:] in *galla* (Pind, 1995; Einarsson, 1927). Therefore, Icelandic vowel quantity is often described by a proportion, formulated as $V/(V+C)$, the ratio of vowel duration to total vowel+consonant durations (Pind, 1995); this calculation variously incorporates segments from either one or two syllables, as consonants in C are in either the coda of the stressed syllable or the onset of the next. Properties like vowel quality have also been identified as secondary cues to quantity for some vowels (Pind, 1999; Kristinsson et al., 1985). However, the acoustic research draws on narrowly restricted samples of few or one speaker(s), minimal vowel/syllable types, or only sentence-initial words. Audio and annotations are generally not accessible, and much in the language remains undescribed, such as any diphthongs, or L2 speech.

2.3 Teaching vowel quantity

Perceiving and producing quantity contrasts, as in *koma* 'come', *komma* 'comma', can be challenging for students of L2 Icelandic whose native language lacks such contrasts (McAllister et al., 2002). Computer assisted pronunciation training (CAPT) can offer help such as interactive exercises with feedback (Arnbjörnsdóttir et al., 2020; Bédi, 2022). Pronunciation accuracy assessment has been developed in coordination with lesson content of the free course *Icelandic Online*, but this does not give feedback on quantity errors, which is difficult to

provide without knowing what learners' acoustic targets are (Bédi, 2022; Bedi et al., 2024).

3 Corpus creation

3.1 Speech data

Audio is drawn from Samrómur, Samrómur Queries, Samrómur Unverified, and Samrómur L2 (Mollberg et al., 2021; Hedström et al., 2021, 2022a,b), recorded from 2019 onwards by native and non-native Icelandic speakers. Excluding child recordings (under age 18) there are in total 1.4 million sentences and 180,598 unique word types in over 1000 hours of speech. As corpora of crowd-sourced read sentences, these are typical of audio conditions that pronunciation training software processes for CAPT users.

Icelandic language proficiency levels and native language backgrounds of L2 Icelandic learners in these corpora are not reported, but plenty of variation in both of these factors was subjectively observed during manual annotation. Overall accuracy of phoneme reproduction and reading suggests that many speakers are intermediate to advanced learners of the language, although some speakers are likely within their first year of study and in certain recordings the speaker's prosody implies failure to semantically understand the sentence. Occasional deviations from Icelandic L1 pronunciation shown by L2 speakers were noted in vowel length and quality, with some relation to apparent first language background.

3.2 Target words

72 words of interest were sampled in two rounds of annotation. A complete list is provided in Appendix A.

The initial *validation* sample (36 words) is parallel to Experiment 2 from Pind (1999) and Experiment 1 of Pind (1995). In the former, 25 speakers read target words *saki*, *saggi*, *seki*, *seggi* within a paragraph; the reading context and number of speakers stand out as a clear choice for comparison to Samrómur data. From the latter, data on *kala*, *gala*, *Kalla*, *galla* (Pind, 1995) includes fewer speakers, but has similar enough acoustic analysis to also draw into comparison. Some of these 8 words are very infrequent, so to better assess reliability and variability, the validation sample is filled out with other two-syllable words that differ from Pind's only in the word onset, e.g. *tala*, *aggi*, *dreki*.

The second *extension* sample (36 words) highlights variation in stressed vowel phenomena, including: diphthongs; a range of vowels preceding different consonantal contexts such as nasals, fricatives, short and long trill, and assorted clusters; words with ‘exceptional’ consonant clusters preceded by long vowels; and quantity alternations within a morpheme as conditioned by compounding, inflection, and/or vowel syncope.

For each of the two samples, the most frequent words matching criteria were selected from Samrómur data. Annotators checked and filtered each word’s carrier sentences (in case of homonyms with different pronunciations), and where possible annotated at most 10 tokens from the same carrier sentence per L1/L2 speaker group.

3.3 Forced Alignment

As fully manual phonetic transcription is excessively time consuming, data was preprocessed by forced alignment, which annotators reviewed and corrected. The Montreal Forced Aligner (MFA) is a widely used toolkit built on Kaldi with standard GMM-HMM triphone acoustic models (McAuliffe et al., 2017). We train the aligner’s acoustic models on 20 hours of Icelandic speech from Samrómur, and use the General Icelandic Pronunciation Dictionary for ASR (Nikulásdóttir and Guðnason, 2017), to which a few target words not already present were manually added.

3.4 Annotation

Recordings were annotated by three of the authors while enrolled in undergraduate degrees on linguistics and/or Icelandic language at the University of Iceland. Two annotators are native Icelandic speakers and all have training in Icelandic phonetics. Annotation was carried out by reviewing and adjusting textgrids from MFA with the standard Praat interface (Boersma, 2024).

Phonetic annotations include only target words, not complete carrier sentence. The validation sample has up to 40 L1 + 40 L2 tokens per word, but in the extension sample this is reduced to 20 each, as pilot evaluation established this to be sufficient. An error tier was added to L2 speakers’ textgrids, using a simple coding scheme to mark when any of consonant, vowel quality, quantity, and/or stress placement errors were present in the target word. Most prominent among errors in vowel quality was a blending of the distinct vowel pairs *i* (L1 [i]) and

Segment	N	Same	10%	25ms	Error
L1-Ons	1617	64%	70%	87%	27ms
L2-Ons	931	69%	77%	91%	29ms
L1-V	1727	48%	62%	79%	31ms
L2-V	980	64%	76%	87%	29ms
L1-C	1727	57%	70%	83%	29ms
L2-C	980	66%	75%	85%	37ms
L1-Ratio	1727	42%	67%	–	19%
L2-Ratio	980	56%	75%	–	20%

Table 1: MFA accuracy for Onset, stressed Vowel, and post-vowel Consonant segment durations, and resulting V/(V+C) Ratio. Columns are: Number of tokens; percent of tokens where MFA’s duration/ratio is the Same, within 10%, or within 25ms of gold; and average magnitude of MFA Errors.

í (L1 [i]), as well as *o* (L1 [ɔ]) and *ó* (L1 [ou]), possibly explained by their orthographic similarity.

4 Evaluations

4.1 MFA Alignment Accuracy

First, automatic (MFA) phone alignments are compared to manual (gold) annotations (Table 1). MFA output has accurate durations for half to 2/3 of relevant segments, and of the rest, annotators’ adjustments are on average around 30ms. MFA inaccuracies affect the V/(V+C) ratio for roughly half of tokens, on average by 19-20% of the actual ratio.

4.2 Quantity classification

For a first look at acoustic correlates of the quantity contrast, K-nearest-neighbour (K=1,3,5,10,20) and linear regression classifiers were trained to predict vowels’ phonological length, using the following features extracted from gold (manual) annotations and MFA (automated) forced alignments: V/(V+C) Ratio, and segment durations **OnsDur**, **VDur**, **CDur**, and **WordDur** of respectively the target syllable Onset, Vowel, following Consonant(s), and whole Word. Classifiers use 5-fold cross validation, or leave-one-out cross validation for samples under 100 tokens. In §4 only the most informative feature sets are reported, using 5-nearest-neighbour classifiers which were typical of overall results.

In Table 2, a classifier for **All** tokens in the dataset has mediocre accuracy (L1 gold: 75%) using the V/(V+C) Ratio, with limited improvement from other available features. §4.3-4.6 therefore use linguistically restricted subsets of the data, aiming to isolate factors that moderate quantity cues.

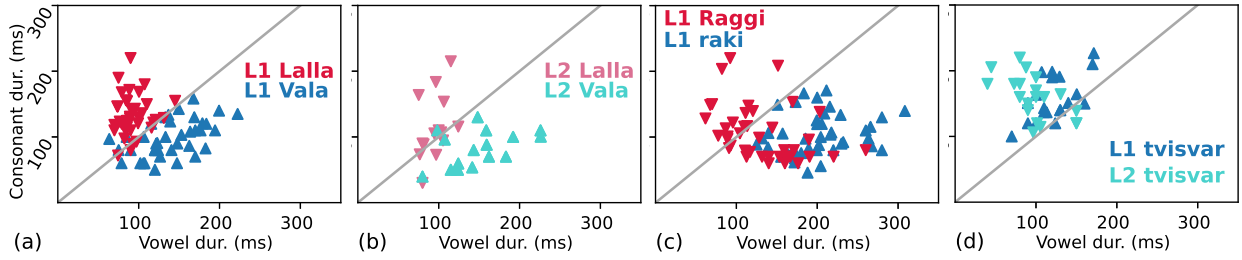


Figure 1: Stressed vowel and following consonant durations in *Lalla*, *Vala*, *Raggi*, *raki*, and *tvisvar*.

Sample	Features	L1-Gold	L1-MFA	L2-Gold	L2-MFA
All	Ratio	75%	74%	69%	69%
All	VDur	68%	71%	61%	60%
All	OnsDur, VDur, CDur	79%	79%	70%	70%
[C]ALa	Ratio	98%	93%	91%	84%
[C]ALa	VDur	84%	80%	66%	62%
[C]ALa	VDur, Cdur	99%	95%	91%	88%
*ALa	Ratio	94%	95%	81%	89%
*ALa	OnsDur, VDur, CDur	96%	96%	87%	90%
*AKi	Ratio	66%	68%	71%	68%
*AKi	VDur	67%	71%	78%	69%
*AKi	VDur, Cdur, WordDur	74%	76%	67%	72%
haus-	Ratio	100%	98%	76%	74%
Diphthong	Ratio	98%	97%	74%	76%

Table 2: Vowel length KNN classifier accuracy for L1 and L2 speech, with features computed from gold (manual) annotations and MFA alignments. Samples consist of: **All** 72 words of the dataset; **[C]ALa**: *dala*, *gala*, *tala*, *balla*, *galla*, *kalla*, *palla*; ***ALa**: the previous class plus *ala*, *fala*, *vala*, *dvala*, *svala*, *lalla*, *malla*; ***AKi**: *aki*, *aggi*, *baki*, *baggi*, *taki*, *kaggi*, *raki*, *raggi*, *þaki*, *blaki*, *maki*, *maggi*; **haus-**: *hausinn*, *hausnum*; **Diphthong**: *ása*, *ásta*, *hausinn*, *hausnum*, *jónas*, *jónsson*.

4.3 -ala, -alla

Results for [C]ALa in Table 2 examine two-syllable words of a plosive followed by [a:la] or [al:a], parallel to Pind (1995). Ratio is almost completely sufficient to distinguish L1 quantity (gold: 98% accuracy), while as expected, vowel duration (VDur) alone is not. However, VDur and CDur jointly may be slightly more useful than Ratio, especially with MFA features. ***ALa**, with more syllable onset types, is harder to classify by Ratio, but providing onset duration as a moderating factor may make up some of the difference, especially for L2 speakers. In all cases L2 speech was not classified as accurately as L1; examples of short (*Lalla*, personal name) and long (*Vala*, personal name) vowels in Figures 1a-b illustrate how short and long vowel cues overlap less for L1.

4.4 -aki, -aggi

***AKi** in Table 2 finds far worse ability to discriminate vowel quantity than either Pind (1999)’s 94%

(L1) classification accuracy for similar words with only single plosive onsets, or to our ***ALa** sample with varied onsets. Figure 1c gives an example of L1 speech for minimal pair *raki* [ra:cɪ] ‘humidity’, *Raggi* [rac:ɪ] (personal name), clearly not separable by the features that were sufficient for ***ALa**. For L1 but not L2, whole token duration is somewhat useful; this feature can reflect local speech rate and aspects of onset consonants.

4.5 Consonant cluster exceptions

In *tvisvar* ‘twice’ (Figure 1d), long [ɪ:] precedes an ‘exceptional’ cluster [sv]. L1 and L2 consonant cluster durations are all around 100-225ms, but L1 vowel durations (most 100-200ms) are notably longer than L2 (many under 100ms, few over 125ms). Reading *i* in *tvisvar* as a short vowel may show partially successful L2 acquisition of a vowel quantity system, but failure to incorporate nuances.

4.6 Diphthongs

[œi:] and [œi] in the words *hausinn*, *hausnum* ('the head', nominative and dative respectively) are distinct for L1 speakers, but not L2, who tend to insufficiently reduce diphthong duration in *hausnum*. This is unsurprising, as contrastive 'short' diphthongs are typologically rare. The observation generalises to **Diphthongs** (Table 2) with more vowel qualities and contexts, indicating promise for an area where CAPT may provide valuable feedback.

5 Discussion

At a high level, RQ1 is answered positively, as the conventional ratio proves to be an informative and interpretable feature, and more useful than absolute vowel duration alone. More specifically, for *-ala*, *-alla* words, expectations from a controlled study were strongly upheld in our crowdsourced data. For *-aki*, *-aggi*, aggregated data also would seem to match expectations, but a substantial proportion of individual tokens occupy an ambiguous region, at least in all currently examined feature spaces.

Regarding RQ2, quantity can be classified from the Ratio feature, but long and short vowels are not always well separable, and absolute durations of vowel and consonant carry some useful information beyond the ratio. Location of a best threshold for any features also varies based on several other factors. In some cases, factors are identified and controlled for, with good to excellent classifier performance. In other cases this work is ongoing, and a general-purpose solution remains to be developed; it could require phoneme identity labels, representations of syllable and word structure, prosodic environment, spectral features, etc. Qualitatively, during annotation we had observed noticeable length errors in some of the same L2 samples (e.g. *tvisvar*, *hausnum*) where the measured features indicated loss of contrast for L2 speakers as compared to L1, which is an encouraging sign that the features can capture perceptually important dimensions of contrast.

Addressing RQ3, MFA frequently mismeasures segments in this corpus by around one-third of the true duration, although the particular values for all MFA measures arise from a specific acoustic model and do not generalise to others. The relevant interpretation is that typical applications of MFA, like ours obtaining decent word alignments from 20 hours of in-domain training speech, cannot be relied on for the accuracy desired by primary de-

scriptive research in phonetic segments. MFA preprocessing may also introduce bias in the gold annotations, which would not critically affect CAPT development and is well worth the saved time over full manual transcription, but true inaccuracy of MFA may be underestimated.

Despite considerable room for improvement, alignment errors had small impacts (RQ4) on classifiers' ability to distinguish short and long vowels. The relative utility of various features also appears similar whether using manual or automated data.

5.1 Contributions

We freely release our annotations, whose audio and metadata is already public. The data is available at <https://github.com/catiR/length-contrast-data-isl> accompanied by an online platform for visualisations/analyses as in §4. This is the most accessible data on L1 Icelandic vowel length, and the first L2 data. Preliminary analysis reveals L2 acquisition of a quantity contrast to an extent, but also some systematic challenges.

Towards CAPT software development, MFA and temporal features derived from it are identified as an adequate starting point to (i) characterise distinctiveness or ambiguity of typical pronunciations across phonological quantity contrasts; and (ii) classify apparent quantity of L2 pronunciations, and alert learners to errors if their pronunciation is unambiguously not what it should be.

Acknowledgments

We would like to thank Branislav Bédi, Gunnar Thor Örnólfsson, Luke O'Brien, Marc Daníel Skipstað Volhardt, Staffan Hedström, Þorsteinn Daði Gunnarsson, and all of our colleagues involved in pedagogic development of Icelandic computer assisted language learning and collecting L1 and L2 speech.

This work was supported by The Icelandic Centre for Research (RANNÍS), under the Icelandic Student Innovation Fund project *Speech corpus of L1 and L2 Icelandic vowel length*, Grant 2412155-1101.

References

- Kristján Árnason. 1998. Vowel shortness in Icelandic. *Phonology and morphology of the Germanic languages*, pages 3–25.
- Kristján Árnason. 2011. *The phonology of Icelandic and Faroese*. Oxford University Press.

- Birna Arnbjörnsdóttir, Kolbrún Friðriksdóttir, and Branislav Bédi. 2020. Icelandic online: twenty years of development, evaluation, and expansion of an LMOOC. *CALL for widening participation: short papers from EUROCALL*, pages 13–19.
- Branislav Bédi, Jane O’Toole, and Monica Ward. 2024. Resourceful approaches in call for less-commonly taught languages (LCTLs): Case studies on Icelandic, Irish, and Nawat. *EuroCALL 2023: CALL for all Languages*.
- Paul Boersma. 2024. Praat: doing phonetics by computer [Computer program]. <http://www.praat.org/>.
- Branislav Bédi. 2022. Development of online tools supporting the learning of Icelandic as a foreign and second language. In Branislav Bédi, Halldóra J. Þorláksdóttir, and Kolbrún Friðriksdóttir, editors, *Tungumál í samhengi / Perspectives on Language and Context*. Reykjavík: Stofnun Vigdísar Finnbogadóttur í erlendum tungumálum.
- Radu Craioveanu. 2023. *Weighing Preaspiration: Phonetics, Phonology, & Typology of a Laryngeal Phenomenon*. Ph.D. thesis, University of Toronto (Canada).
- Stefán Einarsson. 1927. *Beiträge zur Phonetik der isländischen Sprache*. AW Brøgggers.
- Stefán Einarsson. 1945. *Icelandic: Grammar, texts, glossary*. The Johns Hopkins Press.
- Marcin Fortuna. 2016. Icelandic post-lexical syllabification and vowel length in CVCV phonology. *The Linguistic Review*, 33(2):239–275.
- Edmund Gussmann. 2011. Getting your head around: the vowel system of Modern Icelandic. *Folia Scandinavica Posnaniensia*, 12:71–91.
- Staffan Hedström, Judy Y Fong, Ragnheiður Þórhallsdóttir, David Erik Mollberg, Smári Freyr Guðmundsson, Ólafur Helgi Jónsson, Sunneva Þorsteinsdóttir, Eydís Huld Magnúsdóttir, and Jon Guðnason. 2021. Samromur queries 21.12. CLARIN-IS.
- Staffan Hedström, Judy Y. Fong, Ragnheiður Þórhallsdóttir, David Erik Mollberg, Smári Freyr Guðmundsson, Ólafur Helgi Jónsson, Sunneva Þorsteinsdóttir, Eydís Huld Magnúsdóttir, and Jon Guðnason. 2022a. Samromur unverified 22.07. CLARIN-IS.
- Staffan Hedström, Judy Y. Fong, Ragnheiður Þórhallsdóttir, David Erik Mollberg, Thomas Mestrou, Smári Freyr Guðmundsson, Ólafur Helgi Jónsson, Sunneva Þorsteinsdóttir, Eydís Huld Magnúsdóttir, Caitlin Laura Richter, Ragnar Pálsson, and Jon Guðnason. 2022b. Samromur L2 22.09. CLARIN-IS.
- Ari Páll Kristinsson. 1988. *The pronunciation of modern Icelandic: a brief course for foreign students*. Málvísindastofnun Háskóla Íslands.
- Ari Páll Kristinsson, Friðrik Magnússon, Margrét Pálsdóttir, and Sigrún Þorgeirsdóttir. 1985. Um and-stæðuáherslu í íslensku. *Íslenskt mál og almenn maálfræði*, 7:7–47.
- Robert McAllister, James E Flege, and Thorsten Piske. 2002. The influence of L1 on the acquisition of Swedish quantity by native speakers of Spanish, English and Estonian. *Journal of phonetics*, 30(2):229–258.
- Michael McAuliffe, Michaela Socolof, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger. 2017. Montreal forced aligner: Trainable text-speech alignment using kaldi. In *Interspeech*, volume 2017, pages 498–502.
- David Erik Mollberg, Ólafur Helgi Jónsson, Sunneva Þorsteinsdóttir, Jóhanna Vigdís Guðmundsdóttir, Steinþór Steingrímsson, Eydís Huld Magnúsdóttir, Judy Y Fong, Michal Borsky, and Jon Guðnason. 2021. Samromur 21.05. CLARIN-IS.
- Anna Björk Nikulásdóttir and Jón Guðnason. 2017. General pronunciation dictionary for ASR. CLARIN-IS.
- Jörgen Pind. 1995. Speaking rate, voice-onset time, and quantity: The search for higher-order invariants for two Icelandic speech cues. *Perception & Psychophysics*, 57(3):291–304.
- Jörgen Pind. 1999. Speech segment durations and quantity in Icelandic. *The Journal of the Acoustical Society of America*, 106(2):1045–1053.
- Höskuldur Þráinsson. 1994. Icelandic. In *The Germanic Languages*, pages 142–189. Routledge.

Appendix A. Detailed annotation contents

Word	L1 tokens	L2 tokens	Carriers
ala	2	3	4
dala	40		14
dvala	40		22
fala	1	1	2
gala	5		2
svala	40	4	17
tala	40	40	47
vala	40	16	41
aki	9		3
baki	40	40	65
blaki	16	2	4
maki	24	4	10
raki	42	1	13
taki		19	15
þaki		9	6
breki		40	30
dreki	40	5	18
leki	66	1	17
speki	40	2	17
veki	25		7
balla	1		1
galla	10	5	4
kalla	33	2	15
lalla	40	14	36
malla	37	4	14
palla	40	9	26
aggi	19		6
baggi	26		8
kaggi	11		1
maggi	40	25	34
raggi	41	11	25
beggi	29	7	11
eggi	40	7	23
leggi	41	3	20
skeggi	40	2	16
veggi	40	6	28

Table 3: Counts of L1 and L2 tokens, and unique carrier sentences, for words in the validation sample. While even distribution was a guiding principle, the contents are necessarily a compromise between balance and availability of data.

Word	L1 tokens	L2 tokens	Carriers
ása	20	20	16
bera	21	20	28
betri	20	23	15
brosir	20	20	25
fara	21	20	15
færa	20	18	29
færi	20	20	38
hausinn	20	20	14
jónas	20	20	28
katrín	20	20	31
kisa	20	12	15
koma	20	20	33
leyfa	20	20	21
muna	20	21	22
nema	20	20	15
sama	20	20	20
sækja	20	20	39
sömu	20	20	16
tvisvar	20	20	28
vinur	20	20	35
ásta	20	19	18
farðu	21	18	23
fossinn	20	15	13
færði	20	20	34
hausnum	20	22	18
herra	20	20	21
jónsson	20	20	29
leyfðu	26	20	23
mamma	20	20	15
missa	20	20	37
mömmu	20	20	15
nærri	20	20	34
snemma	20	20	17
sunna	20	20	21
tommi	20	20	32
vinnur	20	10	27

Table 4: Counts of L1 and L2 tokens, and unique carrier sentences, for words in the extension sample. Long vowels are in the upper section of the table and short vowels in the lower.

The BRAGE Benchmark: Evaluating Zero-shot Learning Capabilities of Large Language Models for Norwegian Customer Service Dialogues

Mike Riess

Research and Innovation

Telenor Group

Oslo, Norway

mike.riess@telenor.com

Tollef Emil Jørgensen

Department of Computer Science

Norwegian University of Science and Technology

Trondheim, Norway

tollef.jorgensen@ntnu.no

Abstract

This study explores the capabilities of open-weight Large Language Models in a zero-shot learning setting, testing their ability to classify the content of customer service dialogues in Norwegian from a single instruction, named the BRAGE benchmark. By comparing results against widely used downstream tasks such as question-answering and named entity recognition, we find that (1) specific instruction models greatly exceed base models on the benchmark, (2) both English and multilingual instruction models outperform the tested Norwegian models of similar sizes, and (3) the difference between base and instruction models is less pronounced than in other generative tasks, suggesting that BRAGE is a challenging benchmark, requiring precise and generalizable instruction-tuning.

1 Introduction

Satisfied customers are critical to any telecommunications provider's long-term success and sustainability. An essential piece of this puzzle is to provide the best possible customer service once a problem has occurred and try to avoid any further negative experiences (PwC, 2018). Advances in Automatic Speech Recognition and text analysis methods have transformed customer service processes, enabling providers to gain aggregated insights from the large volume of daily calls. These insights allow the telecommunications provider to act quickly on issues that influence multiple customers in close to real-time. However, creating models capable of analyzing transcribed conversations remains challenging due to the technical expertise required and the time-intensive development process. Additionally, the distribution of the incoming calls may change over time

due to concept drift (Riess, 2022), requiring frequent updating of models to maintain operational quality – thus increasing costs. In-context learning (Brown et al., 2020) and the ongoing efforts on adapting Large Language Models (LLMs) to lower-resource languages such as Norwegian (NORA AI, 2024) offer a promising solution to this problem.¹ This study explores the potential of open-weight LLMs to enable non-expert users to perform zero-shot content classification. To this end, we introduce *BRAGE*, a private benchmark designed to evaluate zero-shot classification of transcribed conversations in Norwegian. Using the same instructions provided to human annotators for creating ground truth labels, various LLMs are tasked with classifying the content of conversations between customers and customer service agents. To assess the feasibility of this approach, we evaluate base- and instruction-tuned open-weight LLMs, including pre-trained and fine-tuned Norwegian models. Given the sensitive nature of the data, BRAGE is a private benchmark. Aggregated results, however, are publicly shared to ensure transparency. Additionally, we² aim to facilitate academic collaboration by performing benchmark evaluations on BRAGE and sharing the results with the public when requested by researchers in the Nordics. Code for the benchmark is available on GitHub.³

The Customer Service Process

The case process from which we create BRAGE is an analytics unit within a telecommunications provider, supporting customer service with insights on current calls. When customers contact the service center, calls are, if permitted, recorded

¹Low-resource is relative to the amount of openly available resources for fine-tuning large language models, e.g., instruction-tuning. For Norwegian, this is largely limited to machine-translated datasets.

²Telenor Research and Innovation

³<https://github.com/tnresearch/brage.2025>

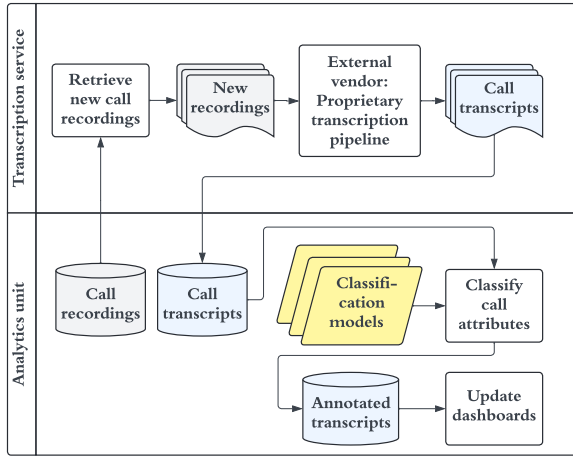


Figure 1: The call transcription and analysis process.

and subsequently transcribed.

Transcription is performed by a proprietary service from an external vendor, similar to *Whisper* (Radford et al., 2022). The transcripts are then processed and annotated using classification models, predicting business-relevant attributes. Figure 1 shows an overview of this process. In our benchmark, we modify the process by replacing the classification model(s) in Figure 1 with a single open-weight LLM, which is prompted using the *codebook* (annotation guidelines, see Forman and Damschroder, 2007) previously used by the human annotators.

Research Questions

We define the following research questions:

RQ1 How do open-weight Norwegian models compare regarding their performance on the BRAGE benchmark?

RQ2 How do these results compare and align with other downstream generative tasks for Norwegian?

To answer RQ1, we benchmark a set of open-weight LLMs, including Norwegian pre-trained and fine-tuned ones on BRAGE, and subsequently compare these results to other downstream evaluations from ScandEval (ScandEval, 2024) to answer RQ2.

2 Related work

In recent research and development of LLMs, it has become clear that these models can adapt to the context they are presented (Brown et al., 2020),

to such a degree that they do not need further training to adapt to a particular task. This is also known as In-Context Learning (ICL) (Li et al., 2023), and can be done using a single instruction with no examples (*zero-shot*) or multiple examples at inference time (*few/N-shot*). ICL dramatically reduces the associated costs in evolving systems, as one no longer relies on expensive training pipelines to support shifts within data and business needs.

Open-weight models such as Llama (Dubey et al., 2024), Mistral (Jiang et al., 2023), and Gemma (Team et al., 2024) have proven to be competitive when considering the efficiency vs performance trade-off. “Smaller” models ($\leq 70B$) have achieved impressive results across numerous tasks (Chiang et al., 2023; Wolfe et al., 2024), and we have reached a point where the performance gap between open-weight models and larger proprietary models is quickly diminishing. However, evaluating LLMs and quantifying this gap is incredibly difficult (Chandran et al., 2024; Biderman et al., 2024), e.g., because of benchmark data leaks (Xu et al., 2024). To circumvent this, LMSYS (Zheng et al., 2023b) developed an Elo-score system where users rank anonymous models – giving an idea of real-world performance. As of September 2024, several open-weight models ($\leq 70B$) rank on par with much larger models.

Narrowing in on Scandinavian languages, ScandEval (Nielsen, 2023) is a tool for evaluating models on language-specific data for downstream tasks. Focusing on constrained generation with LLMs in this study, the Norwegian generative ScandEval benchmark is highly relevant, with 134 different models currently benchmarked across 9 different datasets.⁴ Because our data is sensitive, the models must run locally to avoid transferring the data during inference. Open-weight models, particularly those explicitly trained on Norwegian data, are included for evaluation. While open models come with the advantage of enabling continued training, supervised fine-tuning (SFT), merging (Yang et al., 2024), and *jailbreaking* (Zou et al., 2023; Zhang et al., 2024) along with an ever-growing set of preference tuning techniques (Gao et al., 2024), the search space of optimization methods is so vast that finding an optimal approach is near impossible. Furthermore, Ghosh et al. (2024) found SFT to degrade

⁴As of October 2024. Visit <https://scandeval.com/norwegian-nlg/> for a full overview of Norwegian benchmark results.

knowledge and reduce output quality of the pre-trained models, with the same observations for using fine-tuned LLMs for telecommunications (Barnett et al., 2024). Moreover, research within out-of-domain generalizability in traditional machine learning and LLMs suggests that domain-specific training will reduce a model’s performance (Wald et al., 2021; Yang et al., 2022; Yu et al., 2024). Mosbach et al. (2023) challenges this idea and performs thorough evaluations on the generalization of models for ICL and SFT in the parameter range of 125M to 30B. While Mosbach et al. find compelling results for the case of SFT, such that a smaller SFT can outperform higher-parameter models with ICL, these findings diminish as the model sizes grow, and ICL performs significantly better when evaluating in-domain than SFT for model sizes $\gtrsim 7B$.

3 Methodology

3.1 Data

The BRAGE benchmark consists of 300 transcribed Norwegian customer service phone calls from a telecommunication provider in its current version. Transcription is done by an external vendor with a proprietary algorithm. An internal validation of ten randomly sampled calls showed an average Word Error-Rate (WER) of 12.41% (overall) and 0.89% for business-critical terms like product names.

Annotation Each transcript has been annotated with several attributes related to each call, among those, the “product” attribute, which includes eight categories: *Annet* (Other), *Mobil* (Mobile), *Tjenester* (Services), *Bredbånd* (Broadband), *TV*, *Bredbånd-mobilt* (Broadband-Mobile), *E-post* (Email), *Forsikring* (Insurance). An internal analytics team developed the definitions for these categories in August 2023 and has since refined them multiple times to remove ambiguous categories. Experiments were conducted in October 2024. The 300 calls included in this study were randomly sampled and subsequently annotated by a Senior Analyst with 25+ years of domain experience (Customer Service) in two iterations: an initial annotation and a review two weeks after the initial annotation.

Class distribution The exact distribution of these categories cannot be shared due to business sensitivity. However, to provide the reader

with an impression of the class imbalance, a randomly ordered overview is as follows: *Category 1* (5.7%), *Category 2* (13%), *Category 3* (8%), *Category 4* (7%), *Category 5* (37%), *Category 6* (9.3%), *Category 7* (9%), *Category 8* (11%). The expected accuracy of random guessing will thus be $\sum_{i=1}^8 p_i^2 = 19.72\%$, where p_i is the individual class probability, while classifying all calls as *Category 5* will yield an accuracy of 37%. As this study evaluates *zero-shot* classification on a private test set, the models cannot use a zero-rate strategy (Devasena et al., 2011) or overfit to the majority class. To further account for class imbalance, we report our results with Macro-F1 and Matthews Correlation Coefficient in addition to Accuracy. 2 shows a modified example of a call transcript. This example shows the nature of the transcribed phone calls in the BRAGE benchmark, which is anonymized with `_blacklist_`, `_number_`, and `_name_` tokens. An English version can be found in Figure 7 in Appendix A.

3.2 Experiments

Each experiment consists of multiple runs, where a run represents a unique combination of a prompt and a model. For each run, we concatenate a zero-shot instruction (discussed in detail in section 3.3) with a truncated call transcript (the first 250 tokens), asking the model to determine which product the conversation is about. We utilize settings consistent with those employed in ScandEval (Nielsen, 2023), including a temperature of 0.0, a fixed seed value, and 10 iterations of bootstrap sampling for each run to ensure robustness.

The output space of the LLM is constrained to the valid product categories using Outlines (Willard and Louf, 2023) and Transformers (Wolf et al., 2020) for inference. We compare Pre-trained multilingual base models (P) Pre-trained base models in Norwegian Bokmål (PNB), Instruction-Tuned multilingual models (IT) and models Fine-tuned in Norwegian Bokmål (FNB). We use a single prompt for all models. Prompt formatting varies depending on the model type, but follow the guidelines in the model card from the respective authors. We use *ChatML* and *Alpaca* formats for instruction-tuned models, while base models receive prompts without any prior formatting. Upon completing all runs, we calculate aggregated metrics to evaluate the performance and outcomes of our experiments. To put our results

Hei, du snakker med _blacklist_. Jeg har gått over fra privat mobilabonnement til å få man dekket av jobben det skjedde for cirka to og en halv uke siden, og så ser jeg i nettbanken at de har en faktura som står til godkjenning for januar. Ja, ja. _number_ desember, da skulle liksom mobing av Mobilabonnementet det private avsluttes nå, så jeg lurte på nå kan jeg sjekke at faktureringen som har blitt riktig. _blacklist_. Ja, det skal hjelpe deg meg, kan jeg få ditt følge navn, fødselsdato og adresse. _blacklist_ _name_, _blacklist_ _blacklist_, _blacklist_ _blacklist_ _blacklist_ på _blacklist_ _blacklist_. Ja og postnummer A? _number_ _number_ _number_ _blacklist_ _blacklist_, men det var en telefon, ja, ja, du lurte på om faktum, altså du hadde en utestående faktura, sa du. Ja, i banken så legger jeg en faktura til godkjenning for januar. Ja. På _name_ sa du nei, den er betalt, fakturaen er betalt, ja. _number_ _number_ og _blacklist_ komma _blacklist_. Greit. Men når jeg lagt ned den fakturaen, så står det at det er for januar. Ja, da vil du få tilbake hele månedsprisen tilbake faktisk siden du abonnementet ditt ble endra, så ble endre da før januar starta, så får du alt jeg tilbake, ja, det er det, så du vil faktisk få tilbake skal vi sjå _number_ _number_ og og _blacklist_ det samme kontonummer du sist betalte med. Okay, ja, så da, da ble det på en måte avsluttet. Ja. Okay. Ja. På _number_ kroner, så da trekker vi fra den eller _blacklist_, men _blacklist_ ja. Det, ja. Den bare å avsette. Okay, ja for _blacklist_, så da, da får vi litt motta en sluttfaktura da. Greit, da glemmer jeg den fint ha det godt. Ha det, fint du.

Figure 2: Modified call example with similar quality as the transcripts in our dataset. The topic of this call is 'Mobile'. The terms _blacklist_, _name_ and _number_ are anonymized entities.

into perspective, we have retrieved the ScandEval scores of each model included in our study. The selected benchmarks cover the downstream tasks of named entity recognition (NorNE, Jørgensen et al., 2020), sentiment analysis (NoReC, Vellidal et al., 2018), question-answering (NorQuAD, Ivanova et al., 2023) and commonsense reasoning (a truncated and machine-translated HellaSwag dataset Zellers et al., 2019, as implemented in ScandEval).

3.3 Prompting

The benchmark aims to assess LLMs' zero-shot performance on annotation tasks using human-equivalent instructions, evaluating the potential of automating the annotation task in a user-friendly manner. We, therefore, use the same guidelines when creating the ground truth. The only adaptation is to add a short introduction sentence and a final instruction for the model. The full prompt can be seen in Figure 3, which shows an anonymized version of this prompt. An English translation can be found in Figure 8 in Appendix A.

4 Results

4.1 RQ1: General Performance on the BRAGE benchmark

Looking at the right side of Figure 4, we observe that the variation across the base models is very low and that the average accuracy is around the same value as the expected accuracy of a random guess (19.72%). In stark contrast, the in-

struct models (left side) vary much more and have higher average accuracy, fostering the hypothesis that instruction fine-tuning is essential for our zero-shot classification task. Looking at the best performing models in Table 1, we find that the English/Multilingual *Gemma2* models outperform any model explicitly pre-trained (PNB) and/or fine-tuned on Norwegian data (FNB).

The average accuracy of the *Gemma2* models (43.53%, 62.1%, 60.53% for 2B, 9B and 27B versions, respectively) also exceed random guess and the zero-rate classification. Amongst the models pre-trained or fine-tuned on Norwegian Bokmål we observe that the best-performing model is *NorskGPT Mistral 7B* with an average accuracy of 30.5%.

4.2 RQ2: BRAGE Performance Compared to ScandEval

To put our BRAGE results into perspective, we have organized a selection of ScandEval benchmarks into a radar chart to visualize the differences. The radar chart shows the relative accuracy across five benchmarks, where each polygon represents a model, and the area of the polygon the model performance. Figure 5 shows the results for English and multilingual Base and Instruct models, whereas Figure 6 shows the Norwegian Instruct models (right) and their corresponding models used for fine-tuning (left).

Looking at Figure 5, *Gemma2 9B IT* stands out with the highest average BRAGE accuracy

Her kommer det en liste med produktkategorier hos **_brand_**:
 - Mobil: **_brand_** tilbyr mobilabonnementer med bred dekning, ulike datapakker og tilbud på siste telefonmodeller. Kategorien inneholder også datapakker og SIM-kort.
 - Forsikring: **_brand_** tilbyr forsikring for mobiltelefoner, som dekker tap, tyveri og skade, samt andre forsikringsprodukter gjennom samarbeidspartnere. Produktene er **_service_** og **_service_**. Kategorien inneholder også henvendelser relatert til forsikringssaker, som behandles i en egen avdeling. Kategorien skal ikke inneholde **_service_** **_service_**, som skal kategoriseres som Tjenester.
 - Annet: Kategorien når produkter ikke er spesifikt oppgitt i samtalen. Gjelder særlig ved samtaler som er brutte eller når kunden har ringt feil. I disse samtalenene blir det ikke snakket om verken produkttype eller abonnement.
 - E-post: **_brand_** leverer sikre og pålitelige e-posttjenester med funksjoner for personlig og profesjonell bruk, inkludert spamfiltrering og god brukervennlighet.
 - Bredbånd-mobilt: **_brand_** mobile bredbåndstjenester gir rask internettilgang på farten, eller installert på fast adresse med utvendig antenne. Kategorien inneholder produktene **_service_**, **_service_**, **_service_** og **_service_**.
 - Tjenester: **_brand_** tilbyr digitale tjenester slik som sikkerhetsløsninger og skytjenester. Eksempler på tjenester er **_service_**, **_service_**, **_service_**, **_service_**, **_service_**, **_service_**. I kategorien finnes også **_brand_** **_service_**, samt Trejepartstjenester som bl.a. omfatter innholdstjenester som **_service_**.
 - Bredbånd: **_service_** gir pålitelig internett med ulike hastighetsalternativer, kombinert med kundevennlig service og teknisk support. I kategorien finnes **_service_** og **_service_**.
 - TV: **_brand_** TV-tjenester inkluderer et utvalg av kanalpakker, strømmetjenester og muligheter for opptak, alt tilpasset kundens underholdningsbehov. Sentralt er produktet **_service_**, som er **_brand_** TV-løsning.

Her er tekst fra en samtale mellom kundeservice og en kunde. Angi hvilken produktkategori samtalen handler om, og svar kun med navnet på produktkategorien:

<transcript>

Figure 3: Anonymized version of the prompt used. The text in **bold blue** is the prompt instruction added to the original guidelines used by the annotators, and <transcript> indicate where the conversation transcript is inserted.

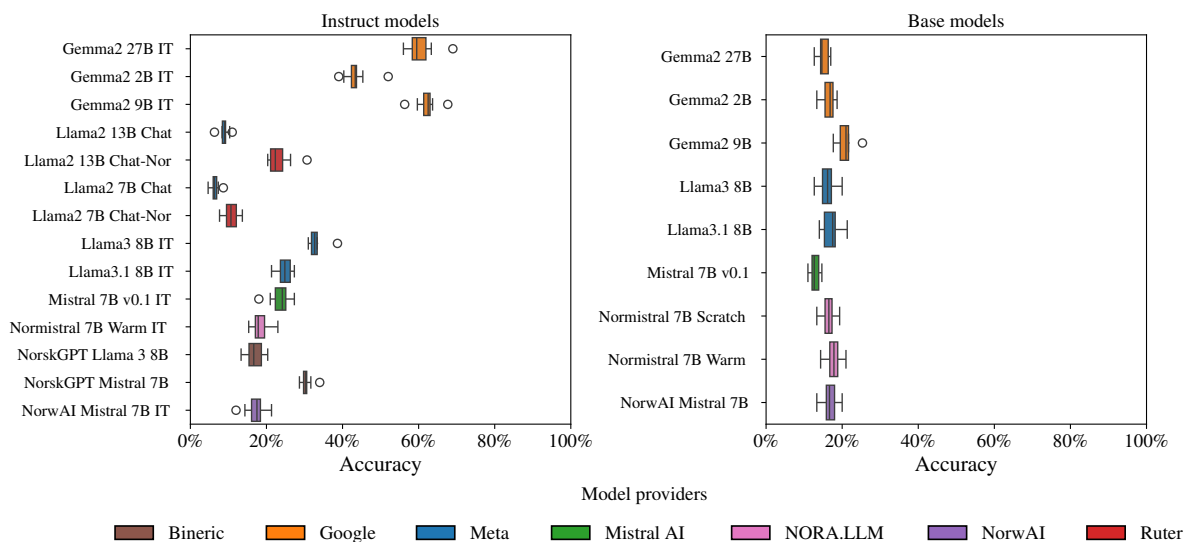


Figure 4: Comparison of accuracy distributions for instruction and base models. The color groupings separate them according to their respective provider/organization.

Benchmark /Metric(s)	BRAGE			ScandEval			
				NorNE-nb	NoReC	NorQuAD	HellaSwag
Model	Accuracy	Macro F1	MCC	Micro F1	Macro F1	F1	Accuracy
Category: IT							
Gemma2 27B IT	60.53 ± 2.31	50.56 ± 1.54	54.27 ± 2.13	56.75 ± 3.04	78.63 ± 0.96	73.41 ± 1.61	77.92 ± 1.72
Gemma2 2B IT	43.53 ± 2.15	32.62 ± 1.50	33.25 ± 1.87	28.77 ± 2.22	63.18 ± 1.91	63.84 ± 1.50	49.42 ± 0.79
Gemma2 9B IT	62.10 ± 1.80	53.50 ± 1.50	55.13 ± 1.91	44.91 ± 3.62	73.45 ± 0.94	70.14 ± 1.53	75.79 ± 1.47
Llama2 13B Chat	8.93 ± 0.77	6.12 ± 0.85	-0.69 ± 1.36	40.40 ± 2.79	57.45 ± 3.77	69.24 ± 2.68	41.00 ± 1.40
Llama2 7B Chat	6.40 ± 0.71	2.69 ± 0.41	0.08 ± 1.32	38.59 ± 2.84	57.09 ± 3.80	61.99 ± 2.34	31.84 ± 1.05
Llama3 8B IT	33.03 ± 1.34	26.44 ± 1.23	25.38 ± 1.67	65.57 ± 2.39	65.69 ± 3.50	69.90 ± 3.17	45.85 ± 1.93
Llama3.1 8B IT	24.87 ± 1.13	21.80 ± 1.36	16.38 ± 1.93	71.87 ± 0.97	71.58 ± 0.90	70.96 ± 3.00	54.03 ± 0.82
Mistral 7B v0.1 IT	23.60 ± 1.74	17.33 ± 1.81	8.86 ± 2.33	34.52 ± 1.17	60.88 ± 1.36	63.67 ± 2.98	35.89 ± 1.06
Category: IT + FNB							
Llama2 13B Chat-Nor	23.27 ± 1.98	19.79 ± 1.10	14.59 ± 2.05	47.74 ± 2.83	58.47 ± 3.79	65.76 ± 3.07	41.29 ± 1.19
Llama2 7B Chat-Nor	10.80 ± 1.14	8.69 ± 1.29	2.80 ± 1.51	20.44 ± 2.47	23.50 ± 3.03	50.11 ± 1.80	24.48 ± 0.70
NorskGPT Llama 3 8B	16.87 ± 1.51	15.14 ± 1.50	6.14 ± 1.90	60.25 ± 3.14	61.42 ± 3.56	74.57 ± 2.20	59.11 ± 2.44
NorskGPT Mistral 7B	30.50 ± 0.91	26.89 ± 1.05	22.54 ± 1.47	47.72 ± 3.74	70.81 ± 1.30	74.38 ± 3.92	60.59 ± 1.18
Category: P							
Gemma2 27B	15.07 ± 0.85	13.97 ± 0.96	6.08 ± 1.06	43.06 ± 1.89	76.14 ± 1.68	80.21 ± 4.49	63.55 ± 4.76
Gemma2 2B	16.43 ± 1.00	13.08 ± 1.02	4.98 ± 1.34	21.28 ± 2.58	47.91 ± 2.11	63.31 ± 3.73	28.89 ± 1.54
Gemma2 9B	20.80 ± 1.31	17.02 ± 1.20	7.51 ± 1.41	34.62 ± 1.80	75.53 ± 0.73	72.99 ± 3.16	63.52 ± 3.49
Llama3 8B	16.23 ± 1.35	14.16 ± 1.18	3.38 ± 1.57	47.65 ± 2.94	66.15 ± 1.44	74.98 ± 3.70	42.47 ± 2.74
Llama3.1 8B	17.07 ± 1.43	14.48 ± 1.49	4.36 ± 1.70	53.50 ± 3.27	68.71 ± 1.01	75.98 ± 2.62	46.84 ± 1.59
Mistral 7B v0.1	12.87 ± 0.74	10.31 ± 0.73	-0.20 ± 0.62	43.55 ± 2.21	64.53 ± 3.71	70.86 ± 2.79	32.43 ± 2.67
Category: PNB							
Normistral 7B Scratch	16.47 ± 1.09	13.89 ± 1.41	1.99 ± 1.51	15.44 ± 5.52	36.85 ± 2.01	38.93 ± 2.59	24.84 ± 0.71
Normistral 7B Warm	17.83 ± 1.29	13.76 ± 1.27	2.31 ± 1.54	31.45 ± 1.88	45.30 ± 3.46	61.85 ± 3.07	25.00 ± 0.83
NorwAI Mistral 7B	16.83 ± 1.19	12.55 ± 1.24	1.22 ± 1.46	20.45 ± 2.65	65.98 ± 2.95	68.04 ± 5.37	27.82 ± 1.56

Table 1: Aggregated performance metrics of BRAGE and a selection of ScandEval results from 10 runs. BRAGE performance is reported in Accuracy, Macro F1, and Mathews Correlation Coefficient (MCC), each with their corresponding $\pm 95\%$ confidence intervals (CI). ScandEval results include individual scores per benchmark and confidence intervals (see Nielsen, 2023; Nielsen et al., 2024). Model category abbreviations: Pre-trained on Norwegian Bokmål (PNB), Fine-tuned on Norwegian Bokmål (FNB), Pre-trained (P), Instruction-tuned (IT). The highest scores for each category are boldfaced.

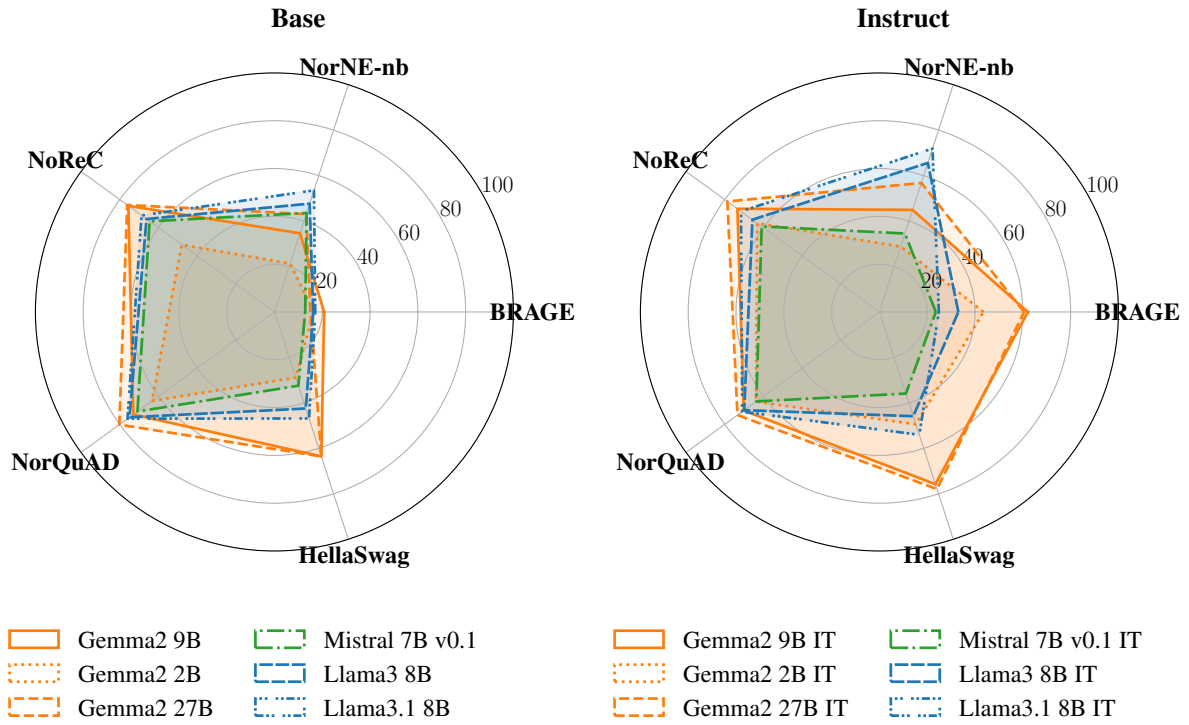


Figure 5: English/Multilingual models. Left: base models. Right: instruction-tuned. Radar chart of scores on selected ScandEval datasets and BRAGE results.

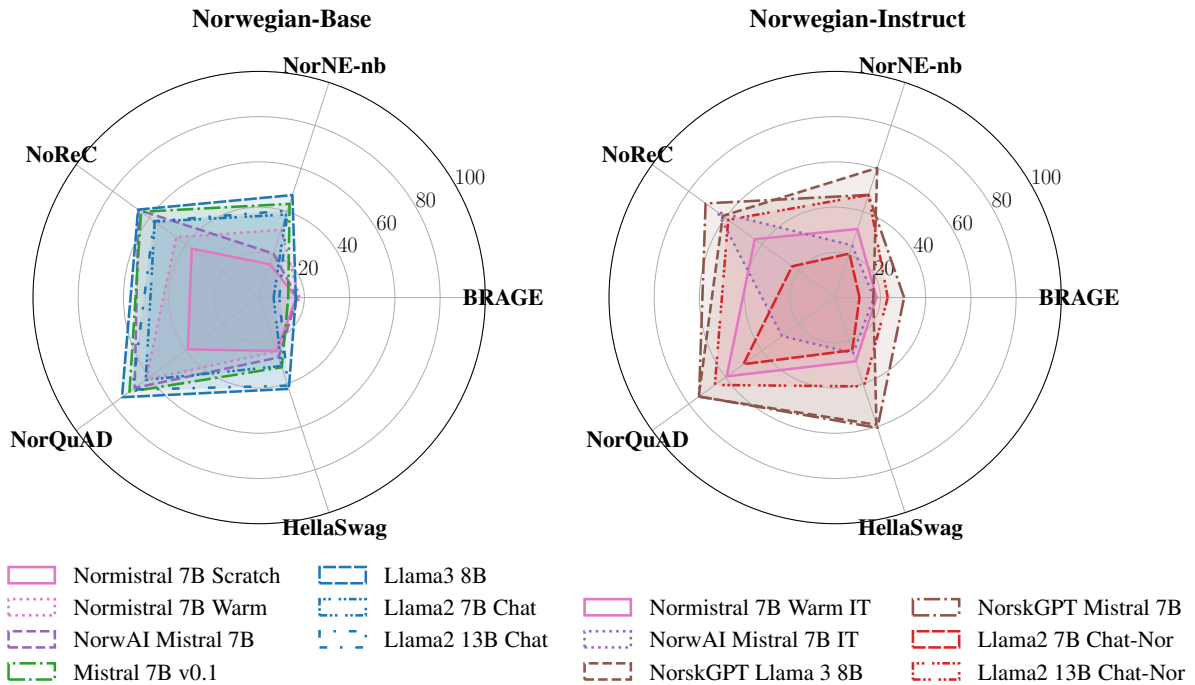


Figure 6: Norwegian models (and their corresponding base models used for fine-tuning). Left: base models. Right: fine-tuned on Norwegian. Radar chart of scores on selected ScandEval datasets and BRAGE results.

of 62.1%. Although somewhat below the 27B model for the ScandEval benchmarks, these findings are mostly consistent, except for the named entity task (NorNE-nb), where the Llama-models (e.g. *Llama3.1 8B IT*) surpass all other models, with a micro-F1 of 71.87 compared to 56.75 of the *Gemma2 27B IT*.

Moving on to the Norwegian models in Figure 6, pre-training (PNB) and fine-tuning (FNB) tend to lag behind the models fine-tuned on non-public data sources (IT + FNB). The architectural choices, and especially the fine-tuning procedures, seem to have a much higher importance for the BRAGE benchmark, as well as for HellaSwag (commonsense reasoning tasks), where we see a close relationship in terms of performance deltas: high scores on HellaSwag indicates high scores for BRAGE. In contrast, high scores on NorNE, for example, do not follow this pattern.

5 Discussion

Bigger is not Always Better

While larger models tend to get better results overall, we observe that *Gemma2 9B IT*, through its knowledge distillation training process (Team et al., 2024), approximates (and even outperforms) the larger version at 27B parameters, which is in alignment with other public benchmarks, such as *open llm leaderboard*.⁵ Moreover, the 8B Llama models perform well on several tasks, especially for named entity recognition, outscoring the larger models. These effects will likely become more prominent as smaller models are trained through knowledge distillation and fine-tuned on domain-specific tasks.

Instruction Tuning

Good results for instruction-tuned (IT) models on other benchmarks did not necessarily translate to BRAGE. We have noted the relation between HellaSwag for IT models, but the base models still achieve relatively high scores on all downstream tasks. In contrast, BRAGE requires specific fine-tuning to achieve good results, as exemplified by only the Gemma2 models reaching acceptable accuracy scores. This, too, is the case for the base model Mistral 7B v0.1 (score of 13.60) compared to 29.58 for the NorskGPT Mistral 7B, while increasing its HellaSwag score from

32.43 to 60.59, while other tasks remain nearly unchanged in comparison. The fine-tunes by NorwAI and NORA.LLM (PNB + FNB) have approximately equal scores and lower stability than the base models. Additionally, we believe some Norwegian models are fine-tuned using the presented datasets, which, in turn, results in poor generalizability. Note the high deviance by, e.g., *Norwai Mistral 7B (PNB)* scoring surprisingly high on NoReC and NorQuAD, but not NorNE. The opposite is the case for *Normistral 7B Warm (PNB)*.

Suggestions for Future Work

Evidently, modeling decisions, data, post-training fine-tuning, and alignment require extra attention. Few organizations share end-to-end details – besides the OLMo initiative (Groeneveld et al., 2024), and we are typically left with a higher-level view of potential improvements for future developments of LLMs. Based on our findings, the Gemma2 architecture seems suitable for most of our tests and public benchmarks, and we leave the following suggestions for language-specific LLM development in the post-training stage:

- Distillation to student distributions, keeping compute-optimal token counts in mind (Gu et al., 2023; Agarwal et al., 2024).
- Different reward setups through RLHF (Christiano et al., 2017) and other alignment procedures (Gao et al., 2024).
- Incorporating prompts from, e.g., LMSYS-CHAT-1M (Zheng et al., 2023a), with responses from larger teacher models.
- Studies on instruction formatting.

Business Perspectives

The potential value contribution of zero-shot LLM-based content classification to customer service operations is significant in terms of both user-friendliness and development time. However, our results suggest that the current performance level is not yet sufficient for full production deployment, suggesting a need for further research and development in this area.

Furthermore, the suggested approach relies on ground truth to assess model quality pre-production, only partially automating the content classification process. Finally, using highly resource-consuming LLMs for a task that can be

⁵https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard

solved using smaller, more energy-efficient models raises questions regarding sustainability and cost versus benefit (Rigutini et al., 2024).

6 Conclusion

We have presented BRAGE, a private zero-shot benchmark for classifying transcribed calls between customers and customer service. Based on these preliminary results, we observe that the task can be accomplished to a somewhat acceptable level using open-weight LLMs. Based on our results, we can conclude that this is a challenging benchmark and that instruction fine-tuned models generally perform better on this type of zero-shot task. Specifically, instruction fine-tuning (FNB) on a multilingual base model, in the case of *NorskGPT Mistral 7B*, was superior to any of the other Norwegian models on BRAGE. We, therefore, stress the importance of creating more open instruction datasets in Norwegian, as this might foster progress in zero-shot settings such as the BRAGE case. Surprisingly, we found that the *English-only* and significantly smaller *Gemma-2 2B IT* did better than any of the Norwegian models. These results may also apply to other European languages, especially those with a higher presence in multilingual training corpora, e.g., German and Spanish. We plan to expand this benchmark by adding new tasks as well as to include all of the Scandinavian languages.

7 Limitations

As these experiments were conducted on a real business case, relevant information, such as distribution details about our data, had to be left out due to its sensitivity. However, we hope BRAGE, as a private benchmark can still be a contribution to the academic community, when committing ourselves to share aggregated results with the public (keeping data private on local infrastructure) going forward. Our conclusions also remain limited by the amount of information publicly available on the models included in the study, we therefore specifically hope to see more published data concerning pre-training and instruction-tuning for the current and future research-funded models (e.g., by NORA.LLM and NorwAI).

8 Sustainability

We have tracked power consumption and estimated emissions for all experiments using Code-

Carbon (Schmidt et al., 2021). Hardware: 4x RTX 3090 GPUs over 29.2 hrs, resulting in a total emission of 0.8394 kgCO_{2e} given the energy mix in Oslo, Norway.

Acknowledgments

The authors would like to thank the team at the case company for their cooperation and contributions: Tom Arne Gjesdal Karlsen, Ole Jonas Liahagen, Stig Hodnebrog and Noman Bashir.

References

- Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. 2024. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*.
- Scott Barnett, Zac Brannelly, Stefanus Kurniawan, and Sheng Wong. 2024. Fine-tuning or fine-failing? debunking performance myths in large language models. *arXiv preprint arXiv:2406.11201*.
- Stella Biderman, Hailey Schoelkopf, Lintang Sutawika, Leo Gao, Jonathan Tow, Baber Abbasi, Alham Fikri Aji, Pawan Sasanka Ammanamanchi, Sidney Black, Jordan Clive, et al. 2024. Lessons from the trenches on reproducible evaluation of language models. *arXiv preprint arXiv:2405.14782*.
- Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.
- Nishanth Chandran, Sunayana Sitaram, Divya Gupta, Rahul Sharma, Kashish Mittal, and Manohar Swaminathan. 2024. Private benchmarking to prevent contamination and improve comparative evaluation of llms. *arXiv preprint arXiv:2403.00393*.
- Wei-Lin Chiang, Zhuohan Li, Zilong 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.
- Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30.
- C Lakshmi Devasena, T Sumathi, VV Gomathi, and M Hemalatha. 2011. Effectiveness evaluation of rule based classifiers for the classification of iris data set. *Bonfring International Journal of Man Machine Interface*, 1:5.

- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Jane Forman and Laura Damschroder. 2007. Qualitative content analysis. In *Empirical methods for bioethics: A primer*, pages 39–62. Emerald Group Publishing Limited.
- Bofei Gao, Feifan Song, Yibo Miao, Zefan Cai, Zhe Yang, Liang Chen, Helan Hu, Runxin Xu, Qingxiu Dong, Ce Zheng, et al. 2024. Towards a unified view of preference learning for large language models: A survey. *arXiv preprint arXiv:2409.02795*.
- Sreyan Ghosh, Chandra Kiran Reddy Evuru, Sonal Kumar, Deepali Aneja, Zeyu Jin, Ramani Duraiswami, Dinesh Manocha, et al. 2024. A closer look at the limitations of instruction tuning. *arXiv preprint arXiv:2402.05119*.
- Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. 2024. Olmo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838*.
- Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. 2023. Knowledge distillation of large language models. *arXiv preprint arXiv:2306.08543*.
- Sardana Ivanova, Fredrik Andreassen, Matias Jentoft, Sondre Wold, and Lilja Øvrelid. 2023. Norquad: Norwegian question answering dataset. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 159–168.
- Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.
- Fredrik Jørgensen, Tobias Aasmoe, Anne-Stine Ruud Husevåg, Lilja Øvrelid, and Erik Velldal. 2020. Norne: Annotating named entities for norwegian. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4547–4556.
- Yingcong Li, Muhammed Emrullah Ildiz, Dimitris Papailiopoulos, and Samet Oymak. 2023. Transformers as algorithms: Generalization and stability in in-context learning. In *International Conference on Machine Learning*, pages 19565–19594. PMLR.
- Marius Mosbach, Tiago Pimentel, Shauli Ravfogel, Dietrich Klakow, and Yanai Elazar. 2023. Few-shot fine-tuning vs. in-context learning: A fair comparison and evaluation. *arXiv preprint arXiv:2305.16938*.
- Dan Nielsen. 2023. ScandEval: A benchmark for Scandinavian natural language processing. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 185–201, Tórshavn, Faroe Islands. University of Tartu Library.
- Dan Saattrup Nielsen, Kenneth Enevoldsen, and Peter Schneider-Kamp. 2024. Encoder vs decoder: Comparative analysis of encoder and decoder language models on multilingual nlu tasks. *arXiv preprint arXiv:2406.13469*.
- NORA AI. 2024. Big steps towards a norwegian answer to chatgpt. Accessed: 2024-10-03.
- PwC. 2018. Experience is everything: Here’s how to get it right. Consumer intelligence series, PricewaterhouseCoopers.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust speech recognition via large-scale weak supervision. *arXiv preprint arXiv:2212.04356*.
- Mike Riess. 2022. Automating model management: a survey on metaheuristics for concept-drift adaptation. *Journal of Data, Information and Management*, 4:211–229.
- Leonardo Rigutini, Achille Globo, Marco Stefanelli, Andrea Zugarini, Sinan Gultekin, and Marco Fernandes. 2024. Performance, energy consumption and costs: A comparative analysis of automatic text classification approaches in the legal domain. *International Journal on Natural Language Computing*.
- ScandEval. 2024. Danish NLU. A Natural Language Understanding Benchmark.
- Victor Schmidt, Kamal Goyal, Aditya Joshi, Boris Feld, Liam Conell, Nikolas Laskaris, Doug Blank, Jonathan Wilson, Sorelle Friedler, and Sasha Lucioni. 2021. CodeCarbon: Estimate and Track Carbon Emissions from Machine Learning Computing.
- Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.
- Erik Velldal, Lilja Øvrelid, Eivind Alexander Bergem, Cathrine Stadsnes, Samia Touileb, and Fredrik Jørgensen. 2018. NoReC: The Norwegian Review Corpus. In *Proceedings of the 11th edition of the Language Resources and Evaluation Conference*, pages 4186–4191, Miyazaki, Japan.
- Yoav Wald, Amir Feder, Daniel Greenfeld, and Uri Shalit. 2021. On calibration and out-of-domain generalization. *Advances in neural information processing systems*, 34:2215–2227.
- Brandon T. Willard and Rémi Louf. 2023. Efficient guided generation for large language models.

- 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.
- Robert Wolfe, Isaac Slaughter, Bin Han, Bingbing Wen, Yiwei Yang, Lucas Rosenblatt, Bernease Herman, Eva Brown, Zening Qu, Nic Weber, and Bill Howe. 2024. Laboratory-scale ai: Open-weight models are competitive with chatgpt even in low-resource settings. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency*, volume 35 of *FAccT '24*, page 1199–1210. ACM.
- Ruijie Xu, Zengzhi Wang, Run-Ze Fan, and Pengfei Liu. 2024. Benchmarking benchmark leakage in large language models. *arXiv preprint arXiv:2404.18824*.
- Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. 2024. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities.
- Linyi Yang, Shuibai Zhang, Libo Qin, Yafu Li, Yidong Wang, Hanmeng Liu, Jindong Wang, Xing Xie, and Yue Zhang. 2022. Glue-x: Evaluating natural language understanding models from an out-of-distribution generalization perspective. *arXiv preprint arXiv:2211.08073*.
- Han Yu, Jiashuo Liu, Xingxuan Zhang, Jiayun Wu, and Peng Cui. 2024. A survey on evaluation of out-of-distribution generalization. *arXiv preprint arXiv:2403.01874*.
- Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.
- Tianrong Zhang, Bochuan Cao, Yuanpu Cao, Lu Lin, Prasenjit Mitra, and Jinghui Chen. 2024. Wordgame: Efficient effective llm jailbreak via simultaneous obfuscation in query and response.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Tianle Li, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zhuohan Li, Zi Lin, Eric Xing, et al. 2023a. Lmsys-chat-1m: A large-scale real-world llm conversation dataset. *arXiv preprint arXiv:2309.11998*.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023b. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.
- Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J. Zico Kolter, and Matt Fredrikson. 2023. Universal and transferable adversarial attacks on aligned language models.

A Translated Transcripts and Prompts

Figure 7 shows the translated call transcript, and Figure 8 shows the translated prompt template for transcripts.

Hi, you're speaking with _blacklist_. I have switched from private mobile subscription to having it covered by work, this happened about two and a half weeks ago, and then I see in the online bank that they have an invoice pending approval for January. Yes, yes. _number_. December, that's when mobing of mobile subscription, the private one, was supposed to be terminated, so I was wondering now can I check that the billing has been correct. _blacklist_. Yes, I'll help you with that, can I have your last name, date of birth and address. _blacklist_. _name_, _blacklist_. _blacklist_, _blacklist_. _blacklist_. _blacklist_. at _blacklist_. _blacklist_. Yes and postal code A? _number_. _number_. _number_. _blacklist_. _blacklist_, but it was a phone, yes, yes, you're wondering about the invoib, so you had an outstanding invoice, you said. Yes, in the bank I put an invoice pending approval for January. Yes. For _name_ you said no, it's paid, the invoice is paid, yes. _number_. _number_. and _blacklist_. comma _blacklist_. Okay. But when I put down that invoice, it says it's for January. Yes, then you will get back the entire monthly fee actually since your subscription was changed, it was changed before January started, so you get everything back, yes, that's it, so you will actually get back let's see _number_. _number_. and and _blacklist_. to the same account number you last paid with. Okay, yes, so then, then it was kind of terminated. Yes. Okay. Yes. For _number_. kroner, so then we deduct that or _blacklist_, but _blacklist_. yes. That, yes. Just set it aside. Okay, yes for _blacklist_, so then, then we'll receive a final invoice then. Alright, then I'll forget about that, fine goodbye. Goodbye, you too.

Figure 7: English translation of a modified call example with similar quality as the transcripts in our dataset. The topic of this call is 'Mobile'. The terms _blacklist_, _name_ and _number_ are anonymized entities.

Here comes a list of product categories at _brand_:
 \n - Mobile: _brand_ offers mobile subscriptions with broad coverage, various data packages and offers on latest phone models. The category also includes data packages and SIM cards.
 \n \n - Insurance: _brand_ offers insurance for mobile phones, covering loss, theft and damage, as well as other insurance products through collaboration partners. The products are _service_ and _service_. The category also includes inquiries related to insurance cases, which are handled in a separate department. The category should not include _service_ _service_, which should be categorized as Services.
 \n \n - Other: The category when products are not specifically mentioned in the conversation. Applies particularly to conversations that are broken or when the customer has dialed wrong. In these conversations, neither product type nor subscription is discussed.
 \n \n - Email: _brand_ delivers secure and reliable email services with features for personal and professional use, including spam filtering and good user-friendliness.
 \n \n - Broadband-mobile: _brand_ mobile broadband services provide fast internet access on the go, or installed at a fixed address with external antenna. The category contains the products _service_, _service_, _service_ and _service_.
 \n \n - Services: _brand_ offers digital services such as security solutions and cloud services. Examples of services are _service_, _service_, _service_, _service_, _service_, _service_. The category also includes _brand_ _service_, as well as Third-party services which include content services like _service_.
 \n \n - Broadband: _service_ provides reliable internet with various speed options, combined with customer-friendly service and technical support. The category includes _service_ and _service_.
 \n \n - TV: _brand_ TV services include a selection of channel packages, streaming services and recording options, all adapted to the customer's entertainment needs. Central is the product _service_, which is _brand_'s TV solution.
 \n \n **Here is text from a conversation between customer service and a customer. Indicate which product category the conversation is about, and respond only with the name of the product category:**
 \n <transcript>

Figure 8: English translation of the anonymized version of the prompt used. The text in **bold blue** is the prompt instruction added to the original guidelines used by the annotators, and <transcript> indicate where the conversation transcript is inserted.

Mixed Feelings: Cross-Domain Sentiment Classification of Patient Feedback

Egil Rønningstad*, Lilja Charlotte Storset*, Petter Mæhlum, Lilja Øvrelid, Erik Velldal

Department of Informatics, University of Oslo, Norway

{egilron, liljacs, pettemae, liljao, erikve}@uio.no

Abstract

Sentiment analysis of patient feedback from the public health domain can aid decision makers in evaluating the provided services. The current paper focuses on free-text comments in patient surveys about general practitioners and psychiatric healthcare, annotated with four sentence-level polarity classes – positive, negative, mixed and neutral – while also attempting to alleviate data scarcity by leveraging general-domain sources in the form of reviews. For several different architectures, we compare in-domain and out-of-domain effects, as well as the effects of training joint multi-domain models.

1 Introduction

Sentiment analysis (SA), the computational analysis of opinions and emotions expressed in text, is one of the applications of natural language processing (NLP) that have found the most widespread use across many different areas, including medical domains (Yadav et al., 2018). As the task is mostly approached as one of supervised learning, access to sufficient amounts of labeled data is the main driver of performance. However, as manual annotation is costly, labeled data also represents a main bottleneck. For this reason it is typically desirable to be able to reuse existing resources when developing SA tools for a new area of application. Unfortunately, domain-sensitivity is a well-known effect across many different NLP tasks. Models trained on data from one domain (or genre or text-type) often underperform when applied to another due to variations in language use, terminology, and contextual nuances (Al-Moslmi et al., 2017; Gräßer et al., 2018).

*The authors contributed equally.

This paper investigates cross-domain effects in polarity classification of public health data, more specifically free-text comments from patient surveys for general practitioners and psychiatric healthcare providers. We here investigate the usefulness of data from a different domain and genre, i.e. professionally authored reviews collected from Norwegian news publishers. The datasets are annotated at the sentence level with the same four-class polarity labels; positive, negative, mixed, and neutral. In the following, we compare non-neural and neural architectures in both in-domain and cross-domain settings with the goal of providing high-quality sentiment analysis for Norwegian patient comments.

2 Datasets

We here briefly describe the two annotated SA datasets that form the basis of our experiments, also discussing some of their key differences.

NorPaC For the health domain we will be using a dataset introduced by Mæhlum et al. (2024), comprising free-text comments from surveys conducted by the Norwegian Institute of Public Health (NIPH), as part of their so-called patient-reported experience measures (PREMs). The dataset is dubbed NorPaC – short for Norwegian Patient Comment corpus – and comprises two related subdomains, corresponding to feedback on General Practitioners (GPs) and Special Mental Healthcare (SMH), with a total of 7693 sentences (4002 from GP and 3691 from SMH) annotated for polarity.

The NorPaC dataset is a valuable accession to Norwegian corpora, as it gives valuable insights to the national public health system. The texts are written by patients after encounters with the healthcare system, and gives rise to language with an everyday character, such as sentences with a conversational tone or even incomplete sentences and spelling mistakes. Example 1 shows a positive

patient feedback sentence that is written solely in capital letters, in addition to containing a typing mistake in the personal pronoun *jeg*, 'I'. Example 2 shows a negative review with a colloquial tone, containing three exclamation marks at the end of the utterance.

- (1) *FIKK HENVISNING DA JGE BA*
Got referral when (I) asked
OM DET, OG GÅR STADIG TIL
about it, and goes constant to
UTREDNING DER.
examination there.
'Got a referral when I asked for it, and am constantly going for examination there.'
- (2) *Det er for dårlig!!!*
It is too bad!!!
'It is too bad!!!'

NoReC The Norwegian Review Corpus (NoReC; Velldal et al., 2018) comprises full-text reviews collected from major Norwegian news sources, covering a range of different domains (movies, music, literature, restaurants, various consumer products, etc.). We here use a version dubbed NoReC_{fine} (Øvrelid et al., 2020), a subset of roughly 11,000 sentences across more than 400 reviews with fine-grained sentiment annotations, here aggregated to the sentence-level (Kutuzov et al., 2021) using the above-mentioned label set of four classes.¹ In contrast to NorPaC, the reviews are written by professional authors, meaning more creative writing but with sentences that are typically complete and grammatically correct.

- (3) *Den er en pølse i salatens tid,*
It is a sausage in salad's.the time,
en slags mumlemanisk
a kind.of mumblemaninc
manns-modernitets-manifestasjon
man-modernity-manifestation
'It is a sausage in the age of the salad, a kind of mumble-manic male-modernity-manifestation'

Example 3 shows one of many creative sentences in NoReC. *En pølse i salatens tid*, 'a sausage in the age of the salad', is a figurative way to emphasize the fact that this movie is not among the trendy, i.e. 'the salad', but rather acts like 'a sausage'. Further, the author describes

the movie as a *mumlemanisk manns-modernitets-manifestasjon*, 'mumble-manic male-modernity-manifestation'. This exemplifies the complexity of many of the texts in the NoReC dataset where authors may construct new and creative expressions.

Genre and text type The two datasets can be said to be found at opposite ends in terms of language and writing style. In contrast to the professionally authored reviews in NoReC, containing grammatically correct texts with higher complexity and creativity, the NorPaC patient comments consist of more colloquial language. It also comes with many of the other hallmarks of user-generated content, such as more frequent spelling mistakes and incomplete sentences, as well as unorthodox use of case and punctuation. While such properties will generally contribute to increasing the vocabulary size, NoReC still contains almost three times as many unique lemmas as NorPaC, due to the fact that it contains more creative and varied language (with a higher degree of figurative expressions, etc.), as mentioned above, in addition to covering multiple domains.

Class distribution Table 1 summarizes some relevant statistics for the two corpora, showing the number of examples across the four classes, as well as average token length of sentences.

For the NoReC reviews, we see that we have many more examples for the positive than the negative category. For the NorPaC patient feedback, in contrast, the negative category is notably larger, although the number of positive and negative examples are more balanced than in NoReC.

Another striking difference is the much higher ratio of neutral sentences in NoReC compared to NorPaC; 47% vs. 12%, respectively. This is not surprising if we consider the genre differences; professional reviews need to provide a lot of non-sentiment bearing background and descriptions of the object under review. The ratio of sentences with mixed polarity, however, is similar across the datasets, and is also the smallest sentiment class.

Related to the class distribution, we also observe some interesting differences with respect to the average token length of sentences. While the length is the same across the positive and negative sentences in the NoReC reviews, the length of negative sentences in the NorPaC patient comments tend to be substantially longer than the positive ones. However, for both datasets we see that neu-

¹https://huggingface.co/datasets/lmg/norec_sentence

		Positive	Negative	Neutral	Mixed	Total
GP	Sentences	1265 (32%)	1903 (48%)	654 (16%)	174 (4%)	4002
	Avg. tokens	11.8	15.61	10.38	19.99	13.81
SMH	Sentences	1524 (41%)	1604 (44%)	291 (8%)	266 (7%)	3691
	Avg. tokens	13.1	18.48	10.53	23.68	15.94
NorPaC (GP+SMH)	Sentences	2789 (36%)	3507 (46%)	945 (12%)	440 (6%)	7693
	Avg. tokens	12.53	17.03	10.78	22.29	14.93
NoReC	Sentences	3514 (31%)	1663 (15%)	5393 (47%)	867 (8%)	11437
	Avg. tokens	18.57	18.18	13.78	25.92	16.78

Table 1: For each polarity class we show the distribution of number of sentences and average sentence length across the GP and SMH datasets within NorPaC, and for the NoReC dataset.

tral sentences tend to be shorter, while the mixed class displays substantially longer average length, which intuitively makes sense given that they per definition must express at least two opposing sentiments.

3 Experimental results

Below we report experimental results for a range of different models and architectures on the datasets described above. We start by providing details about the models and the experimental set-up, before discussing the results for both in-domain and cross-domain classification.

3.1 Models and experimental set-up

The NorBERT3 series of models (Samuel et al., 2023; Kutuzov et al., 2021) represent the 3rd generation of pre-trained Norwegian masked language models (MLMs) based on the BERT transformer architecture (Devlin et al., 2019). We fine-tuned text classifiers for two different sizes of NorBERT3 – Base and Large – with 123M and 353M parameters, respectively. GPU memory requirements were 8 and 35 GB. The NorT5 (Samuel et al., 2023) models are pretrained on the same Norwegian data as NorBERT3, and we fine-tune NorT5 Large to generate sentiment labels as a sequence-to-sequence task. NorT5 Large has 808M parameters. During fine-tuning with a batch size of 24, 71GB GPU memory was used. For all these models we report the mean weighted average F_1 over 3 runs. More details of the hyperparameter search are found in Appendix A. As a baseline, we also train a Support Vector Machine (SVM) model with a linear kernel and bag-of-words features.² The random baseline for the task yields an

²The features correspond to the full vocabulary of the tokenized texts for each corpora, as preliminary experiments

F_1 -score of between 22% and 23% for all training datasets, averaged across 1000 runs.

For NoReC we use the predefined data split, with 80-10-10 percentages respectively for the training, validation and test set. We define a similar split for NorPaC, randomly selected on the comment-level to make sure sentences from the same comment are not separated across splits, while also ensuring a balanced class distribution.

3.2 In-domain patient comment results

Table 2 shows results when training and testing on sentences from the NorPaC corpus. While the main focus of this section is to assess the in-domain performance of models trained on the NorPaC patient comments, recall that this corpus comprises two different sources; feedback regarding General Practitioners (GPs) and Special Mental Healthcare (SMH). We therefore also report results for training and testing on data from the individual sources separately – including cross-source training and testing.

We see that training on GP yields very strong test results: Not only are in-domain results for training and testing for SMH lower, but test results on SMH are competitive when training on GP. In the same vein, we see that for most models, joint training on the entire NorPaC data boosts results for SMH, with the only exception being NorBERT3 Large, where the best results for SMH are actually found when training on GP only (although the differences are marginal). In sum, we find that, within the NorPaC domain(s), the generalization capabilities of the GP-trained models

showed that best results were obtained without any feature selection or weighting (i.e. no TF-IDF, frequency cutoffs, etc.). The number of features range from approximately 5K for the GP/SMH models, through 8K for the full NorPaC data and 22K for NoReC, and finally 27K for NorPaC+NoReC.

Model	Train	Test		
		GP	SMH	NorPaC
SVM (BoW)	GP	63.65	66.42	64.96
	SMH	57.86	66.77	62.26
	NorPaC	62.90	68.34	65.52
NorBERT3 (Base)	GP	84.13	82.02	83.14
	SMH	79.43	82.96	81.22
	NorPaC	83.61	83.34	83.49
NorBERT3 (Large)	GP	85.79	84.85	85.41
	SMH	81.23	84.61	82.95
	NorPaC	86.00	84.28	85.22
NorT5 (Large)	GP	84.34	83.65	84.08
	SMH	81.03	84.24	82.70
	NorPaC	85.03	85.05	84.54

Table 2: Results for training and testing on the GP and SMH datasets within NorPaC.

are so good that the benefit of joint training on GP and SMH are less than anticipated. One contributing factor here might be that the GP data overall is written in a more explicit and straightforward manner compared to the SMH data, which might contain parts that are perceived as noisy for the model. Hence, training on GP and testing on PHV yields better results than vice versa. Finally, and as expected, we see that the neural models outperform the SVM model and that larger models generally tend to outperform smaller ones, although NorT5 Large actually tends to be outperformed by the smaller NorBERT3 Large model.

3.3 Cross-domain results

Table 3 shows results for several combinations of training and testing on both NorPaC and NoReC. First, we note that the in-domain results for NorPaC are substantially higher than the in-domain results for NoReC. This makes sense, given that NoReC in practice covers many different domains and has a much more diverse vocabulary than NorPaC. This observation most likely also has bearings on the cross-domain results, where we see a smaller relative drop in performance when testing the NoReC-trained models on NorPaC, than vice versa. Another contributing factor to the (expected) drops in performance for the cross-domain results can be the differences in the class distribution for the two datasets, as discussed above.

Turning to the joint training on the combination of NoReC and NorPaC, we again see that the

Model	Train	Test	
		NorPaC	NoReC
SVM (BoW)	NorPaC	65.52	37.84
	NoReC	42.11	54.42
	NorPaC+NoReC	66.20	53.35
NorBERT3 (Base)	NorPaC	83.49	59.09
	NoReC	68.03	75.63
	NorPaC+NoReC	83.71	76.14
NorBERT3 (Large)	NorPaC	85.22	59.19
	NoReC	66.38	78.88
	NorPaC+NoReC	85.03	78.40
NorT5 (Large)	NorPaC	84.54	58.14
	NoReC	70.88	76.73
	NorPaC+NoReC	85.06	75.79

Table 3: Results for training and testing on sentences from both the NorPaC patient comments and the NoReC reviews.

test scores are substantially higher on NorPaC than NoReC for all models. For the NorBERT3 Base model, the joint training improves results across both datasets. However, for NorBERT3 Large, we see that the in-domain variants gives the highest scores for both datasets, but only by a small margin. For the SVM model, we see the same tendency with in-domain training on NoReC, yielding slightly better performance than joint training.

In an error analysis of in-domain vs. out-of-domain results for NorBERT3 Large evaluated on the NorPaC test set, we observe that the model trained on NorPaC is better at predicting negative sentences, compared to the model trained on NoReC. Here, the in-domain model classifies 92% of the negative samples correctly, whereas the out-of-domain model only identifies 39% of them. Out of the true negative samples, the NoReC-trained model predicts 59% of them as neutral. We believe the prediction of the negative class is the largest contributor to the lower performance of the NoReC-trained model, as this class makes up 46% of the NorPaC test set. However, there is one class for which this model performs slightly better than the in-domain model. As we recall from Table 1, the neutral class is the largest class in the NoReC dataset. This is most likely the reason why the NoReC-trained model classifies 95% of these instances in the NorPaC test set correctly, as opposed to the NorPaC trained model, which correctly classifies 69% of them. In sum, a closer

look at the per-class results reveals clear effects of the class distribution in the training set on model performance.

Learning curves for in-domain data To gauge the effect of the number of in-domain training examples, we computed learning curves where models are trained on partitions that are created by successively halving the NorPaC training set, with and without including the full NoReC training data. Figure 1 plots the effect on fine-tuning NorBERT3 Large. Utilizing only 6.25% (386 samples) of the NorPaC training set we find a strong performance gain of adding the cross-domain NoReC dataset. The effect is reduced, but present up to 50% (3087 samples). However, with the full NorPaC training set containing 6175 samples, we find that adding cross-domain data is harmful for the model performance. This shows how cross-domain data can help when in-domain datasets are small, but should not be added indiscriminately.

4 Summary

This paper has reported experimental results for polarity classification of sentences in a Norwegian dataset dubbed NorPaC, comprising free-text comments from patient surveys collected as part of evaluating public healthcare services. In addition to assessing cross-domain effects between two healthcare sub-domains – feedback on general practitioners and psychiatric healthcare – we have also assessed the effect of leveraging general-domain sentiment annotations, based on the NoReC review data. Rather than just annotating the simple binary classification of positive/negative sentences, our datasets additionally indicate both neutral and mixed sentences. We show how several of our tested model configurations surpass 85% weighted F_1 for this four-class set-up. We also show how including out-of-domain data improves model performance when in-domain data is limited, but that better performance can be achieved with in-domain data alone once the the amount of annotated data crosses a critical threshold. Our analyses give new insights into both the NorPaC and NoReC datasets, including the differences and similarities between them.

Acknowledgments

This work was supported by two research projects funded by the Research Council of Norway

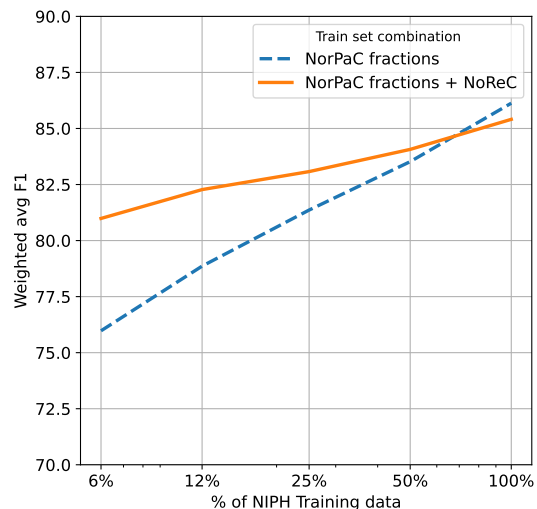


Figure 1: Learning curves, for two configurations: **NorPaC fractions**: The model is trained on fractions of the NorPaC training split, from 6.25% \approx 6% (386 samples) successively doubling the training set up to the full NorPaC training split. **NorPaC fractions + NoReC**: The same fractions of the NorPaC training split, mixed with the full NoReC training split. All evaluations are on the full NorPaC test set, averaged over three runs with different seeds, and with the amounts of in-domain training data shown on log-scale.

(RCN), namely ‘Sentiment Analysis for Norwegian Text’ (SANT), funded by an IKTPLUSS grant from RCN (project no. 270908), and ‘Strengthening the patient voice in health service evaluation: Machine learning on free-text comments from surveys and online sources’, funded by a HELSEVEL grant from RCN (project no. 331770). Moreover, the computations were performed on resources provided through Sigma2 – the national research infrastructure provider for High-Performance Computing and large-scale data storage in Norway.

References

Tareq Al-Moslmi, Nazlia Omar, Salwani Abdullah, and Mohammed Albared. 2017. Approaches to cross-domain sentiment analysis: A systematic literature review. *IEEE Access*, 5:16173–16192.

- 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.
- Felix Gräßer, Surya Kallumadi, Hagen Malberg, and Sebastian Zaunseder. 2018. Aspect-based sentiment analysis of drug reviews applying cross-domain and cross-data learning. In *ACM International Conference Proceeding Series*, volume 2018-, pages 121–125, New York, NY, USA. ACM.
- Andrey Kutuzov, Jeremy Barnes, Erik Velldal, Lilja Øvrelid, and Stephan Oepen. 2021. Large-scale contextualised language modelling for Norwegian. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 30–40, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- Petter Mæhlum, David Samuel, Rebecka Maria Norman, Elma Jelin, Øyvind Andresen Bjertnæs, Lilja Øvrelid, and Erik Velldal. 2024. It’s difficult to be neutral – human and LLM-based sentiment annotation of patient comments. In *Proceedings of the First Workshop on Patient-Oriented Language Processing (CL4Health) @ LREC-COLING 2024*, pages 8–19, Torino, Italia. ELRA and ICCL.
- Lilja Øvrelid, Petter Mæhlum, Jeremy Barnes, and Erik Velldal. 2020. A fine-grained sentiment dataset for Norwegian. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5025–5033, Marseille, France. European Language Resources Association.
- David Samuel, Andrey Kutuzov, Samia Touileb, Erik Velldal, Lilja Øvrelid, Egil Rønningstad, Elina Sigdel, and Anna Palatkina. 2023. NorBench – a benchmark for Norwegian language models. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 618–633, Tórshavn, Faroe Islands. University of Tartu Library.
- Erik Velldal, Lilja Øvrelid, Eivind Alexander Bergem, Cathrine Stadsnes, Samia Touileb, and Fredrik Jørgensen. 2018. NoReC: The Norwegian review corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Shweta Yadav, Asif Ekbal, Sriparna Saha, and Pushpak Bhattacharyya. 2018. Medical sentiment analysis using social media: Towards building a patient assisted system. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

A Hyperparameter tuning for NorBERT3-based models

We chose NorBERT3 base and large as the models to fine-tune for the text classification task. This model series has proven to perform well on previous comparisons for sentiment analysis on Norwegian sentences (Samuel et al., 2023). In order to find the best hyperparameters for our task, we first experimentally determine the best combination of learning rate and batch size. Table 4 shows the results for the two model sizes. All experiments are evaluated by accuracy on the development split, using the best of 10 epochs and one seed per hyperparameter combination. With the best performing settings for learning rate and batch size, we further search for improved performance by adjusting dropout in the classifier head, warm-up ratio and weight decay during fine-tuning. The search space for these hyperparameters are shown in Table 5. The best performing settings are shown in Table 6. The final choice of hyperparameters are shown in Table 7.

Model	lr	16	32	64
base	1e-05	78.28	77.82	77.76
base	2e-05	77.92	78.12	77.79
base	5e-05	76.09	77.04	78.18
large	1e-05	80.44	81.12	80.37
large	2e-05	80.96	81.35	80.89
large	5e-05	79.23	80.44	80.60

Table 4: Learning rate and batch size hyperparameter search for NorBERT3-base and large.

Model	Search space
classifier dropout	[0.05, 0.1, 0.25, 0.4]
warm-up ratio	[0.01, 0.05, 0.1, 0.2]
weight decay	[0.001, 0.01, 0.1]

Table 5: Search space for classifier dropout, warmup ratio and weight decay for NorBERT3 base and large, after best learning rate and batch size was identified.

Model	Dropout	Wu_ratio	W_decay	Dev acc.
base	0.25	0.20	0.010	78.77%
base	0.10	0.20	0.010	78.71%
base	0.25	0.20	0.100	78.58%
large	0.25	0.10	0.100	82.10%
large	0.25	0.10	0.001	81.91%
large	0.40	0.20	0.001	81.84%

Table 6: Top-3 performing models, for NorBERT3 base and large, when searching for optimal parameters for classifier dropout, warm-up ratio and weight decay.

Model	Base	Large
batch size	16	32
learning rate	1e-05	2e-05
classifier dropout	0.25	0.25
warmup ratio	0.20	0.10
weight decay	0.01	0.10

Table 7: Final hyperparameters selected for the NorBERT3 base and large finetuning, as informed by our hyperparameter search. Other hyperparameters are left as their defaults.

The Impact of Copyrighted Material on Large Language Models: A Norwegian Perspective

Javier de la Rosa¹ Vladislav Mikhailov² Lemei Zhang³ Freddy Wetjen¹ David Samuel²
Peng Liu³ Rolv-Arild Braaten¹ Petter Mæhlum² Magnus Breder Birkenes¹
Andrey Kutuzov² Tita Enstad¹ Hans Christian Farsethås² Svein Arne Brygfjeld¹
Jon Atle Gulla³ Stephan Oepen² Erik Velldal² Wilfred Østgulen¹ Lilja Øvrelid²
Aslak Sira Myhre¹

¹National Library of Norway

²University of Oslo

³Norwegian University of Science and Technology

Correspondence: versae@nb.no

Abstract

The use of copyrighted materials in training language models raises critical legal and ethical questions. This paper presents a framework for and the results of empirically assessing the impact of publisher-controlled copyrighted corpora on the performance of generative large language models (LLMs) for Norwegian. When evaluated on a diverse set of tasks, we found that adding both books and newspapers to the data mixture of LLMs tend to improve their performance, while the addition of fiction works seems to be detrimental. Our experiments could inform the creation of a compensation scheme for authors whose works contribute to AI development.

1 Introduction

Generative language models have radically reshaped the landscape of natural language processing (NLP), enabling the development of systems that can generate and interact with human language at an unprecedented level. This includes Norwegian, for which several large language models (LLMs) have been trained and published in the recent years using different architectures and licensing choices (Kummervold et al., 2021; Kutuzov et al., 2021; Samuel et al., 2023, 2025; Liu et al., 2024).

However, the vast quantities of data required for training these models often include copyrighted materials, presenting novel challenges related to

intellectual property rights and compensation. Additionally, prior research has highlighted significant concerns about dataset composition and quality in large-scale web-crawled datasets, emphasizing the need for more responsible data curation practices (Kreutzer et al., 2022; Artetxe et al., 2022; Penedo et al., 2024). Together, these challenges have led to numerous lawsuits across jurisdictions, fundamentally questioning the legitimacy of training models on copyrighted data without explicit permissions from content creators (Panettieri, 2024; Madigan, 2024; Weisenberger et al., 2024).¹

The first wave of lawsuits emerged shortly after the public release of advanced generative AI models (see Appendix A). Content creators, including authors, visual artists, and musicians, began to express concerns about the unauthorized use of their work in training datasets. Multiple class-action lawsuits were filed in the United States, accusing prominent AI companies such as OpenAI and Meta Platforms of infringing on copyright laws by using copyrighted materials without obtaining explicit permissions. The authors argued that the unauthorized use of their works without any form of compensation or recognition undermines their intellectual property rights and jeopardizes their ability to earn a living from their creative endeavors. In Europe, a coalition of news publishers has taken legal action against Google and Meta Platforms, arguing that the use of journalistic content in training models without fair re-

¹See Gervais et al. (2024) for an in-depth introduction on how LLMs are being interpreted in the legal domain.

muneration constitutes a breach of copyright and undermines the sustainability of high-quality journalism. Likewise, Norwegian rights-holder organizations representing publishing houses across the country, contacted the government in late 2023 expressing their concerns over the use of their material in training generative language models and demanding some sort of compensation were their contents to be used in the training of generative language models. As a result, the Ministry of Culture and Equality (*Kultur- og likestillingsdepartementet*) instructed the National Library to create a data-driven report they could use in order to make informed decisions in the elaboration of a compensation scheme for the authors. Led by the National Library of Norway, a consortium was formed together with the University of Oslo and the Norwegian University of Science and Technology under the umbrella of the so-called Mimir Project.²

In this context, and under the umbrella of Mimir, this paper describes a first attempt at empirically evaluating the impact of copyrighted content in the training of LLMs for Norwegian. We introduce a set of carefully curated datasets that are later used in the training of foundational, domain-tuned, and instruction-tuned models. We establish the proper training conditions to be able to compare models trained on the different datasets. A newly created benchmarking suite is used to evaluate the performance of each individual model and make the comparison meaningful. As a collaborative effort among several institutions, the results of our investigations set the basis to guide policymaking and proper compensation schemes for authors and right-holders in Norway (*Nasjonalbiblioteket*, 2024).

2 Methodology

The methodology involves a comprehensive analysis that spans several stages. Initially, a diverse corpus of primarily Norwegian language data is assembled, incorporating both copyrighted and non-copyrighted materials, plus materials commonly found on the Internet. This corpus serves as the foundation for training various LLMs, each with different configurations and access levels to copyrighted content. By comparing the performance of these models across a range of linguistic and natural language processing tasks, such as text

generation, translation, summarization, question-answering, sentiment analysis and more, we seek to quantify the specific contributions of copyrighted materials to the overall model quality.

To ensure robustness and reliability, the evaluation framework focuses on generation capabilities, natural language understanding, and linguistically-inspired metrics. Quantitative measures include traditional NLP metrics like accuracy, F1, BLEU, and ROUGE, which provide assessments of model accuracy and fluency. Linguistic analysis, on the other hand, involves assessing the coherence, language variability, and contextual relevance of the generated outputs. This dual approach allows for a nuanced understanding of how copyrighted materials impact the performance and utility of LLMs.

3 Data Collection

With the objective of setting up a realistic training scenario where using Internet crawled sources is commonplace, we gathered publicly available text collections like Wikipedia, datasets from the HPLT (*de Gibert et al.*, 2024) and CulturaX (*Nguyen et al.*, 2024) projects, code in different programming languages from *Lozhkov et al.* (2024), governmental reports and publications published under open licenses, and books and newspapers articles in the public domain.

We then collaborated with the National Library of Norway and the rights-holder organizations to gain access to protected materials. Through the legal deposit act, the National Library of Norway has digitized almost all books in Norwegian and around 85% of the newspapers ever published in the country (*Nasjonalbiblioteket*, 2024). Where the quality of the digitized material was not enough (e.g., due to OCR processing), or was not been legally deposited (e.g., paywalled content), specific agreements were put in place to obtain the material from third party organizations such as the Norwegian Broadcasting Corporation (NRK), the TV channel TV2, and the newspaper conglomerates Amedia and Schibsted. In line with provisions that allow research on language technology and data mining (*Åndsverkloven*), and with the consent of the Norwegian right-holders, this study primarily relied on material legally deposited at, or under agreement with, the National Library of Norway. Specifically, we focus our study on the collection of publisher-controlled books and newspapers ar-

²A name chosen after a figure in Norse mythology renowned for his knowledge and wisdom.

Dataset	Documents	Words
base	60,182,586	40,125,975,241
extended	125,285,547	82,149,281,266

Table 1: Number of documents and words in each of the core datasets. Words refer to whitespace-separated sub-strings.

ticles.

3.1 Core Datasets

This mixture of data (see Figure 1 and Appendix C) allowed us to evaluate the impact of high-quality publisher-controlled copyright-protected corpora versus other sources commonly available on the Internet. The models trained on the copyrighted materials will not be made publicly available for further use and only serve the purpose of this study.

We followed the recipe of the Norwegian Colossal Corpus (NCC) by Kummervold et al. (2022), adapting and updating it with new up-to-date contents, re-OCRing some materials, enriching their metadata, and ensuring uniform format and functionality across datasets. The preparation involved cleaning, deduplication, metadata tagging, and language balancing to maintain consistent representation of Norwegian, preventing other languages from overshadowing it. The corpus was divided into two main datasets: a **base** dataset excluding publisher-controlled copyright-protected books and newspapers,³ and an **extended** dataset that included all collected texts, thus including all of **base** (see Table 1).

We decided to include texts from other Scandinavian (Swedish, Danish, and Icelandic) and English sources to boost the performance of the resulting language models via cross-lingual transfer (Conneau et al., 2020b; Xue et al., 2021). To ensure that languages other than Norwegian, and primarily coming via Internet crawling, were balanced, we adapted the perplexity-based sampling strategy from De la Rosa et al. (2022) to maintain a high quality in the selected data. Instead of sampling a fixed number of documents, parameters for a Gaussian curve were calculated from 500,000-1M random documents per source, utilizing Wikipedia-based Kneser-Ney language mod-

³Except for newspapers that fall under the Language Technology Use (*Språkteknologiformål*), as they were already included in other datasets such as NCC.

Subset	Documents	Words
books	492,281	18,122,699,498
newspapers	46,764,024	9,001,803,515
books + newspapers	47,256,305	26,078,915,554
fiction books	117,319	5,287,109,366
nonfiction books	359,979	12,384,323,012
nonfiction books + newspapers	42,083,532	20,340,539,068
original books	392,887	13,352,261,605
original books + newspapers	47,156,911	22,354,065,120
translated books	96,258	4,695,814,506

Table 2: Number of documents and words (comma separated) in each subset of the publisher-controlled corpora.

els from Wenzek et al. (2019) and Conneau et al. (2020a).⁴ We also modified the perplexity calculation to account for normalized text. These parameters then guided dataset sub-sampling to target ratios per language, reducing foreign language content while maintaining quality (Appendix B).

It is also important to notice that in order to maintain the language distributions for foreign languages with respect to the amount of Norwegian texts, the total number of documents in foreign languages in the **extended** dataset is consequently higher and slightly different (due to the sampling strategies) than that of **base**; we keep the same ratios (see Appendix C).

3.2 Domain Specific Subsets

The publisher-controlled copyright-protected materials present in the **extended** dataset were further divided into groups attending to different criteria. These subsets were carefully designed to test the effect of adding them to the training sets for LLMs. We split the books into fiction vs non-fiction, and original works in Norwegian vs translations. While most books in the collection had metadata information regarding the original language in which a work was written in, genre labels were more scarce. To overcome this limitation, we built a Doc2Vec model (Le and Mikolov, 2014) that classified fiction vs nonfiction with 98% accuracy and used it to annotate books for which this information was missing.⁵ As shown in Table 2, we then built domain specific subsets to investigate 1) the effect of books vs newspapers vs books + newspapers, 2) the effect of factuality by adding only fiction words, only nonfiction works,

⁴Built with KenLM (Heafield, 2011).

⁵<https://huggingface.co/mimir-project/literary-form-classifier>

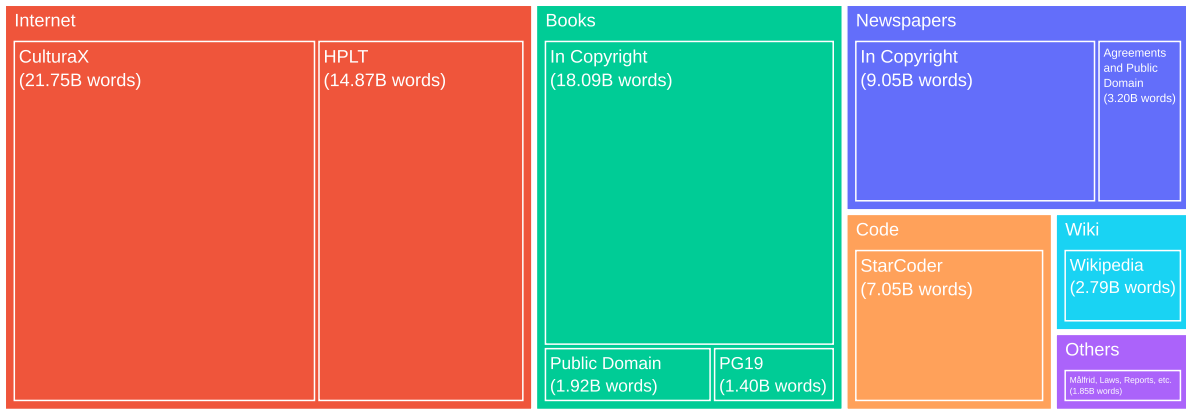


Figure 1: Treemap with the final number of words (comma separated) contributed by each source after cleaning and deduplication.

and nonfiction works + newspapers, and 3) the effect of adding content written originally in Norwegian, such as original books or original books + newspapers, vs translated books.

3.3 Instruction-tuning Datasets

To align the models more closely with human objectives and assess whether instruction tuning with limited high-quality data can enhance the performance of our pre-trained models across various tasks, we built upon prior work and collected nearly 5,000 instructions annotated by research assistants.⁶ The instructions were formatted as (*instruction*, *input*, *output*) triplets, where *instruction* refers to the directive provided by humans for the model, *input* is an optional field containing task-related information, and *output* denotes the desired response that follows the given instruction.

The instruction tuning dataset combines three key categories –Reading Comprehension, Norwegian Culture, and Words and Expressions– with diverse domains to enhance model performance. The domains include Literature, Commonsense, Geography, Language, History, Sports, Entertainment, Food, Politics, Science, Art, Music, and Culture. The variety of the instructions seeks to improve the model’s ability to understand complex texts, provide culturally relevant responses, and handle language nuances, resulting in more versatile, knowledgeable, and context-aware LLMs.

4 Model Training

The training phase involved multiple models, each based on the Mistral architecture (Jiang et al.,

2023). The training was conducted in the following stages.

1. To measure the overall impact of publisher-controlled copyrighted corpora and its effect in realistic scenarios, we conducted pre-training on the **base** and **extended** datasets, both from scratch and using the existing weights (warm) of the pre-trained model Mistral 7B v0.1.⁷ These four *core models* were trained on the same amount of data, 64,000 steps of 4 million sub-word tokens each, using identical sets of hyperparameters (see Table 7 in Appendix D). This roughly translates to 3 epochs for the **base** dataset and 2 for the **extended** dataset, which according to Muennighoff et al. (2023) is still far from saturating the available data.
2. To further isolate the effect of different ablations of the publisher-controlled copyright-protected corpora, we continuously fine tuned the model trained on **base** from scratch for an extra 10,000 steps on each of the 9 domain specific subsets.
3. The core models were also fine tuned on the instruction data for 4 iterations to evaluate their performance on downstream tasks.

Overall, we trained 17 models (7 billion parameters each) using a total of 270,000 GPU-hours. Model training specifications are shown in Table 3. The infrastructure for training included the LUMI supercomputer, Idunn cluster, and Google

⁶<https://huggingface.co/datasets/mimir-project/mimir-instruction>

⁷<https://huggingface.co/mistralai/Mistral-7B-v0.1>

Model	Initialization	GPU/hours	Accelerator
Core Models			
base	From scratch	50K	AMD MI250X
extended	From scratch	50K	AMD MI250X
base (warm)	Mistral 7B v0.1	13.8K	NVIDIA H100
extended (warm)	Mistral 7B v0.1	55.6K	AMD MI250X
Domain Tuned Models			
base + fiction books	base	7.5K	AMD MI250X
base + nonfiction books	base	7.5K	AMD MI250X
base + nonfiction books + newspapers	base	7.5K	AMD MI250X
base + newspapers	base	4.8K	Google TPUv4
base + books	base	4.8K	Google TPUv4
base + books + newspapers	base	4.8K	Google TPUv4
base + original books + newspapers	base	9.1K	AMD MI250X
base + original books	base	9.1K	AMD MI250X
base + translated books	base	9.1K	AMD MI250X
Instruction Fine Tuned Models			
base <i>instruct</i>	base	14.2	NVIDIA H100
extended <i>instruct</i>	extended	14.2	NVIDIA H100
base (warm) <i>instruct</i>	base (warm)	14.2	NVIDIA H100
extended (warm) <i>instruct</i>	extended (warm)	14.2	NVIDIA H100

Table 3: Model training specifications, where *Model* represents the model identifier and the data used for training, *Initialization* represents the base model used for training, *GPU/hours* indicates the total GPU hours required for model training, and *Accelerator* represents the type of accelerator used.

TPUs through the Tensor Research Cloud program⁸. Besides, we trained two tokenizers with the **base** and **extended** datasets separately, both with the same vocabulary size of 32,768. After an initial test of the fertility of the tokenizers,⁹ we found the difference between them was only 0.0013. Therefore, we decided to use the same tokenizer trained with the **base** dataset for all the models.

5 Evaluation Framework

In our empirical evaluation experiments, we utilize NorEval,¹⁰ a publicly available framework for evaluating Norwegian generative LLMs built on lm-evaluation-harness (Gao et al., 2024). We consider 28 tasks, which test model’s various Norwegian language understanding and generation abilities. NorEval covers both Norwegian language varieties (Bokmål and Nynorsk) and provides a set of 4–6 prompts for each downstream task. The tasks can be grouped into nine higher level **skills**:

⁸To assess the deviation introduced by differences in training infrastructures and platforms across the participating institutions, each team trained a control model with 1.5B parameters based on the Llama 2 architecture. The training setups were identical, utilizing the **base** dataset. After comparing the validation loss curves from each team, we found that the curves were almost the same, with a deviation of less than 0.05 in terms of the final convergence validation loss.

⁹Fertility expresses the fragmentation rate of a tokenizer and is $\frac{\#tokens}{\#words}$ in one corpus.

¹⁰<https://github.com/lmgoslo/noreval>

1. **Sentiment Analysis**, here defined as binary polarity classification on both the sentence- and document-level based on the existing NoReC datasets of professional reviews (Velldal et al., 2018; Øvrelied et al., 2020).
2. **Fairness & Truthfulness**. Fairness refers to the absence of bias in the predictions and outputs of a model. Evaluating fairness ensures that the model does not favor or discriminate against particular groups based on attributes like race, gender, or ethnicity. This skill was evaluated using a newly-created dataset,¹¹ which covers a wide range of bias types, including race, religion, gender, geography, occupation, age etc. Truthfulness involves the accuracy and reliability of the information produced by the model, ensuring it generates factual and verifiable content. This skill was evaluated using NorTruthfulQA (Mikhailov et al., 2025), which assesses whether a model is truthful in selecting and generating answers to questions that involve common human misconceptions.¹²
3. **Reading Comprehension**, which measures the ability of a model to understand and interpret text. It involves answering questions

¹¹<https://huggingface.co/datasets/mimir-project/mimir-bias>

¹²https://huggingface.co/datasets/lmg/nortruthfulqa_mc and https://huggingface.co/datasets/lmg/nortruthfulqa_gen

about a given passage, summarizing content, or explaining the meaning of specific phrases or sentences. This skill estimates how well the model grasps the context and details in the text. It was evaluated using the existing extractive question-answering `NorQUAD` dataset (Ivanova et al., 2023) and multiple-choice question-answering `Belebele` dataset (Bandarkar et al., 2024).

4. **World Knowledge**, which assesses the extent of factual information a language model has about the world. This includes historical events, geographical data, scientific facts, cultural knowledge, and more. The model should correctly answer questions or provide information based on real-world knowledge. This skill was evaluated using the `NorOpenBookQA` and `NRK-Quiz-QA` by Mikhailov et al. (2025).¹³
5. **Commonsense Reasoning**, which involves the ability of a model to make logical inferences based on everyday knowledge and understanding of the world. The model should reason about situations that require practical, everyday knowledge that people take for granted. This skill was evaluated using `NorCommonsenseQA` (Mikhailov et al., 2025),¹⁴ which consists of multiple-choice commonsense question answer-pairs which adapts the corresponding English `CommonsenseQA` dataset (Talmor et al., 2019) to Norwegian.
6. **Norwegian Language** evaluation focuses on the ability of a model to understand and generate text in Norwegian, specifically its grammar, structure, and sentence construction. This skill is important for assessing how well the model handles Norwegian and their specific syntactic rules. It was evaluated using the existing `NCB` (Farsethås and Tjøstheim, 2024) and `ASK-GEC` (Jentoft, 2023) datasets, and the newly-created `NorIdiom` dataset.¹⁵
7. **Summarization**, which measures the ability of a model to condense longer pieces of text

into shorter, coherent summaries that capture the main points. This skill is crucial for applications where users need a quick understanding of large volumes of information, such as news articles or research papers. It was evaluated using the `NorSumm` dataset (Touileb et al., 2025).¹⁶

8. **Translation**, which assesses how accurately a language model can convert a text from one language to another while preserving the meaning, tone, and context. It was evaluated using the existing `Tatoeba` dataset (Tiedemann, 2020). The following six language pairs are considered: `Bokmål` ↔ `Nynorsk`, `Bokmål` ↔ `English`, and `English` ↔ `Nynorsk`.
9. **Variation and Readability**, which consists of measuring the lexical diversity of a model by looking at the amount of redundancy in the text it produces and at the readability of these texts measured by average sentence length and the proportion of long words. As such, this skill evaluation did not require any specific benchmarking datasets.

We follow the standard in-context learning evaluation design for pretrained decoder-only language models (e.g., Brown et al., 2020; Touvron et al., 2023), which includes zero-shot and few-shot evaluation. In this paper, for the sake of simplicity, we selected the most common metrics per task and aggregated scores using a simple cumulative sum per higher-level skill. In order to aggregate results into an overall score, with the caveats of aggregating metrics of different nature, scores were extracted for the best available $\{0, 1, 4, 16\}$ -shot configuration for each task and the best score for each of the prompts. Metrics were normalized to exhibit the same higher-is-better behavior in a range of 0 to 100.

6 Results

The evaluation of the trained models demonstrated that incorporating publisher-controlled copyright-protected corpora provided a measurable boost in performance across a range of NLP tasks. To illustrate the overall performance differences, Figure 2 shows the total scores across all skills, averaged by task for each model. Non-aggregated scores

¹³<https://huggingface.co/datasets/litg/noropenbookqa> and https://huggingface.co/datasets/litg/nrk_quiz_qa

¹⁴<https://huggingface.co/datasets/litg/norcommonsenseqa>

¹⁵<https://huggingface.co/datasets/mimir-project/noridiom>

¹⁶<https://huggingface.co/datasets/SamiaT/NorSumm>

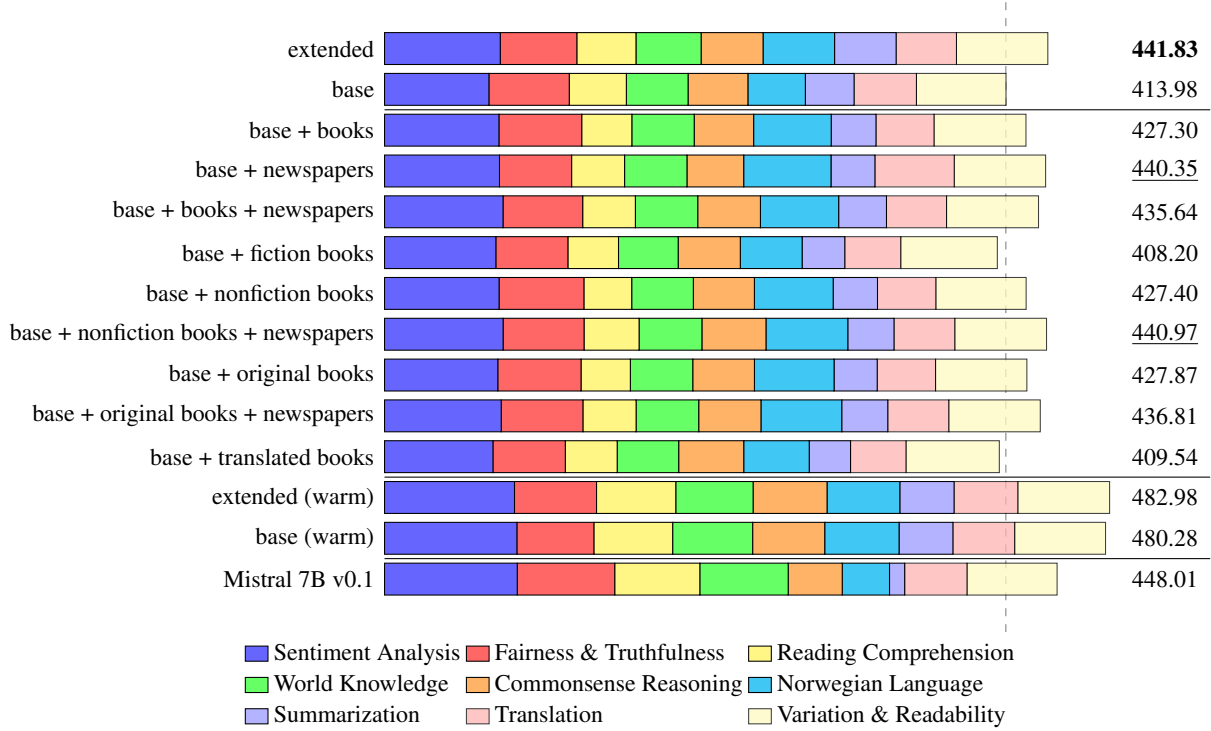


Figure 2: Total summed scores across all skills averaged by task for each model. Best scores among from-scratch models underlined, best overall from-scratch in **bold**. Dashed line at the **base** score.

for all tasks, prompts, and models are available at the Mimir repository.¹⁷

6.1 Core Models

As shown in Table 4 and Figure 2, the performance analysis of core models across various tasks reveals distinct strengths for different configurations. The **base** (warm-started) configuration consistently excels in Sentiment Analysis, World Knowledge, and Norwegian Language. In contrast, the **extended** (warm-started) configuration leads in Fairness & Truthfulness, Reading Comprehension, Commonsense Reasoning, Translation, and Variation & Readability, indicating its robust performance for language-intensive tasks. The **base** configuration generally lags behind others, scoring the lowest across multiple tasks. Meanwhile, the **extended** configuration performs well, particularly in Summarization. Furthermore, it indicates that we could leverage the existing metadata available at the National Library to tailor subsets of the publisher-controlled copyrighted corpora and build models that excel at specific tasks. However, the difference between the **base** and **extended** warm-started models is very small.

¹⁷<https://github.com/mimir-project/mimir-evaluation>

Model	SA	FT	RC	WK	RC	NL	S	T	VR
extended	3	2	3	3	2	2	1	3	2
base	4	3	4	4	3	4	3	4	3
extended (warm)	2	3	1	2	1	1	2	1	1
base (warm)	1	1	2	1	1	3	2	2	4

Table 4: Results for ranking the core models on all tasks by skill via (i) finding the best k-shot configuration for each task and (ii) aggregating metric-wise rankings. SA=Sentiment Analysis. FT=Fairness & Truthfulness. RC=Reading Comprehension. NL=Norwegian Language. WK=World Knowledge. CR=Commonsense Reasoning. S=Summarization. T=Translation. VR=Variation & Readability. Lower is better.

Further testing is required to assess whether this difference is still statistically significant.¹⁸

While warm-started models generally outperformed models trained from scratch, there was reduced sensitivity to the presence of copyrighted materials. This suggests that the pre-existing weights, which were primarily trained on English data, diminished the impact of adding high-quality Norwegian copyrighted texts (see also Section 7).

¹⁸Detailed scores available in Appendix F Table 8.

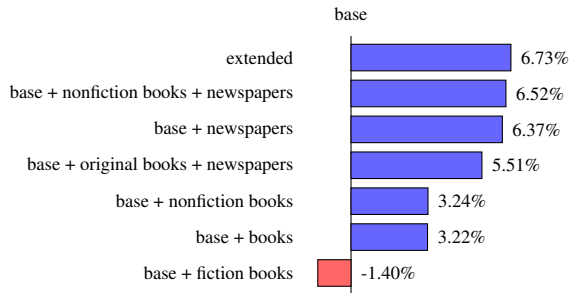


Figure 3: Average percentage gains over the performance of the base model. Negative results indicate a decrease in performance over base, positive results a gain.

6.2 Domain-tuned Models

To further explore the specific effects of different types of training data, we analyzed the gains in performance by focusing on different sub-corpora, such as newspapers, books, and mixed datasets. Figure 3 provides an overview of the average percentage gains for models trained on various data configurations compared to the **base** model. It shows that the **extended** model exhibits the highest average gain at 6.73%, indicating substantial overall improvement. The addition of nonfiction books and newspapers follows with a 6.52% gain, and the addition of only newspapers shows a 6.37% improvement. Other configurations, such as adding original books and newspapers or nonfiction books, also demonstrate positive gains of 5.51% and 3.22%, respectively. Conversely, the addition of fiction books is the only one to show a negative performance, with a decrease of 1.40%. Interestingly, when decomposed by skill, the addition of fiction books makes the model excel at generating more diverse texts (see Figure 5 in Appendix E).

6.3 Instruction-tuned Models

Lastly, as shown in Figure 4, when the core models are further fine-tuned on data to follow instructions, the gains across models are all consistent, showing that the core advantage lies in the pre-training data, while further training on instructions gives a consistent boost in performance. Instruction tuning also seems to reduce the gap between the **base** and **extended** configurations, suggesting that publisher-controlled copyrighted corpora might become less relevant as supervised fine-tuning datasets increase in size in the post-training phases of LLMs. Interestingly, adding Norwe-

gian instruction data on top of the **extended** model seems enough to improve over the performance of Mistral 7B v0.1.

7 Discussion

Our findings underline the value of copyrighted materials in improving the performance of generative language models, particularly for specialized NLP tasks in Norwegian. The inclusion of these curated publisher-controlled texts provide a substantial advantage in terms of language richness, coherence, and context-specific understanding. However, these advantages are significantly less evident in models that are warm-started using weights pre-trained on other languages, primarily English. We see two possible reasons for this:

1. The *amount* of training data matters more than its quality or licensing status. Warm-started models are effectively trained on more data than the ‘from-scratch’ models, and at some point adding even more data brings diminishing returns (with a given model size).
2. Publisher-controlled copyrighted Norwegian data is indeed beneficial for LLMs, but the original models used for warm-starting *were presumably already pre-trained on datasets that may share similarities with this data*. Due to the lack of transparency regarding the exact composition of training datasets in models like Mistral, concerns about potential data contamination remain relevant. This overlap could explain why continuous pre-training on similar content did not yield the expected benefits for the warm-started extended models (Li et al., 2024; Dong et al., 2024; Xu et al., 2024; Samuel et al., 2024).

7.1 Ethical and Legal Considerations

The use of copyrighted materials in model training raises significant ethical and legal questions. The observed improvements in model quality must be balanced against the rights of content creators, who have not consented to the use of their work. This highlights the need for guidelines and compensation mechanisms that recognize the value of copyrighted materials in LLM development.

7.2 Implications for Policy

The empirical evidence gathered in our research is crucial for informing copyright policy in the digital age. Policymakers can use these findings to

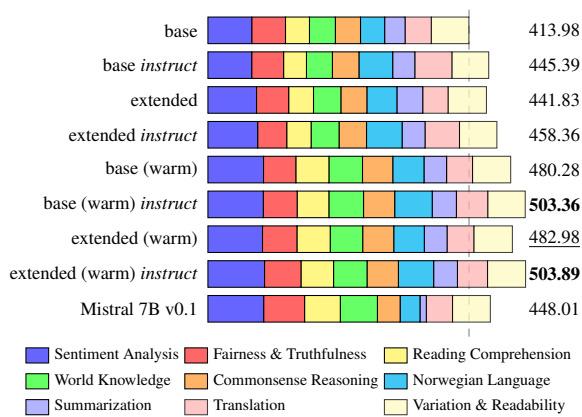


Figure 4: Total scores (sum) of all averaged scores per skill for the core models and their instruct versions, with original Mistral 7B v0.1 for reference. Dashed line at the **base** score. Best scores in **bold**, second best underlined.

establish frameworks that ensure creators are adequately compensated, balancing the needs of LLM innovation with the rights of authors and publishers. This is particularly relevant in light of ongoing lawsuits against major AI companies.

8 Conclusion

Our study represents a pioneering effort to quantify the impact of copyrighted materials on LLMs for Norwegian. Our results indicate that high-quality publisher-controlled copyrighted content significantly enhances model performance, especially for complex NLP tasks. However, these benefits bring forth ethical and legal challenges that must be addressed to ensure a sustainable and fair approach to LLM development. By providing empirical evidence, we aim to contribute to the ongoing discourse on the role of copyright in AI and inform future policies that support both innovation and the rights of content creators.

9 Future Work

Future work should focus on testing models at various scales and different pre-trained open weights to better understand how dataset composition affects performance. By experimenting with models of different sizes, we could identify any scaling thresholds where the impact of copyrighted material varies significantly. In retrospect, one notable flaw in the experimental design is the lack of fully traceable and transparent models, such as

OLMo (Groeneveld et al., 2024), which provide detailed documentation of their training data and processes. Without utilizing models with verifiable data provenance, it becomes challenging to accurately assess how specific dataset compositions, including copyrighted or genre-specific materials, influence model behavior and performance for warm-started models. Incorporating traceable models would improve the reproducibility and reliability of findings, ensuring that conclusions drawn about the impact of various text genres are well-founded.

Additionally, the observed effects of fiction on model performance highlight the need to 1) examine how different types of fiction—such as fantasy or historical fiction—impact tasks like Sentiment Analysis and Commonsense Reasoning, and 2) design new and adequate benchmarks for evaluating the contribution of fiction in Norwegian LLMs for tasks such as creative writing, plot understanding, or descriptive language use. This investigation could clarify the role of fiction in model training and help refine data curation strategies.

Lastly, exploring genre-specific influences more deeply, including essays, technical writing, and narrative nonfiction, may reveal distinct benefits or biases tied to each genre. Analyzing these nuances, even in a diachronic manner, will guide balanced genre representation in datasets and support the development of better performing models.

10 Distribution

The **base** dataset and models were intended to be freely distributed, as all materials included were granted redistribution permissions under different agreements. After we communicated the results of our investigations to the different partners, some right-holders demanded a reinterpretation of the agreements (primarily the Language Technology Use, *Språkteknologiformål*), in the light of the results and this new era of generative AI. This prevented us from sharing publicly the exact models trained within the Mimir project, but instead we built a subset of **base**, which we are calling **core**, excluding the affected newspapers (around 1B words) and trained models both from scratch and from Mistral 7B v0.1. Their performance is on par with their **base** counterparts. We are also releasing these models under a permissive license.¹⁹

¹⁹<https://huggingface.co/mimir-project/mimir-mistral-7b-core-scratch>

Acknowledgments

We extend our sincere gratitude to Hans Eide from Sigma2 for facilitating access to the LUMI supercomputer, enabling the computationally intensive tasks integral to this study. Additionally, we thank Google for providing compute resources via the Tensor Research Cloud program, which significantly supported our model training efforts.

This project would not have been possible without the trust and collaboration of the Ministry of Culture and Equality, which empowered the National Library of Norway to spearhead this endeavor, with the invaluable contributions of the Norwegian University of Science and Technology (NTNU) and the University of Oslo (UiO), whose expertise and insights were instrumental throughout the process. We are grateful for their vision and faith in the potential of this research.

We are also deeply appreciative of Olaus Bergstrøm and the entire legal team at the National Library of Norway for their guidance on the legal dimensions of this research. Their expertise was invaluable in navigating the complexities of copyright law and ensuring compliance with the unique considerations surrounding the materials used in this project.

A special thanks goes to the representatives of the Norwegian rights-holder organizations, who not only agreed to the use of their materials for this project but were steadfast in their support of the initiative. Their cooperation and encouragement have been vital in ensuring the project's success and advancing research on the intersection of copyright and AI development.

References

- Mikel Artetxe, Itziar Aldabe, Rodrigo Agerri, Olatz Perez-de Viñaspre, and Aitor Soroa. 2022. [Does corpus quality really matter for low-resource languages?](#) In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7383–7390, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabisa. 2024. [The belebele benchmark: a parallel reading comprehension dataset in 122 language variants.](#) In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775, Bangkok, Thailand. 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.](#)
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020a. [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.
- Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020b. [Emerging cross-lingual structure in pretrained language models.](#) In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034, Online. Association for Computational Linguistics.
- Javier De la Rosa, Eduardo González Ponferrada, Paulo Villegas, Pablo González de Prado Salas, Manu Romero, and María Grandury. 2022. [BERTIN: Efficient pre-training of a Spanish language model using perplexity sampling.](#) *Procesamiento de Lenguaje Natural*, 68:13–23.
- Yihong Dong, Xue Jiang, Huanyu Liu, Zhi Jin, Bin Gu, Mengfei Yang, and Ge Li. 2024. [Generalization or memorization: Data contamination and trustworthy evaluation for large language models.](#) In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*.
- Hans Christian Farsethås and Joakim Tjøstheim. 2024. Norwegian comma benchmark. <https://huggingface.co/datasets/hcfa/ncb>.
- Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac’h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailley Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024. [A framework for few-shot language model evaluation.](#)

and <https://huggingface.co/mimir-project/mimir-mistral-7b-core>

- Daniel Gervais, Noam Shemtov, Haralambos Marmaris, and Catherine Rowland. 2024. [The heart of the matter: Copyright, AI training, and LLMs](#).
- Ona de Gibert, Graeme Nail, Nikolay Arefyev, Marta Bañón, Jelmer van der Linde, Shaoxiong Ji, Jaume Zaragoza-Bernabeu, Mikko Aulamo, Gema Ramírez-Sánchez, Andrey Kutuzov, Sampo Pyysalo, Stephan Oepen, and Jörg Tiedemann. 2024. [A new massive multilingual dataset for high-performance language technologies](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 1116–1128, Torino, Italia. ELRA and ICCL.
- Dirk Groeneveld, Iz Beltagy, Evan Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muenighoff, Aakanksha Naik, Crystal Nam, Matthew Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, William Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah Smith, and Hannaneh Hajishirzi. 2024. [OLMo: Accelerating the science of language models](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15789–15809, Bangkok, Thailand. Association for Computational Linguistics.
- Kenneth Heafield. 2011. KenLM: Faster and smaller language model queries. In *Proceedings of the sixth workshop on statistical machine translation*, pages 187–197.
- Sardana Ivanova, Fredrik Andreassen, Matias Jentoft, Sondre Wold, and Lilja Øvrelid. 2023. [NorQuAD: Norwegian question answering dataset](#). In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 159–168, Tórshavn, Faroe Islands. University of Tartu Library.
- Matias Jentoft. 2023. [Grammatical error correction with byte-level language models](#). Master’s thesis, University of Oslo.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Léo Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. [Mistral 7B](#).
- Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.
- Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, Iroko Orife, Kelechi Ogueji, Andre Niyongabo Rubungo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. 2022. [Quality at a glance: An audit of web-crawled multilingual datasets](#). *Transactions of the Association for Computational Linguistics*, 10:50–72.
- Per E Kummervold, Javier de la Rosa, Freddy Wetjen, and Svein Arne Brygfeld. 2021. [Operationalizing a national digital library: The case for a Norwegian transformer model](#). In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 20–29, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- Per E Kummervold, Freddy Wetjen, and Javier de la Rosa. 2022. [The Norwegian Colossal Corpus: A text corpus for training large Norwegian language models](#). In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3852–3860, Marseille, France. European Language Resources Association.
- Andrey Kutuzov, Jeremy Barnes, Erik Velldal, Lilja Øvrelid, and Stephan Oepen. 2021. [Large-scale contextualised language modelling for Norwegian](#). In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 30–40, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32, ICML’14*, page II–1188–II–1196. JMLR.org.
- Yucheng Li, Frank Guerin, and Chenghua Lin. 2024. [An open source data contamination report for large language models](#). In *Proceedings of the 2nd Workshop on Mathematical and Empirical Understanding of Foundation Models at ICLR 2024*.
- Peng Liu, Lemei Zhang, Terje Farup, Even W. Lauvra, Jon Espen Ingvaldsen, Simen Eide, Jon Atle Gulla, and Zhirong Yang. 2024. [NLEBench+NorGLM: A comprehensive empirical analysis and benchmark](#)

- dataset for generative language models in Norwegian. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5543–5560, Miami, Florida, USA. Association for Computational Linguistics.
- Anton Lozhkov, Raymond Li, Loubna Ben Allal, Federico Cassano, Joel Lamy-Poirier, Nouamane Tazi, Ao Tang, Dmytro Pykhtar, Jiawei Liu, Yuxiang Wei, Tianyang Liu, Max Tian, Denis Kocetkov, Arthur Zucker, Younes Belkada, Zijian Wang, Qian Liu, Dmitry Abulkhanov, Indraneil Paul, Zhuang Li, Wen-Ding Li, Megan Risdal, Jia Li, Jian Zhu, Terry Yue Zhuo, Evgenii Zheltonozhskii, Nii Osa Osae Dade, Wenhao Yu, Lucas Krauß, Naman Jain, Yixuan Su, Xuanli He, Manan Dey, Edoardo Abati, Yekun Chai, Niklas Muennighoff, Xiangru Tang, Muhtasham Oblokulov, Christopher Akiki, Marc Marone, Chenghao Mou, Mayank Mishra, Alex Gu, Binyuan Hui, Tri Dao, Armel Zebaze, Olivier Dehaene, Nicolas Patry, Canwen Xu, Julian McAuley, Han Hu, Torsten Scholak, Sebastian Paquet, Jennifer Robinson, Carolyn Jane Anderson, Nicolas Chapados, Mostofa Patwary, Nima Tajbakhsh, Yacine Jernite, Carlos Muñoz Ferrandis, Lingming Zhang, Sean Hughes, Thomas Wolf, Arjun Guha, Leandro von Werra, and Harm de Vries. 2024. *StarCoder 2 and The Stack v2: The next generation*.
- Kevin Madigan. 2024. *Mid-year review: AI lawsuit developments in 2024*. Accessed: 2024-10-07.
- Vladislav Mikhailov, Petter Mæhlum, Victoria Ovedie Chruickshank Langø, Erik Velldal, and Lilja Øvrelid. 2025. A collection of question answering datasets for Norwegian. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, Tallinn, Estonia.
- Niklas Muennighoff, Alexander M Rush, Boaz Barak, Teven Le Scao, Aleksandra Piktus, Nouamane Tazi, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. 2023. Scaling data-constrained language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 50358–50376.
- Nasjonalbiblioteket. 2024. *Mímir-prosjektet: Evaluering av virkningen av opphavsrettsbeskyttet materiale på generative store språkmodeller for norske språk*. Technical Report. Accessed: 2024-10-26.
- Nasjonalbiblioteket. 2024. *Årsrapportar*. Accessed: 2024-10-26.
- Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2024. *CulturaX: A cleaned, enormous, and multilingual dataset for large language models in 167 languages*. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4226–4237, Torino, Italia. ELRA and ICCL.
- Lilja Øvrelid, Petter Mæhlum, Jeremy Barnes, and Erik Velldal. 2020. *A fine-grained sentiment dataset for Norwegian*. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5025–5033, Marseille, France. European Language Resources Association.
- Joe Panettieri. 2024. *Generative AI lawsuits timeline: Legal cases vs. OpenAI, Microsoft, Anthropic, Nvidia, Intel and more*. Accessed: 2024-10-07.
- Guilherme Penedo, Hynek Kydlíček, Loubna Ben Allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. *The FineWeb datasets: Decanting the web for the finest text data at scale*. *arXiv preprint arXiv:2406.17557*.
- David Samuel, Andrey Kutuzov, Samia Touileb, Erik Velldal, Lilja Øvrelid, Egil Rønningstad, Elina Sigdel, and Anna Palatkina. 2023. *NorBench – a benchmark for Norwegian language models*. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 618–633, Tórshavn, Faroe Islands. University of Tartu Library.
- David Samuel, Vladislav Mikhailov, Erik Velldal, Lilja Øvrelid, Lucas Georges Gabriel Charpentier, and Andrey Kutuzov. 2025. Small Languages, Big Models: A Study of Continual Training on Languages of Norway. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, Tallinn, Estonia.
- Vinay Samuel, Yue Zhou, and Henry Peng Zou. 2024. *Towards data contamination detection for modern large language models: Limitations, inconsistencies, and oracle challenges*. In *Proceedings of the 29th International Conference on Computational Linguistics (COLING 2024)*.
- Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. *CommonsenseQA: A question answering challenge targeting commonsense knowledge*. 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 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.
- Jörg Tiedemann. 2020. *The tatoeba translation challenge – realistic data sets for low resource and multilingual MT*. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1174–1182, Online. Association for Computational Linguistics.
- Samia Touileb, Vladislav Mikhailov, Marie Ingeborg Kroka, Øvrelid Lilja, and Erik Velldal. 2025. Benchmarking abstractive summarisation: A dataset

of human-authored summaries of Norwegian news articles. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, Tallin, Estonia.

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](#).

Erik Velldal, Lilja Øvrelid, Eivind Alexander Bergem, Cathrine Stadsnes, Samia Touileb, and Fredrik Jørgensen. 2018. [NoReC: The Norwegian review corpus](#). In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Theresa M. Weisenberger, Diana C. Milton, Harrison A. Enright, and Jiwon Kim. 2024. [Case tracker: Artificial intelligence, copyrights, and class actions](#). Accessed: 2024-10-07.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzm'an, Armand Joulin, and Edouard Grave. 2019. [CCNet: Extracting high quality monolingual datasets from web crawl data](#). In *International Conference on Language Resources and Evaluation*.

Cheng Xu, Shuhao Guan, Derek Greene, and M-Tahar Kechadi. 2024. [Benchmark data contamination of large language models: A survey](#). In *Proceedings of the 1st Workshop on Data Contamination (CONDA) at ACL 2024*.

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*, pages 483–498, Online. Association for Computational Linguistics.

Appendices

A Legal Cases

- **Bartz v. Anthropic PBC**, No. 3:24-cv-05417 (N.D. Cal. Aug. 19, 2024)
- **The Ctr. for Investigative Reporting v. OpenAI, Inc.**, No. 1:24-cv-04872 (S.D.N.Y. Jun. 27, 2024)
- **UMG Recordings, Inc. v. Uncharted Labs, LLC**, No. 1:24-cv-04777 (S.D.N.Y. Jun. 24, 2024)
- **UMG Recordings, Inc. v. Suo, Inc.**, No. 1:24-cv-11611 (D. Mass. Jun. 24, 2024)
- **J. L. v. Alphabet, Inc.**, No. 3:23-cv-03440, 2024 WL 3282528 (N.D. Cal. June 6, 2024)
- **In re OpenAI ChatGPT Litigation**, No. 3:23-cv-03223, 2024 WL 2044625 (N.D. Cal. May 7, 2024)
- **Makkai v. Databricks, Inc.**, No. 4:24-cv-02653 (N.D. Cal. May 2, 2024)
- **Dubus v. NVIDIA Corp.**, No. 3:24-cv-02655 (N.D. Cal. May 2, 2024)
- **Daily News LP v. Microsoft Corp.**, No. 1:24-cv-03285 (S.D.N.Y. Apr. 30, 2024)
- **Zhang v. Google LLC**, No. 3:24-cv-02531 (N.D. Cal. Apr. 26, 2024)
- **Nazemian v. NVIDIA Corp.**, No. 4:24-cv-01454 (N.D. Cal. Mar. 8, 2024)
- **The Intercept Media, Inc. v. OpenAI, Inc.**, No. 1:24-cv-01515 (S.D.N.Y. Feb. 28, 2024)
- **Raw Story Media, Inc. v. OpenAI Inc.**, No. 1:24-cv-01514 (S.D.N.Y. Feb. 28, 2024)
- **Tremblay v. OpenAI, Inc.**, No. 3:23-cv-03233, 2024 WL 557720 (N.D. Cal. Feb. 12, 2024)
- **Universal Music Group v. Anthropic** (February 2024)
- **The New York Times v. OpenAI & Microsoft** (December 2023)
- **Kadrey v. Meta Platforms, Inc.**, No. 3:23-cv-03417, 2023 WL 10673221 (N.D. Cal. Dec. 1, 2023)
- **Alter v. OpenAI Inc.**, No. 1:23-cv-10211 (S.D.N.Y. Nov. 21, 2023)
- **Andersen v. Stability AI Ltd.**, No. 3:23-cv-00201, 2023 WL 7132064 (N.D. Cal. Oct. 30, 2023)
- **Concord Music Group, Inc. v. Anthropic PBC**, No. 3:23-cv-01092 (M.D. Tenn. Oct. 18, 2023)
- **Huckabee v. Meta Platforms, Inc.**, No. 3:23-cv-06663 (N.D. Cal. Oct. 17, 2023)
- **Authors Guild v. OpenAI Inc.**, No. 1:23-cv-08292 (S.D.N.Y. Sept. 18, 2023)
- **Silverman v. OpenAI Inc.**, No. 3:23-cv-03416 (N.D. Cal. July 7, 2023).
- **Thaler v. Perlmutter**, 687 F. Supp. 3d 140 (D.D.C. 2023)
- **Doe 1 v. Github, Inc.**, No. 4:22-cv-06823, 2023 WL 3449131 (N.D. Cal. May 11, 2023)
- **Getty Images (US), Inc. v. Stability AI, Inc.**, No. 1:23-cv-00135 (D. Del. Feb. 3, 2023)

B Sampling

We built three custom perplexity models for specific Norwegian domains that proved too divergent from Wikipedia: books, newspapers, and government documents. These perplexity models were used to score each document in the datasets. Based on their perplexity scores, the documents were further divided into three segments corresponding to their quartile distribution. Documents with scores below the first quartile were classified as “good”, those between Q_1 and Q_3 as “medium”, and those above Q_3 were considered “bad”. The documents in each segment were randomized. While the intention was to train all models on progressively better data, starting from “bad” segment, then “medium” and finally the “good” segment, we never got around to test whether this approach would result in better performing models.

Moreover, from the clean and deduplicated sources, we sub-sampled each non-Norwegian language at an specific sampling ratio until achieving the proportion of documents shown in Figure 5. Pseudo-code for the algorithm used to subsample is shown in Algorithm 1.²⁰ We also discovered that a good amount of documents were misclassified by the fastText language identifier (Joulin et al., 2016).

Language	Sampling ratio	Final ratio
Bokmål	100.00%	35.74%
Danish	43.00%	8.01%
English	81.00%	4.53%
Icelandic	100.00%	1.31%
Nynorsk	100.00%	2.02%
Swedish	15.40%	4.46%
Code	62.00%	4.53%

Table 5: Percentage of documents kept from the clean and deduplicated sources and the final proportion of documents in each language present in the final dataset. Code was considered its own language when sampling.

C Sources

Source	Raw	Clean	extended	base
Books	3.7B	2.5B	1.9B	1.9B
CulturaX	52.7B	52.1B	21.8B	16.9B
Digimanus	9.6M	4.6M	3.4M	3.3M
Evaluerings- rapport	76.7M	68.6M	61.2M	61.5M
HPTL v1.2	35.5B	34.1B	14.9B	11.3B
LovData	57.1M	57.1M	53.7M	54.8M
Målfrid	7.5B	1.9B	1.7B	1.7B
Newspapers	4.6B	3.6B	3.2B	3.3B
Parlamint	170.3M	84.4M	83.4M	83.3M
PG19	2.0B	1.9B	1.4B	428.6M
StarCoder	19.7B	9.8B	7.1B	3.4B
Wikipedia	4.0B	3.9B	2.8B	996.2M
Books (restricted)	21.7B	20.0B	18.1B	0
Newspapers (restricted)	14.3B	9.8B	9.1B	0
Total	166.1B	139.8B	82.1B	40.1B

Table 6: Number of words (comma separated) per source at the start of the data pipeline (raw count), after cleaning, and in the **extended** and **base** datasets.

²⁰<https://huggingface.co/mimir-project/mimir-perplexity>

Algorithm 1 Sub-sampling Dataset Based on Perplexity Distribution

```
1: Input: Dataset  $D$  with perplexity distribution, target sampling ratio  $R$ 
2: Output: Sub-sampled dataset  $D'$ 
3: procedure SUBSAMPLE( $D, R$ )
4:   Compute the quartile values  $q_1$  and  $q_3$  from the perplexity distribution of  $D$ 
5:   Define an initial Gaussian PDF with mean  $\mu = (q_1 + q_3)/2$  and standard deviation  $\sigma$  such that
    $q_1$  and  $q_3$  align with the corresponding positions in the Gaussian curve
6:   Compute the histogram  $H$  of perplexity values from  $D$ 
7:   Combine  $H$  with the Gaussian weights to estimate the initial sampling ratio  $R_0$ 
8:   Compute the normalization factor  $N$  such that  $R_0 \times N = R$ 
9:   while Error in central quartile probabilities exceeds tolerance do
10:    Adjust the parameters  $\mu$  and  $\sigma$  of the Gaussian curve to minimize the error in the desired
    probabilities within the central quartiles  $[q_1, q_3]$ 
11:    Update the normalization factor  $N$  to match the target ratio  $R$ 
12:  end while
13:  for each sample  $s$  in  $D$  do
14:    Compute the perplexity  $p_s$  of sample  $s$ 
15:    Estimate the probability  $P(s)$  of retaining sample  $s$  based on the normalized Gaussian PDF
16:    if  $P(s) \geq$  random threshold then
17:      Retain  $s$  in the sub-sampled dataset  $D'$ 
18:    end if
19:  end for
20: end procedure
21: return  $D'$ 
```

D Hyperparameters

Hyperparameter	Core Models	Domain-Tuned Models	Instruction-tuned Models
Model size	7B	7B	7B
Hidden layers	32	32	32
Attention heads	32	32	32
Hidden size	4096	4096	4096
Intermediate size	14336	14336	14336
Max position embeddings	2048	2048	2048
Key-value heads	8	8	8
Sliding window	4096	4096	4096
Precision	bfloat16	bfloat16	bfloat16
Optimizer	AdamW	AdamW	AdamW
Optimizer parameters	$\beta_1 = 0.9, \beta_2 = 0.95, \epsilon = 10^{-8}$	$\beta_1 = 0.9, \beta_2 = 0.95, \epsilon = 10^{-8}$	$\beta_1 = 0.9, \beta_2 = 0.95, \epsilon = 10^{-8}$
Global batch size	4M (2048 \times 2048) tokens	4M (2048 \times 2048) tokens	512 seqs
Initial/final learning rate	$3.0 \times 10^{-4} / 3.0 \times 10^{-5}$	$3.0 \times 10^{-5} / 3.0 \times 10^{-6}$	$3.0 \times 10^{-6} / 3.0 \times 10^{-7}$
Vocabulary size	32768	32768	32768
Training steps	64k	10k	4 epochs
Dropout	0	0	0
Warm-up steps	2000	200	20
Weight decay	0.1	0.1	0.1
Checkpoints	Every 1000 steps	Every 1000 steps	Every 1 epoch
Shuffle	Shuffle after each epoch	Shuffle after each epoch	Shuffle after each epoch

Table 7: Hyperparameters for the Mimir model set.

E Percentage Gains

Figure 5 illustrates the percentage gains of each domain-tuned model with respect to the performance of the **base** model, per higher level skill. Training on different materials shows distinct trade-offs: newspaper data excels at Translation (27.20% gain) and Norwegian Language (51.92%), while fiction books improve Variation & Readability (7.83%). Combining books and newspapers often yields balanced improvements, though most configurations struggle with Reading Comprehension and Translation. The **extended** configuration, which supplements books and newspapers with Internet data, shows strong all-around performance, particularly in Summarization (26.37%) and World Knowledge (5.60%).

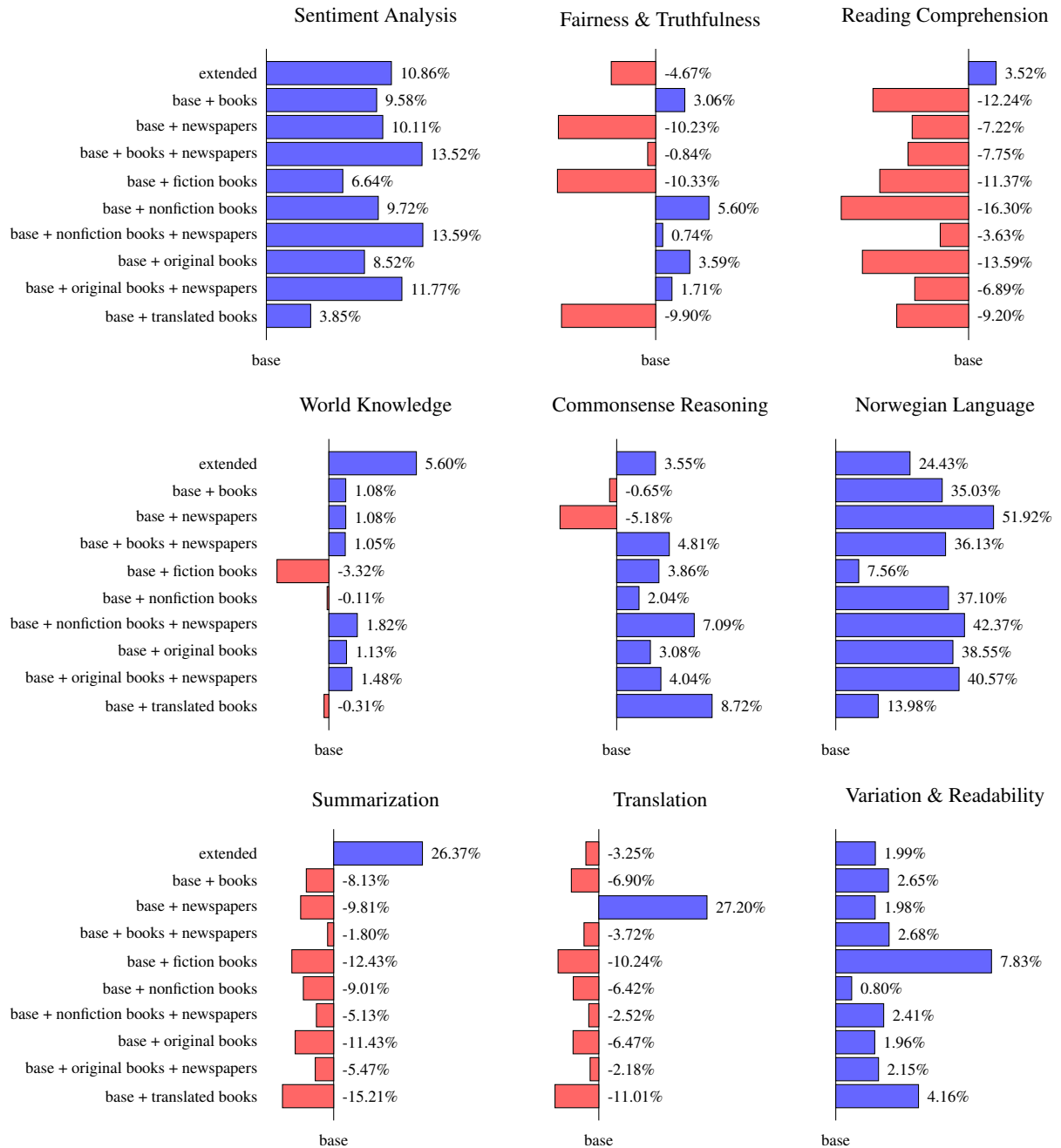


Figure 5: Percentage gains over the performance of the **base** model per skill.

F Evaluation Scores

Model	SA	FT	RC	WK	CR	NL	S	T	VR	Score
Core Models										
base	69.54	53.51	38.04	41.10	39.85	38.28	32.45	41.55	59.66	413.98
extended	77.09	51.01	39.38	43.40	41.26	47.64	41.00	40.20	60.85	441.83
base (warm)	<u>88.17</u>	51.28	52.48	<u>53.27</u>	48.02	<u>49.51</u>	35.88	41.14	60.52	480.28
extended (warm)	86.57	<u>54.64</u>	<u>52.79</u>	51.51	<u>49.25</u>	48.48	36.14	<u>42.48</u>	<u>61.12</u>	<u>482.98</u>
Domain Tuned Models										
base + books	76.20	55.15	33.38	41.54	39.59	51.69	29.81	38.68	61.24	427.30
base + newspapers	76.57	48.04	35.29	41.55	37.79	<u>58.16</u>	29.26	<u>52.85</u>	60.85	440.35
base + books + newspapers	78.94	53.06	35.09	41.53	41.77	52.11	<u>31.86</u>	40.00	61.27	435.64
base + fiction books	74.16	47.99	33.71	39.74	41.39	41.18	28.41	37.29	64.33	408.20
base + nonfiction books	76.30	<u>56.51</u>	31.84	41.06	40.66	52.48	29.52	38.88	60.14	427.40
base + nonfiction books + newspapers	<u>78.99</u>	53.91	<u>36.66</u>	<u>41.85</u>	42.68	54.50	30.78	40.50	61.10	<u>440.97</u>
base + original books	75.46	55.43	32.87	41.56	41.08	53.04	28.74	38.86	60.83	427.87
base + original books + newspapers	77.72	54.43	35.42	41.71	41.46	53.81	30.67	40.64	60.95	436.81
base + translated books	72.22	48.21	34.54	40.97	<u>43.33</u>	43.63	27.51	36.97	62.15	409.54
Instruction Fine Tuned Models										
base (warm) <i>instruct</i>	87.83	53.70	50.33	<u>54.98</u>	49.42	59.53	<u>38.36</u>	49.75	59.46	503.36
extended (warm) <i>instruct</i>	89.81	<u>57.80</u>	<u>51.69</u>	53.09	49.76	55.91	37.75	47.72	<u>60.35</u>	503.89
base <i>instruct</i>	69.45	50.59	36.27	41.18	42.06	53.53	35.14	58.83	58.35	445.39
extended <i>instruct</i>	78.90	46.10	38.68	44.32	43.57	56.46	36.40	54.64	59.29	458.36
Mistral 7B v0.1	88.41	64.93	56.68	58.86	36.01	31.49	10.09	41.55	59.99	448.01

Table 8: Detailed scores across all skills for each model configuration. Abbreviations: SA = Sentiment Analysis, FT = Fairness & Truthfulness, RC = Reading Comprehension, WK = World Knowledge, CR = Commonsense Reasoning, NL = Norwegian Language, S = Summarization, T = Translation, VR = Variation & Readability. Best overall scores per skill in **bold**. Best score per skill and model group underlined. Mistral 7B v0.1 also added for reference.

Encoder vs Decoder: Comparative Analysis of Encoder and Decoder Language Models on Multilingual NLU Tasks

Dan Saattrup Nielsen
The Alexandra Institute
dan.nielsen@alexandra.dk

Kenneth Enevoldsen
University of Aarhus
kenneth.enevoldsen@cas.au.dk

Peter Schneider-Kamp
University of Southern Denmark
petersk@imada.sdu.dk

Abstract

This paper explores the performance of encoder and decoder language models on multilingual Natural Language Understanding (NLU) tasks, with a broad focus on Germanic languages. Building upon the ScandEval benchmark, initially restricted to evaluating encoder models, we extend the evaluation framework to include decoder models. We introduce a method for evaluating decoder models on NLU tasks and apply it to the languages Danish, Swedish, Norwegian, Icelandic, Faroese, German, Dutch, and English. Through a series of experiments and analyses, we also address research questions regarding the comparative performance of encoder and decoder models, the impact of NLU task types, and the variation across language resources. Our findings reveal that encoder models can achieve significantly better NLU performance than decoder models despite having orders of magnitude fewer parameters. Additionally, we investigate the correlation between decoders and task performance via a UMAP analysis, shedding light on the unique capabilities of decoder and encoder models. This study contributes to a deeper understanding of language model paradigms in NLU tasks and provides valuable insights for model selection and evaluation in multilingual settings.

1 Introduction

Language models have attained remarkable Natural Language Understanding (NLU) performance, both with encoder-based architectures like BERT (Devlin et al., 2018) and decoder-based architectures like GPT-3 (Brown, Tom and Mann, Benjamin and Ryder, Nick and Subbiah, Melanie and

Kaplan, Jared D and Dhariwal, Prafulla and Nee-lakantan, Arvind and Shyam, Pranav and Sastry, Girish and Askell, Amanda and others, 2020). The encoder models have excelled in capturing contextual information for downstream tasks through masked language modeling objectives, while decoder models have shown strong generative capabilities by autoregressively predicting subsequent tokens based on preceding context.

Since the “ChatGPT boom” in 2023, the research community has been increasingly focused on decoder models (Zhao et al., 2023) for both Natural Language Generation (NLG) and NLU tasks. However, few studies have systematically compared the performance of encoder and decoder models across a diverse range of NLU tasks, and the studies that exist have primarily focused on English. This leaves a gap in our understanding of how the two language model paradigms perform in multilingual settings across different languages and tasks.

Nielsen (2023) introduced the ScandEval benchmark and evaluated encoder language models on four different natural language understanding tasks in Danish, Swedish, Norwegian (Bokmål and Nynorsk), Icelandic and Faroese. In this paper, we bridge this gap by extending the ScandEval benchmark to encompass the evaluation of decoder models on multilingual NLU tasks, as well as expanding the language resources to include German, Dutch and English.

Our **main research question** is

Which language model paradigm is better suited for NLU?

We will answer this question with the languages Danish, Swedish, Norwegian, Icelandic, Faroese, German, Dutch and English as a case study. To concretise our main question, we will study the following research questions in this paper:

(Q1) Can state-of-the-art finetuned encoder models

achieve significantly better NLU performance than state-of-the-art decoder models?

- (Q2) Does the answer to (Q1) depend on the type of NLU task?
- (Q3) Does the answer to (Q1) vary along the language resource spectrum, from low- to high-resource?

Our main contributions of this paper are the following:

1. We extend the ScandEval benchmarking framework with few-shot evaluation of decoder models and release this extension open-source.
2. We extend the languages supported by the ScandEval benchmarking framework by German, Dutch and English. Together with Danish, Swedish, Norwegian, Icelandic and Faroese, ScandEval now provides coverage of all Germanic languages except Afrikaans and the Frisian languages.
3. We evaluate an extensive suite of both encoder and decoder models on NLU tasks in all of the supported languages and publish these on public leaderboards.
4. We give a positive answer to (Q1), showing that encoder models achieve significantly better NLU performance than decoder models in several languages. This depends on the language in question however, giving a partially positive answer to (Q3).
5. We also show that the decoder models are heavily biased towards the question answering task (even models that are not instruction tuned), and a UMAP analysis shows that the performance distribution of decoder models follow a different “path” than encoder models, from the worst to best performing models. This gives a positive answer to (Q2).

2 Related Work

2.1 Comparing Encoder and Decoder Models

There has been a number of studies in recent years comparing encoder models to decoder models. Zhong et al. (2023) compared GPT-3.5-turbo (January 2023 version) to (finetuned versions of) the base and large versions of BERT (Devlin et al.,

2018) and RoBERTa (Liu et al., 2019) on the English GLUE benchmark (Wang et al., 2018). They find that GPT-3.5-turbo is on average on par with the base-sized encoder models, but falls short of the large-sized ones. They also note that despite being on par with the base-sized models, there is a big discrepancy between the models on individual tasks, with GPT-3.5-turbo for instance being better on the inference tasks while being worse on the paraphrase tasks. We note however that they only evaluate the decoder model in a zero-shot setting, and furthermore they only evaluate the models on 25 samples for each class in the development split, leading to a potential lack of robustness in their evaluation.

Wang et al. (2023) compares GPT-3.5-turbo (January 2023 version) to a finetuned version of the base-sized BERT model on 18 English benchmark datasets related to sentiment analysis. Like Zhong et al. (2023), they find that the zero-shot performance of GPT-3.5-turbo is on par with the base-sized BERT model, and that the few-shot performance of GPT-3.5-turbo (with 27 few-shot examples) is slightly better than BERT, on average. Their test sets contained, on average, 538 samples, which is a significant improvement over Zhong et al. (2023). However, the narrow focus on the evaluation tasks as well as only benchmarking a single encoder and decoder model makes it hard to generalise the results to other tasks and models.

Kocoń et al. (2023) built a benchmark suite of 25 tasks, where 21 of these tasks are classification tasks (binary, multi-class and multi-label), 3 being question answering tasks and the last one being a token classification task. Two of the classification tasks are in Polish and the rest in English. They compare the zero-shot and few-shot performance of GPT-3.5-turbo (January 2023 version) to the state-of-the-art encoder performance on each task. GPT-3.5-turbo is generally found to be worse than state-of-the-art encoder models. They also evaluate GPT-4 on five of the tasks (inference, question-answering and emotion datasets), and only find GPT-4 to be marginally better than GPT-3.5-turbo, still far off the encoder models.

Qiu and Jin (2024) compare GPT-3.5-turbo (January 2023 version) to a finetuned version of the base-sized BERT model on three manually curated English multi-class classification datasets with 19, 12 and 7 test samples, respectively, where they find that the BERT model performs marginally better

than GPT-3.5-turbo in a few-shot setting (and that the zero-shot performance is significantly worse). The tiny test sets make it hard to generalise the results, however.

2.2 Benchmarks of Generative Language Models

In recent times, several benchmarks of generative language models have been introduced. The major ones are EleutherAI’s Evaluation Harness (Gao et al., 2023), Hugging Face’s Open LLM Leaderboard (hug) which uses the Evaluation Harness as evaluation engine, and Stanford University’s HELM (Bommasani et al., 2023). These are firstly all English-only benchmarks, making it hard to generalise the results to other languages, and they only include point estimates of the dataset performance, and thus do not necessarily provide a robust assessment of the models. Further, these benchmarks are exclusively for decoder models, and thus does not provide a way to compare encoders with decoders.

There has been several language-specific benchmarks introduced as well. NorBench (Samuel et al., 2023) is a collection of Norwegian evaluation datasets moreso than a dedicated evaluation framework. Further, several datasets in this collection (NorQuAD, NoReC and NorNE) are already part of ScandEval. SuperLim (Berdičevskis et al., 2023) falls into the same category for Swedish. DUMB (de Vries et al., 2023) is a Dutch benchmarking framework, which is only focused on encoder models. Danoliterate (Holm, 2024) is a Danish benchmarking framework which is solely focused on evaluating decoder models, and whose datasets largely overlap with the Danish datasets in ScandEval, albeit with a different evaluation methodology. Aside from language modelling performance, the Danoliterate benchmark also measures calibration, efficiency, toxicity and fairness. While the development of language-specific benchmarks is important, it leads to too little overview of trends across benchmarks and languages and incentivises model development focused on monolingual models ignoring a potential broader appeal. ScandEval provides a unified and robust approach for comparison across model categories and Germanic languages.

Benchmarking is not the only way to evaluate language models. A new “arena approach” has been popularised by the LMSYS Arena (Chiang et al.), where users can submit a prompt and get two responses from two anonymised models at random,

and have to evaluate the responses. The Arena is predominantly used for English, but also currently supports six other languages. This approach is a promising way to evaluate language models, but we fear that it is not as suitable for low-resource languages due to the need of many volunteers to evaluate the responses.

Lastly, the Scandinavian Embedding Benchmark (Enevoldsen et al., 2024b) complements ScandEval and focuses on evaluating embedding models on a wide range of tasks in the Scandinavian languages.

3 Datasets

In this section we present the datasets that we are evaluating the models on, all of which are now included in the ScandEval framework. We should note that these datasets either (a) already existed prior to this publication or (b) are small extensions of existing datasets. An overview of all the datasets can be found in Table 1.

3.1 Named Entity Recognition

For Norwegian, Swedish and Icelandic we use the NorNE (Jørgensen et al., 2020), SUC 3.0 (Gustafson-Capková and Hartmann, 2006), MIM-GOLD-NER (Ingólfssdóttir et al., 2020) datasets, which were already included in the ScandEval framework. For Faroese we replace the previous WikiANN-fo dataset (Rahimi et al., 2019) with the new human annotated FoNE dataset (Snæbjarnarson et al., 2023). We also replace the previous DaNE dataset (Hvingelby et al., 2020) with the new DANSK dataset (Enevoldsen et al., 2024a) covering a wider variety of domains. For German, Dutch and English we add the established NER datasets GermEval (Benikova et al., 2014), the Dutch part of CoNLL-2002 (Sang, 2002), and the English CoNLL-2003 (Sang and De Meulder, 2003).

3.2 Sentiment Classification

We re-use the sentiment classification datasets AngryTweets (Pauli et al., 2021), NoReC (Velldal et al., 2018) and SweReC (Svensson, 2017), for Danish, Norwegian and Swedish, respectively. For German, Dutch and English we add the existing datasets SB10k (Cieliebak et al., 2017), Dutch Social (Gupta, 2022) and SST5 (Socher et al., 2013). We convert SST5 to the standardised trinary (negative, neutral, positive) format by converting the

Dataset	Language	#Train	#Val	#Test	#Shots
NER					
DANSK (Enevoldsen et al., 2024a)	Danish	1,024	256	1,024	8
SUC 3.0 (Gustafson-Capková and Hartmann, 2006)	Swedish	1,024	256	2,048	8
NorNE-nb (Jørgensen et al., 2020)	Norwegian Bokmål	1,024	256	2,048	8
NorNE-nn (Jørgensen et al., 2020)	Norwegian Nynorsk	1,024	256	2,048	8
MIM-GOLD-NER (Ingólfssdóttir et al., 2020)	Icelandic	1,024	256	2,048	8
FoNE (Snæbjarnarson et al., 2023)	Faroese	1,024	256	2,048	8
GermEval (Benikova et al., 2014)	German	1,024	256	1,024	8
CoNLL-nl (Sang, 2002)	Dutch	1,024	256	1,024	8
CoNLL-en (Sang and De Meulder, 2003)	English	1,024	256	2,048	8
Sentiment Classification					
Angry Tweets (Pauli et al., 2021)	Danish	1,024	256	2,048	12
SweReC (Svensson, 2017)	Swedish	1,024	256	2,048	12
NoReC (Velldal et al., 2018)	Norwegian	1,024	256	2,048	12
SB10k (Cieliebak et al., 2017)	German	1,024	256	1,024	12
Dutch Social (Gupta, 2022)	Dutch	1,024	256	1,024	12
SST5 (Socher et al., 2013)	English	1,024	256	2,048	12
Linguistic Acceptability					
ScaLA-da (Nielsen, 2023)	Danish	1,024	256	2,048	12
ScaLA-sv (Nielsen, 2023)	Swedish	1,024	256	2,048	12
ScaLA-nb (Nielsen, 2023)	Norwegian Bokmål	1,024	256	2,048	12
ScaLA-nn (Nielsen, 2023)	Norwegian Nynorsk	1,024	256	2,048	12
ScaLA-is (Nielsen, 2023)	Icelandic	1,024	256	2,048	12
ScaLA-fo (Nielsen, 2023)	Faroese	1,024	256	1,024	12
ScaLA-de (Nielsen, 2023)	German	1,024	256	2,048	12
ScaLA-nl (Nielsen, 2023)	Dutch	1,024	256	2,048	12
ScaLA-en (Nielsen, 2023)	English	1,024	256	2,048	12
Question Answering					
ScandiQA-da (Nielsen, 2023)	Danish	1,024	256	2,048	4
ScandiQA-sv (Nielsen, 2023)	Swedish	1,024	256	2,048	4
NorQuAD (Ivanova et al., 2023)	Norwegian Bokmål	1,024	256	2,048	2
NQil (Snæbjarnarson and Einarsson, 2022)	Icelandic	1,024	256	1,024	4
GermanQuAD (Möller et al., 2021)	German	1,024	256	2,048	4
SQuAD-nl (Havinga, 2023)	Dutch	1,024	256	2,048	4
SQuAD (Rajpurkar et al., 2016)	English	1,024	256	2,048	4

Table 1: All the datasets used in the NLU evaluation. Note that these have been re-sized and do not represent the sizes of the original dataset.

“very negative” and “very positive” labels to “negative” and “positive”, respectively.

3.3 Linguistic Acceptability

For linguistic acceptability we re-use the ScaLA datasets for all the Scandinavian languages, and extend the ScaLA datasets by applying the ScaLA method from Nielsen (2023) to German, Dutch and English by using the German (McDonald et al., 2013), Dutch (van der Beek et al., 2002) and English (Zeldes, 2017) dependency treebanks.

3.4 Extractive Question Answering

Here we use the ScandiQA dataset (Nielsen, 2023) for Danish and Swedish, but replace the manually translated Norwegian ScandiQA dataset with the new curated NorQuAD dataset (Ivanova et al., 2023). We further add the new Natural Questions in Icelandic dataset (Snæbjarnarson and Einarsson, 2022) for Icelandic. For German and English we add the existing extractive question-answering datasets GermanQuAD (Möller et al., 2021) and SQuAD (Rajpurkar et al., 2016), respectively. For Dutch we add the machine translated version of SQuAD to Dutch (Havinga, 2023).

4 Methodology

4.1 Formulating NLU Tasks as Generative Tasks

In this section we describe how we rephrase the NLU tasks as text-to-text tasks, which makes it possible to evaluate generative models on the tasks. We formulate all the tasks as few-shot tasks, generally formatted as follows:

```
[prefix prompt]

[document prefix]: [document]
[label prefix]: [label]

(...)

[document prefix]: [document]
[label prefix]:
```

We found that the separation of the few-shot examples with double newlines makes it easier to know when to stop the generation - for the same reason, we ensure that there are no double newlines in any of the documents. See the prompts used for the English datasets in Table 2; a full table of the prompts used for all the tasks in all the languages can be found in (Nielsen et al., 2024).

For the sentiment classification task, we simply have the models generate translations of the three labels (positive, negative and neutral). For the linguistic acceptability task, also a text classification task, we use the translations of “yes” and “no” as the two labels, corresponding to whether the document is grammatically correct or not. For the extractive question answering task, we have the model output the answer directly. For this task we found that changing the label prefix from “Answer” to “Answer in max 3 words” resulted in a drastic improvement, due to many of the answers of instruction tuned models starting with unnecessary text akin to “The answer is”. Lastly, for the named entity recognition task, we require the output to be a JSON dictionary (ISO/IEC 21778:2017), with keys being the translated named entity tags, and values being lists of named entities of that category. To ensure that we are not biasing the evaluation toward models knowing the JSON format, we employ structured generation using the outlines package (Louf, 2023), which modifies the logits outputted by the model to ensure that the output is always a valid JSON dictionary in the aforementioned format.

4.2 Evaluation Methodology

We keep the evaluation methodology for the generative models to be as close to the methodology for encoder models in Nielsen (2023). We think of the few-shot examples as analogous to training examples for encoder models. Indeed, as von Oswald et al. (2023) shows, this assumption is theoretically grounded. We thus evaluate the models 10 times, where on each iteration we sample few-shot examples at random from the training split, and we evaluate the model on a bootstrapped version of the test split. As with the encoder models, this allows us to take into account more noise in evaluation process, resulting in more robust evaluation scores.

The number of few-shot examples for each dataset was determined on a heuristic basis, where we wanted to include as many examples as possible, while making sure that the token count was sufficiently low to not bias the evaluation towards models with a longer context length. All the NER, sentiment classification and linguistic acceptability datasets have prompt sizes around 1,000 tokens with the Mistral-7B-v0.1 tokeniser (Jiang et al., 2023), with the question answering datasets having around 2,000 tokens. This is also the reason for

Task	Prefix Prompt	Example Prompt
Named entity recognition	Below are sentences and JSON dictionaries with the named entities that occur in the given sentence.	Sentence: [text] Named entities: [label]
Sentiment classification	The following are tweets are their sentiment, which can be 'positive', 'neutral' or 'negative'.	Tweet: [text] Sentiment: [label]
Linguistic acceptability	The following are sentences and whether they are grammatically correct.	Sentence: [text] Grammatically correct: [label]
Question answering	The following are texts with accompanying questions and answers.	Text: [text] Question: [question] Answer in max 3 words: [label]

Table 2: The English prompt templates used for the datasets. See all the prompt templates in (Nielsen et al., 2024).

the discrepancy with the NorQuAD dataset, as the samples are much longer than the other question answering datasets.

4.3 Score Aggregation Method

From the raw scores of the 10 evaluations per dataset, we need to aggregate the model scores into a single score. We want an aggregation method that satisfies the following criteria:

1. **Task Fairness:** Each task should be weighted equally.
2. **Comparison:** If we evaluate models in multiple languages, then it should be possible to meaningfully compare the language scores of these models with each other.
3. **Robustness:** If two models do not have a significantly different score on a dataset, then the aggregated score should reflect this.
4. **Magnitude Preservation:** The magnitude of the difference between the dataset score of two models should be reflected in the aggregated score.
5. **Minimal Change:** Adding a new model should minimally affect the aggregated scores of the other models.

Before we introduce our chosen aggregation method, we will briefly discuss some common aggregation methods and how they do not satisfy the criteria.

The **mean score** is the most common aggregation method, which would simply be the mean of the 10 scores for each dataset, and then the mean of the dataset scores for each task. This method does not satisfy the Task Fairness criterion, as it does

not take into account that metrics have different ranges and variances. The Comparison criterion is also not satisfied, as datasets vary from language to language, with some datasets being more difficult than others. It *does*, however, satisfy the Robustness, Magnitude Preservation and Minimal Change criteria.

The **mean rank** is another common aggregation method, where we compute the rank of each model on each dataset, and then take the mean of the ranks. This method satisfies the Task Fairness criterion, as it re-casts the scores into a common comparable framework, which therefore weights each task equally. For the same reason, it also satisfies the Comparison criterion (it is important here that we evaluate all the models on all the languages for this to be satisfied). It does not satisfy the Robustness and Magnitude Preservation criteria, by definition of rank. It partially satisfies the Minimal Change criterion, since it only affects the scores of the models which are worse than the new model.

We thus see that the mean score and mean rank methods satisfy a disjoint set of the criteria, but that they together satisfy all the criteria. Based on this observation, we introduce the **mean rank score** method, defined as follows. For each dataset, we start by sorting the models by their mean score on the dataset. As with a rank, we assign the best model with rank score 1. For the next best model, we conduct a one-tailed Welch’s t-test to see if the next best model is significantly worse than the first model ($p < 0.05$). If so, we compute the absolute difference between the mean score of the two models, and divide that by the standard deviation of all the mean scores of the models on the dataset.

We then add this to the rank score of the first model. We continue this process for all the models to get the rank scores for the dataset, and to

compute the overall score for the model, we take the mean of the rank scores for the datasets. An overview of this aggregation method can be found in (Nielsen et al., 2024). We note that the mean rank score has an intuitive interpretation: it is the average number of standard deviations from the best scoring model (+1).

This metric satisfies Task Fairness since we normalise all the scores by dividing by the standard deviation of the dataset scores. The Robustness criterion is satisfied due to our use of a one-tailed Welch’s t-test. The Magnitude Preservation criterion is also satisfied, as the magnitude of the difference between the dataset score of two models is reflected in the rank score. It also satisfies Comparison, as we compare the models on a common scale (same argument as the mean rank method). Finally, the Minimal Change criterion is partially satisfied, as adding new models only minimally changes the score of existing models. Concretely, adding new scores will affect the standard deviation normalising factor (this effect tends to zero as the number of models grows, however), and if the model beats all the other models then all the scores will be affected, due to the relative nature of the metric.

5 Analysis

5.1 Comparative Performance Analysis on High- and Low-resource Languages

Excerpts of the English, Danish and Icelandic leaderboards can be found in Table 3, Table 4 and Table 5, respectively. We found that these three represent three main categories of languages with respect to the open-closed source divide. Similar excerpts for the remaining languages (Swedish, Norwegian, Faroese, German and Dutch) can be found in (Nielsen et al., 2024). The full leaderboards for all the languages can be found at <https://scandeval.com>.

From the English results we see that the state-of-the-art decoder model GPT-4-0613 (Achiam et al., 2023) is still outperformed by the DeBERTa-v3-large and DeBERTa-v3-base models (He et al., 2020) as well as the ELECTRA-base model (Clark et al., 2020). Here GPT-4-0613 is, on average, 0.44 standard deviations worse than the best model. The same pattern is seen for Norwegian, Dutch, German and Faroese; see (Nielsen et al., 2024) for the corresponding leaderboard excerpts.

In contrast, on the Danish leaderboard, the top-3 models are all decoder models, with GPT-4-0613

Model ID	Decoder	Score (↓)
microsoft/deberta-v3-large	✗	1.09
microsoft/deberta-v3-base	✗	1.29
google/electra-base-discriminator	✗	1.39
gpt-4-0613	✓	1.44
FacebookAI/roberta-large	✗	1.46
FacebookAI/roberta-base	✗	1.51
microsoft/mdeberta-v3-base	✗	1.53
gpt-4-1106-preview	✓	1.54
gpt-4o-2024-05-13	✓	1.64
AI-Sweden-Models/roberta-large-1160k	✗	1.64
gpt-3.5-turbo-0613	✓	1.78
mistralai/Mistral-7B-v0.1	✓	1.91

Table 3: Excerpt of the English ScandEval leaderboard.

and GPT-4-1106-preview (OpenAI, 2023b) in the lead, followed by the closed-source DanskGPT-Chat-Llama3-70B model from Syv.AI¹, being a continuation of the Llama-3-70B model (AI@Meta, 2024). The GPT-4-0613 model is, on average, 0.24 standard deviations from the best model. Similar results were found with Swedish; see (Nielsen et al., 2024) for the corresponding leaderboard excerpt.

Lastly, for Icelandic, we see that the encoders and decoders are tied in performance, with the mDeBERTa-v3-base model and the GPT-4-1106-preview model being the top models. The GPT-4-1106-preview model is, on average, 0.24 standard deviations from the best model. We note that Icelandic is the *only* language where the switch from GPT-4 (gpt-4-0613) to GPT-4-turbo (gpt-4-1106-preview) resulted in a significant *increase* in performance. We speculate that this is due to the collaboration between OpenAI and Iceland (OpenAI, 2023a).

We can thus give an affirmative answer to research question (Q1), showing that encoder models *can* achieve significantly better NLU performance than decoder models, even though they have an order of magnitude fewer model parameters. For (Q3), we see that this varies between languages, but without being correlated to the language resource spectrum.

5.2 Task Analysis

In this section we analyse our research question (Q2), asking whether the NLU performance results from the previous section is dependent on the type of NLU task.

Firstly, we analyse whether the score distribution across the four NLU tasks is different for the encoder and decoder models. This is done by applying a UMAP (McInnes et al., 2018) to the results of a given leaderboard, which is a dimensionality

¹<https://www.syv.ai/>

Model ID	Decoder	Score (↓)
gpt-4-0613	✓	1.24
gpt-4-1106-preview	✓	1.25
syvai/dansk-gpt-chat-llama-3-70b	✓	1.29
AI-Sweden-Models/roberta-large-1160k	✗	1.39
danish-foundation-models/encoder-large-v1	✗	1.40
meta-llama/Meta-Llama-3-70B	✓	1.40
AI-Sweden-Models/Llama-3-8B-instruct	✓	1.44
gpt-4o-2024-05-13	✓	1.46
litg/norbert3-large	✗	1.50
NbAiLab/nb-bert-large	✗	1.54
vesteinn/DanskBERT	✗	1.56
google/rembert	✗	1.61
intfloat/multilingual-e5-large	✗	1.62
gpt-3.5-turbo-0613	✓	1.68
FacebookAI/xlm-roberta-large	✗	1.71

Table 4: Excerpt of the Danish ScandEval leaderboard.

Model ID	Decoder	Score (↓)
microsoft/mdebarta-v3-base	✗	1.33
gpt-4-1106-preview	✓	1.34
gpt-4o-2024-05-13	✓	1.43
vesteinn/ScandiBERT-no-faroese	✗	1.48
google/rembert	✗	1.57
vesteinn/XLMR-ENIS	✗	1.59
gpt-4-0613	✓	1.79
mideind/IceBERT-large	✗	1.85
vesteinn/FoBERT	✗	1.87
meta-llama/Meta-Llama-3-70B	✓	2.03
FacebookAI/xlm-roberta-large	✗	2.34
gpt-3.5-turbo-0613	✓	2.51
mistralai/Mistral-7B-v0.1	✓	2.96

Table 5: Excerpt of the Icelandic ScandEval leaderboard.

reduction method that both takes into account the global and local structure of the underlying data - it can thus be viewed as a middle ground between a principal component analysis (Pearson, 1901) and a t-distributed stochastic neighbour embedding (Hinton and Roweis, 2002). The resulting reduction thus contains a single two-dimensional representation of each model. UMAP plots for the English, Danish, Swedish, Norwegian, German and Dutch leaderboards can be found in Figure 1, where we also mark the mean rank score for each model, as well as whether the model is generative.

We see that the worst and best performing models have similar distributions, irrespective of whether they are generative or not. However, we also note that the rest of encoder and decoder models follow different “paths” in the UMAP space, leading to our hypothesis that the different architectures have different task preferences.

In Figure 2 we show the correlation between a model being generative and its performance on the four NLU tasks. We see that being generative is a strong predictor for good question answering performance, as well as poor named entity recognition and linguistic acceptability performance. The correlation is weaker for sentiment

classification and varies across languages. We also see that these findings seem to generalise across languages, both high- and low-resource. The large question answering performance persists for non-instruction-tuned decoder models (see the leaderboards at <https://scandeval.com>), showing a likely side-effect of the pre-training algorithm or the architecture of decoder models making them better at this task. We also note that generative models perform substantially better at the English sentiment classification dataset SST5 compared to the other sentiment classification datasets - we will return to this in the discussion.

6 Discussion

Having a good mean rank score is not the only thing that matters when choosing a model for a given task. Model size, inference speed and whether the model has publicly available weights are all important factors to consider. For this reason, we also include these metadata in the leaderboard, and we encourage the community to consider these factors when choosing a model for a given task.

Some of the datasets in the benchmark are translations of American datasets, which we acknowledge is not ideal and encourage the development of gold-standard replacements of these. This concerns the Dutch question answering dataset, which is machine translated, as well as the Danish and Swedish question answering datasets, where the questions and answers have been manually translated. Manual translations are typically better than machine translations, but it nevertheless means that the content is biased towards questions pertinent to the American context. Some datasets are furthermore missing. This concerns Icelandic and Faroese sentiment analysis, as well as Faroese question answering. Efforts are currently underway to remedy this.

Lastly, we note that the English sentiment classification dataset SST5 is the only dataset where generative models perform substantially better than encoder models. We speculate that this is either due to the dataset simply being significantly easier than the others, or that the test data has leaked into the pretraining datasets of the generative models. The dataset is part of the FLAN collection (Wei et al.), which is for instance included in the Dolma dataset (Soldaini et al., 2024), which is used to pretrain the OLMo model (Groeneveld et al., 2024), being one of the generative models that is performing very

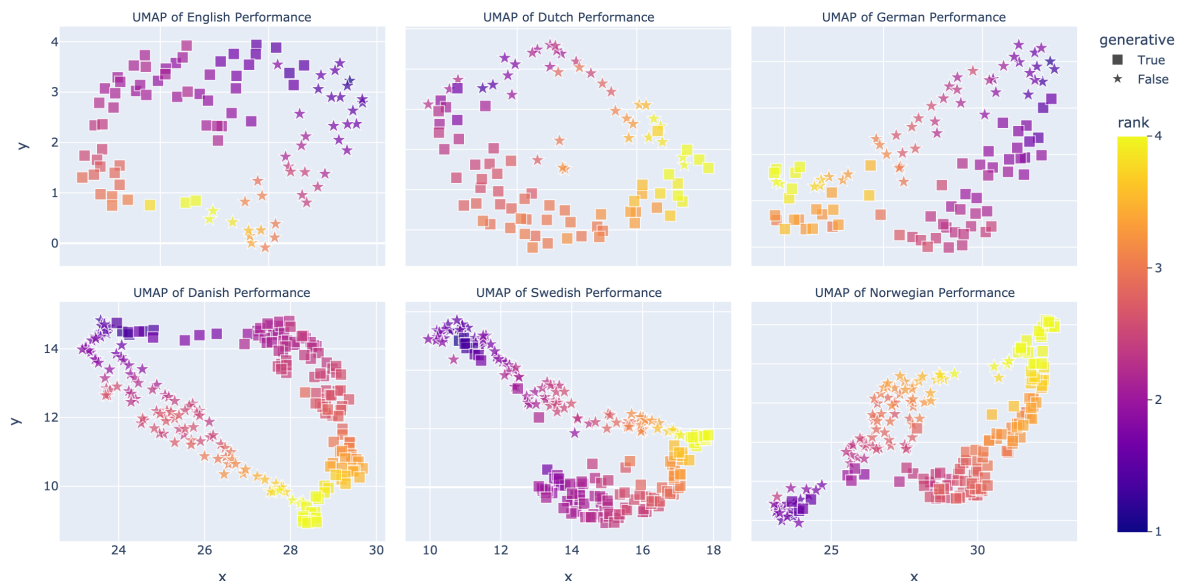


Figure 1: UMAP plots of the models on the ScandEval leaderboards.

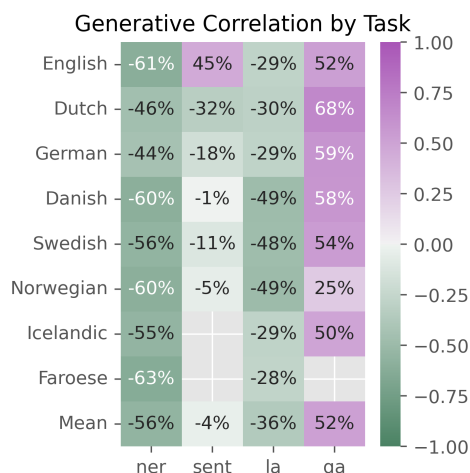


Figure 2: The correlation between a model being generative and its performance on the NLU tasks.

well on this dataset. Leakage is therefore possible, and we encourage the community to investigate this further.

7 Conclusion

We have extended the ScandEval benchmark to include the evaluation of decoder models, as well as including three new languages: German, Dutch and English. From the analysis of the corresponding results we found that encoder models can achieve significantly better NLU performance than

decoder models despite having orders of magnitude fewer parameters, but that this varies between languages. We have also shown that being generative is strongly correlated with both good question answering performance and poor performance for named entity recognition and linguistic acceptability. Our analysis showed that the “path” from the worst to the best-performing models in the UMAP space is different for encoder and decoder models, indicating an architecture-specific task-preference.

Ethics Statement

We have made efforts towards making the evaluation as fair and unbiased as possible, both through our selection of the datasets in the benchmark as well as through our choice of aggregation method of the scores. However, we have not conducted extensive bias analyses on the individual datasets.

Acknowledgements

This work has received funding by the European Union’s Horizon 2023 Research and Innovation Actions, as part of the Artificial Intelligence and Robotics programme, for the project “TrustLLM” (grant agreement number 101135671). Furthermore, this work reflects only the authors’ view and the European Research Executive Agency (REA) is not responsible for any use that may be made of the information it contains.

References

- Open LLM Leaderboard - a Hugging Face Space by open-llm-leaderboard — huggingface.co. https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard. [Accessed 12-06-2024].
- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- AI@Meta. 2024. Llama 3 model card.
- L van der Beek, G Bouma, R Malouf, and G van Noord. 2002. The alpino dependency treebank. In *12th Meeting on Computational Linguistics in the Netherlands (CLIN)*, pages 8–22. Rodopi.
- Darina Benikova, Chris Biemann, and Marc Reznicek. 2014. Nosta-d named entity annotation for german: Guidelines and dataset. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2524–2531.
- Aleksandrs Berdičevskis, Gerlof Bouma, Robin Kurtz, Felix Morger, Joey Öhman, Yvonne Adesam, Lars Borin, Dana Dannélls, Markus Forsberg, Tim Isbister, et al. 2023. Superlim: A swedish language understanding evaluation benchmark. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8137–8153.
- Rishi Bommasani, Percy Liang, and Tony Lee. 2023. Holistic evaluation of language models. *Annals of the New York Academy of Sciences*, 1525(1):140–146.
- Brown, Tom and Mann, Benjamin and Ryder, Nick and Subbiah, Melanie and Kaplan, Jared D and Dhariwal, Prafulla and Neelakantan, Arvind and Shyam, Pranav and Sastry, Girish and Askell, Amanda and others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.
- Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E Gonzalez, et al. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning*.
- Mark Cieliebak, Jan Milan Deriu, Dominic Egger, and Fatih Uzdilli. 2017. A twitter corpus and benchmark resources for german sentiment analysis. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 45–51.
- Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.
- 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*.
- Kenneth Enevoldsen, Emil Trenckner Jessen, and Rebekah Baglini. 2024a. Dansk and dacy 2.6. 0: Domain generalization of danish named entity recognition. *arXiv preprint arXiv:2402.18209*.
- Kenneth Enevoldsen, Márton Kardos, Niklas Muennighoff, and Kristoffer Laigaard Nielbo. 2024b. The scandinavian embedding benchmarks: Comprehensive assessment of multilingual and monolingual text embedding. *arXiv preprint arXiv:2406.02396*.
- Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac’h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2023. A framework for few-shot language model evaluation.
- Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. 2024. Olmo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838*.
- Aakash Gupta. 2022. dutchsocial · Datasets at Hugging Face — huggingface.co. https://huggingface.co/datasets/dutch_social. [revision: 8b7bc6230ebd78f04aa3661acb912f4567f21c76].
- Sofia Gustafson-Capková and Britt Hartmann. 2006. Manual of the Stockholm Umeå corpus version 2.0. *Stockholm University*.
- Yeb Havinga. 2023. squadv2dutch · Datasets at Hugging Face — huggingface.co. [revision: af494fe1b62762178d37c0b71b4a7160f0534f1a].
- Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.
- Geoffrey E Hinton and Sam Roweis. 2002. Stochastic neighbor embedding. *Advances in neural information processing systems*, 15.
- Søren Vejlgård Holm. 2024. Are gllms danoliterate? benchmarking generative nlp in danish.
- Rasmus Hvingelby, Amalie Brogaard Pauli, Maria Barrett, Christina Rosted, Lasse Malm Lidegaard, and Anders Søgaard. 2020. Dane: A named entity resource for danish. In *Proceedings of the 12th language resources and evaluation conference*, pages 4597–4604.

- Svanhvítt L Ingólfssdóttir, Ásmundur A Guðjónsson, and Hrafn Loftsson. 2020. Named Entity Recognition for Icelandic: Annotated Corpus and Models. In *International Conference on Statistical Language and Speech Processing*, pages 46–57. Springer.
- ISO/IEC 21778:2017. 2017. The JSON data interchange syntax. Standard, International Organization for Standardization, Geneva, CH.
- Sardana Ivanova, Fredrik Andreassen, Matias Jentoft, Sondre Wold, and Lilja Øvrelid. 2023. Norquad: Norwegian question answering dataset. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 159–168.
- AQ Jiang, A Sablayrolles, A Mensch, C Bamford, DS Chaplot, D de las Casas, F Bressand, G Lengyel, G Lample, L Saulnier, et al. 2023. Mistral 7b (2023). *arXiv preprint arXiv:2310.06825*.
- Fredrik Jørgensen, Tobias Aasmoe, Anne-Stine Ruud Husevåg, Lilja Øvrelid, and Erik Velldal. 2020. NorNE: Annotating named entities for Norwegian. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4547–4556, Marseille, France. European Language Resources Association.
- Jan Kocoń, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniec, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, et al. 2023. Chatgpt: Jack of all trades, master of none. *Information Fusion*, 99:101861.
- 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*.
- Rémi Louf. 2023. Outlines. <https://github.com/outlines-dev/outlines>.
- Ryan McDonald, Joakim Nivre, Yvonne Quirnbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, et al. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97.
- Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.
- Timo Möller, Julian Risch, and Malte Pietsch. 2021. Germanquad and germandpr: Improving non-english question answering and passage retrieval. In *Proceedings of the 3rd Workshop on Machine Reading for Question Answering*, pages 42–50.
- Dan Saattrup Nielsen. 2023. ScandEval: A Benchmark for Scandinavian Natural Language Processing. In *The 24rd Nordic Conference on Computational Linguistics*.
- Dan Saattrup Nielsen, Kenneth Enevoldsen, and Peter Schneider-Kamp. 2024. Encoder vs decoder: Comparative analysis of encoder and decoder language models on multilingual nlu tasks. *arXiv preprint arXiv:2406.13469*.
- OpenAI. 2023a. Government of Iceland: How Iceland is using GPT-4 to preserve its language. <https://openai.com/index/government-of-iceland>. [Accessed 12-06-2024].
- OpenAI. 2023b. New models and developer products announced at DevDay. <https://openai.com/blog/new-models-and-developer-products-announced-at-devday>. [Accessed 12-06-2024].
- Johannes von Oswald, Eyvind Niklasson, Maximilian Schlegel, Seijin Kobayashi, Nicolas Zucchet, Nino Scherrer, Nolan Miller, Mark Sandler, Max Vladymyrov, Razvan Pascanu, et al. 2023. Uncovering mesa-optimization algorithms in transformers. *arXiv preprint arXiv:2309.05858*.
- Amalie Brogaard Pauli, Maria Barrett, Ophélie Lacroix, and Rasmus Hvingelby. 2021. Danlp: An open-source toolkit for danish natural language processing. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 460–466.
- Karl Pearson. 1901. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11):559–572.
- Yunjian Qiu and Yan Jin. 2024. Chatgpt and finetuned bert: A comparative study for developing intelligent design support systems. *Intelligent Systems with Applications*, 21:200308.
- Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for ner. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164.
- 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.
- David Samuel, Andrey Kutuzov, Samia Touileb, Erik Velldal, Lilja Øvrelid, Egil Rønningstad, Elina Sigdel, and Anna Palatkina. 2023. Norbench—a benchmark for norwegian language models. *arXiv preprint arXiv:2305.03880*.
- Erik Tjong Kim Sang. 2002. Introduction to the conll-2002 shared task: Language-independent named entity recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*.

- Erik Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.
- Vésteinn Snæbjarnarson and Hafsteinn Einarsson. 2022. Natural questions in icelandic. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4488–4496.
- Vésteinn Snæbjarnarson, Annika Simonsen, Goran Glavaš, and Ivan Vulić. 2023. Transfer to a low-resource language via close relatives: The case study on faroese. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 728–737.
- Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.
- Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, et al. 2024. Dolma: An open corpus of three trillion tokens for language model pretraining research. *arXiv preprint arXiv:2402.00159*.
- Kristoffer Svensson. 2017. Sentiment Analysis With Convolutional Neural Networks: Classifying sentiment in Swedish reviews. Bachelor’s thesis.
- Erik Velldal, Lilja Øvrelid, Eivind Alexander Bergem, Cathrine Stadsnes, Samia Touileb, and Fredrik Jørgensen. 2018. NoReC: The Norwegian review corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Wietse de Vries, Martijn Wieling, and Malvina Nissim. 2023. Dumb: A benchmark for smart evaluation of dutch models. *arXiv preprint arXiv:2305.13026*.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.
- Zengzhi Wang, Qiming Xie, Yi Feng, Zixiang Ding, Zinong Yang, and Rui Xia. 2023. Is chatgpt a good sentiment analyzer? a preliminary study. *arXiv preprint arXiv:2304.04339*.
- Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.
- Amir Zeldes. 2017. The gum corpus: creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3):581–612.
- Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.
- Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, and Dacheng Tao. 2023. Can chatgpt understand too? a comparative study on chatgpt and fine-tuned bert. *arXiv preprint arXiv:2302.10198*.

Small Languages, Big Models: A Study of Continual Training on Languages of Norway

David Samuel Vladislav Mikhailov Erik Velldal Lilja Øvrelid
Lucas Charpentier Andrey Kutuzov Stephan Oepen

University of Oslo, Language Technology Group

davisamu@ifi.uio.no

Abstract

Training large language models requires vast amounts of data, posing a challenge for less widely spoken languages like Norwegian and even more so for truly low-resource languages like Northern Sámi. To address this issue, we present a novel three-stage continual training approach that substantially improves the downstream performance together with the inference efficiency for the target languages. Based on our findings, we train, evaluate, and openly release a new generative language model for Norwegian Bokmål, Nynorsk, and Northern Sámi with 11.4 billion parameters: *NorMistral-11B*.

1 Introduction

The development of large language models typically requires massive amounts of training data, which benefits wide-spread languages such as English, but poses a significant challenge for less widely spoken languages. Norwegian, with its two written standards Bokmål and Nynorsk,¹ currently has approximately 24B words available in our filtered text collection – about three orders of magnitude less than English (Penedo et al., 2024). The situation is even more challenging for Northern Sámi with only 40 million words available.²

¹While Bokmål is the main variety, roughly 15% of the Norwegian population uses Nynorsk. The two varieties are so closely related that they may be regarded as ‘written dialects’, but the lexical differences can be relatively large.

²The Sámi languages are a group of Uralic languages, of which Northern Sámi is the most widely used variant. With the number of speakers estimated to be between 15,000 and 25,000 in total across Norway, Sweden and Finland, it is still considered to be an endangered language. As the Sámi people are recognized as an Indigenous people in Norway, Sámi has status as an official language along with Norwegian.

To address this data scarcity, we propose a novel approach combining three key elements: knowledge transfer from existing models, data augmentation with related languages, and targeted upsampling. This method enables us to train an 11.4B-parameter model that achieves state-of-the-art performance across Norwegian language tasks while obtaining strong capabilities in Northern Sámi. The three main research contributions of this paper can be summarized as follows:

- 1. Novel training method for data-constrained language models** We propose a three-stage training method for efficient adaptation of existing language models to lower-resource languages. Our results demonstrate that this approach works well for adapting a Mistral model to Bokmål, Nynorsk and Northern Sámi. Our model achieves the *state-of-the-art* performance on tasks requiring deep linguistic understanding and world knowledge in Norwegian contexts – while being more than 30% faster than the original Mistral model on Norwegian inputs.
- 2. Flexible masked-causal model** We train a general language model that can act as a causal generative model as well as a fully-bidirectional embedding model. This approach allows it to be used as any other generative model while allowing future usage as a finetuned encoder model.
- 3. Truly open source** We openly release NorMistral-11B under a permissive Apache 2.0 license – <https://hf.co/norallm/normistral-11b-warm> – as well as three smaller 7B-parameter models and a new corpus for Northern Sámi. The model is trained on fully transparent corpora and evaluated on a robust set of prompts that are included in the paper. The training and evaluation scripts are available at <https://github.com/ltgoslo/norallm>.

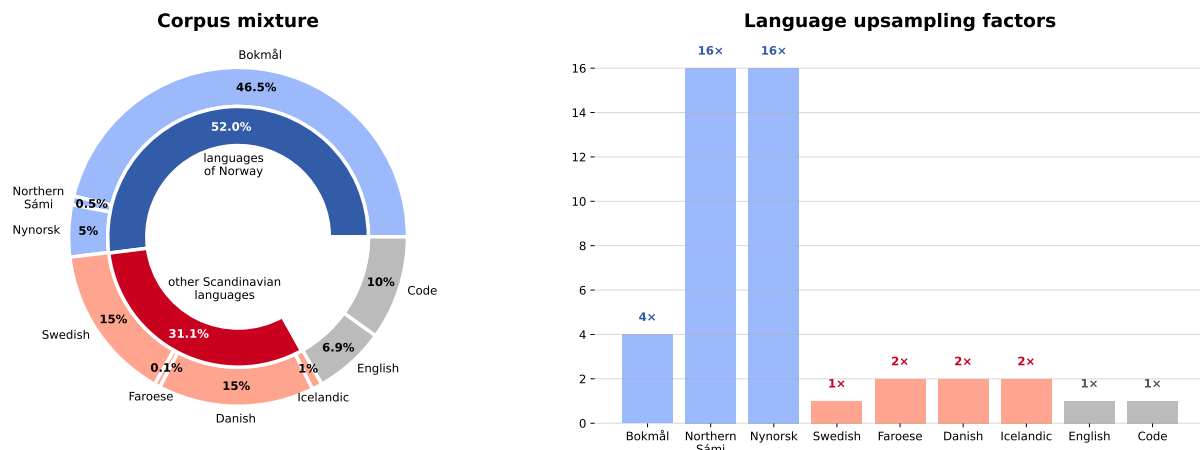


Figure 1: Language composition of training corpus The left figure shows the proportions of languages in the final corpus mixture, with the target languages of Norway in blue, related languages in red, and other data sources in gray. The right figure then displays the upsampling factors used to get the aforementioned proportions.

In the following sections, we first describe the training corpus of NorMistral-11B in [Section 2](#); then the training and evaluation methodology of this model in [Section 3](#). In [Section 4](#), we then evaluate this model and compare it against other existing models. The following [Section 5](#) then goes into more detail by testing the training choices in our methodology. [Section 6](#) describes previous works that inspired this paper. Additional appendices then offer further analyses and a detailed description of the evaluation setup.

2 Training corpus

Our goal is to train a model for the official languages of Norway. However, this task is made difficult by the uneven distribution of these languages and the fact that there is only about 24 billion words in these languages available in the publicly accessible high-quality corpora (see below).

2.1 Combating the data constraints

24B words is about three orders of magnitude less than what is currently available for English language models ([Penedo et al., 2024](#)). Assuming the Chinchilla scaling laws ([Hoffmann et al., 2022](#)), we could ‘optimally’ train only a 1-billion-parameter model on such a small dataset. However, we are able to train a much larger model due to: ① transferring knowledge from a model already trained on a large English-centric corpus; ② augmenting the corpus with other related Scandinavian languages (Danish, Swedish, Icelandic, and Faroese),

as well as English and programming code ([Luukkonen et al., 2024](#)); ③ further increasing the size by repeating the data in target languages – this follows the data-constrained scaling laws by [Muennighoff et al. \(2023\)](#), which showed that four repetitions do not have any noticeable negative effects on the regular scaling laws. The resulting corpus of 250B non-unique tokens is then ‘compute-optimal’ for the 11.4B parameters of our model ([Hoffmann et al., 2022](#)). In the previous work, the NorGPT-23B LLM trained on the available Norwegian data by [Liu et al. \(2024a\)](#) did not outperform smaller 3B models. Similarly, for Finnish, [Luukkonen et al. \(2023\)](#) reported a decrease in performance when moving from 8B to 13B parameters on a similarly sized corpus. These observations support our decision not to move beyond the 11B size.

2.2 Combating the uneven distribution

We target Norwegian and Sámi, the two official languages of Norway. Specifically, we target the *Bokmål* written variant of Norwegian with 24 billion words in our corpus, the *Nynorsk* variant with 0.5 billion words, and *Northern Sámi*, which has only 40 million words in our corpus collection. To mitigate the large size differences, we further up-sample the two lower-resource languages ([Conneau et al., 2020](#)). To avoid overfitting on many repetitions of the same data, we follow the experimental results in [Muennighoff et al. \(2023\)](#) and repeat the data at most 16 times. This approach yields the final language proportions shown in [Figure 1](#).

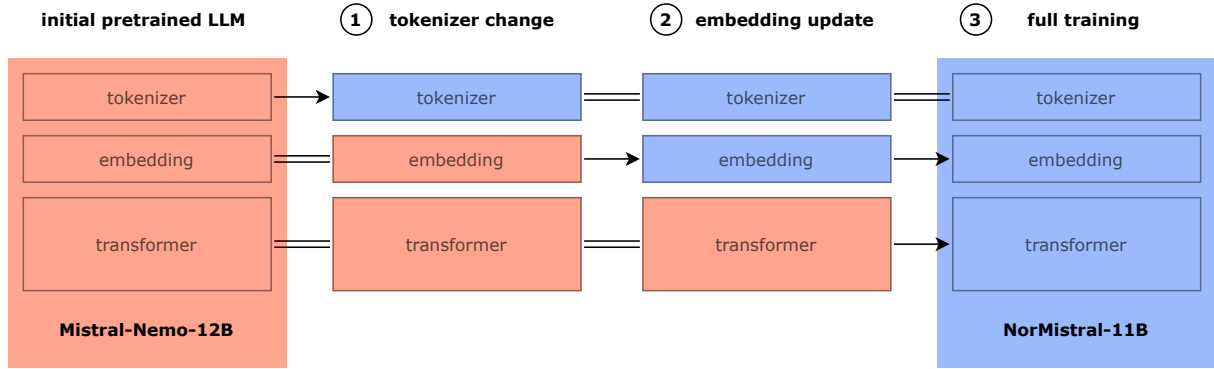


Figure 2: **Three-stage continual pretraining** We propose a novel continual pretraining pipeline consisting of ① creating a new tokenizer optimized for the training corpus, ② realigning the embedding weights to the new tokens, and ③ training the full language model. Arrows symbolize changes between stages, while double-lines represent no changes.

2.3 Data sources

Existing corpora We source most of the data from existing publicly available corpora: ① Bokmål and Nynorsk filtered from the public sources with permissive licenses from the *Mimir Core* corpus from de la Rosa et al. (2025), which itself consists of the Norwegian Colossal Corpus (NCC; Kummervold et al., 2022), CulturaX (Nguyen et al., 2023), and the HPLT corpus v1.2 (de Gibert et al., 2024); ② Bokmål, Nynorsk, Swedish, Danish, and Icelandic from *CulturaX* (Nguyen et al., 2023); ③ high-quality English from *FineWeb-edu* (Penedo et al., 2024); ④ code from the high-quality part of *Stack v2* (Lozhkov et al., 2024); ⑤ Faroese and Northern Sámi from *Glott500* (ImaniGooghari et al., 2023); and ⑥ Northern Sámi from the *SIKOR free corpus* (Giellatekno and Divvun, 2016).

Web crawl for Sámi The only exception to using existing resources is a part of the Sámi corpus. To obtain more texts for this low-resource language, we conducted a web crawl through admissible web pages in Northern Sámi. The crawl was seeded from the external links of the Sámi Wikipedia and continued with a breadth-first search through webpages that were identified as Northern Sámi using GlotLID (Kargaran et al., 2023) and that allowed crawling according to their Robots Exclusion Protocol. The raw HTML documents were converted into natural text using Trafilatura (Barbresi, 2021). We have published the web-crawled texts (fuzzy deduplicated at the document level) online at <https://hf.co/datasets/ltg/saami-web>. In total, it contains about 13 million whitespace-separated words.

3 Training and evaluation of NorMistral

This section describes the training and evaluation pipeline of NorMistral-11B; a continually trained Mistral-Nemo-Base-2407 language model.³ The presented methods are evaluated later in Section 5.

3.1 Three-stage continual pretraining

Our aim is to model three lower-resource languages. To achieve this, we rely on models initially trained on more resource-rich languages and continually train them on our corpus. In order to get a model that works efficiently for the target language, we propose a novel three-stage training process, which consists of tokenizer change, embedding update, and full training (Figure 2).

Stage 1: Tokenizer change Before training the language model, we create a new subword tokenizer optimized for the target distribution of languages. While keeping the original tokenizer might not necessarily worsen performance, the main goal of this step is to improve the efficiency of training and inference. As evident from Table 1, the new tokenizer produces 30% shorter sequences on average, which translates to more than 30% faster inference time; while requiring 800 million less parameters due to the smaller vocabulary size. We measure the inference speed-up on a downstream task in Appendix A, confirming the theoretical benefits.

The tokenizer is optimized for the entire training corpus via the greedy byte-pair encoding algorithm (BPE; Gage, 1994). We use the same tokenizer

³Available on HuggingFace at hf.co/mistralai/Mistral-Nemo-Base-2407

definition as the original Mistral-Nemo-12B: byte-level BPE tokenizer without any Unicode normalization and with a fairly complex pretokenizer regular expression. The pretokenizer splits numbers into individual digits as in [Chowdhery et al. \(2024\)](#). Note that the tokenization is completely lossless and reversible as out-of-vocabulary characters can be split into individual UTF-8 bytes that are always in-vocabulary as atomic tokens.

Stage 2: Embedding update Since all tokens are changed in the previous stage, we need to update the input and output embedding weights next. While it is possible to skip this stage and simply continue training the full model, misaligned embeddings lead to a large initial loss spike, to large (essentially random) gradients for the non-embedding parameters, and thus to catastrophic forgetting ([McCloskey and Cohen, 1989](#)). Instead, we follow the tokenizer adaptation method by [de Vries and Nisim \(2021\)](#), aligning the embedding parameters by continually training the language model for 1 000 steps with frozen non-embedding parameters.

The initial token embeddings are transferred from the original embedding matrix ([Gu et al., 2018](#); [Wang et al., 2019](#)). Since we use the same tokenizer type as the original Mistral model, many tokens are present in both vocabularies; the embeddings for these are initialized by as direct copies of the original vectors. Tokens not present in the original vocabulary are tokenized (with the original tokenizer) to obtain sub-tokens within the vocabulary; the embedding vectors are then initialized by taking the average of the sub-token embeddings.

Stage 3: Full training After realigning the embedding vectors, we continue by unfreezing the remaining parameters and training the full model.

The transformer architecture is inherited from the original Mistral model ([Jiang et al., 2023](#)), which is based on the improved Llama architecture ([Touvron et al., 2023](#)). This mainly entails: ① pre-normalization with the RMSNorm function for improved training stability ([Nguyen and Salazar, 2019](#); [Zhang and Sennrich, 2019](#)), ② SwiGLU activation function for improved expressive power of the feed-forward modules ([Shazeer, 2020](#)), ③ rotary positional embeddings for their ability to generalize to longer sequences ([Liu et al., 2024b](#); [Su et al., 2021](#)), and ④ grouped-query attention for improved inference efficiency ([Ainslie et al., 2023](#)). The remaining architectural details are based on

Tokenizer	# tokens	NOB	NNO	SME
Mistral-Nemo-12B	131 072	1.79	1.87	2.63
NorMistral-11B	51 200	1.22	1.28	1.82

Table 1: **Tokenizer statistics** The vocabulary size and subword-to-word split ratios of different tokenizers for Bokmål (NOB), Nynorsk (NNO) and Northern Sámi (SME). Lower split ratios result in shorter subword sequences and thus in faster training and inference.

the original transformer design by [Vaswani et al. \(2017\)](#). The hidden dimension is set to 5 120, the intermediate one to 14 336, and there are 40 layers in total. The attention modules have 32 query heads and 8 key & value heads, each of dimension 128. There are 51 200 tokens in the subword vocabulary.

We trained the model on 250 billion tokens, which equates to 60 000 steps of $1\,024 \times 4\,096$ tokens (number of samples \times sequence length). We used the trapezoidal learning-rate schedule with a peak learning rate of $1 \cdot 10^{-4}$, 1 000 warm-up steps and 10 000 decay steps; this schedule allows for further pretraining of this model on more tokens in the future ([Hägele et al., 2024](#)). The optimization was performed using AdamW ([Loshchilov and Hutter, 2019](#)), with $\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 10^{-8}$, and weight decay of 0.1. No dropout was applied.

The computations were conducted on 256 AMD MI250X GPUs and used 55 000 GPU hours in total – which equals to 8.5 days of runtime on the distributed setup. The model was trained with a reduced bfloat16 precision and the parameters were sharded with model parallelism – pipeline parallelism of 2, tensor parallelism of 2, and a zero-redundancy optimizer ([Rajbhandari et al., 2020](#); [Rasley et al., 2020](#); [Shoeybi et al., 2020](#)). The overall theoretical computation cost of the training was $1.7 \cdot 10^{22}$ FLOPs, with an average of 38% model FLOP/s utilization (MFU) on the actual hardware.

3.2 Hybrid masked-causal language modeling

While causal LMs have recently become very popular, the limited unidirectional text processing limits their learning abilities ([Lv et al., 2023](#)) and expressive power ([Ewer et al., 2024](#)); especially for finetuning ([Devlin et al., 2019](#); [Raffel et al., 2020](#)). Furthermore, it has been recently demonstrated that fully-bidirectional masked models share the same generative abilities, but without limitations

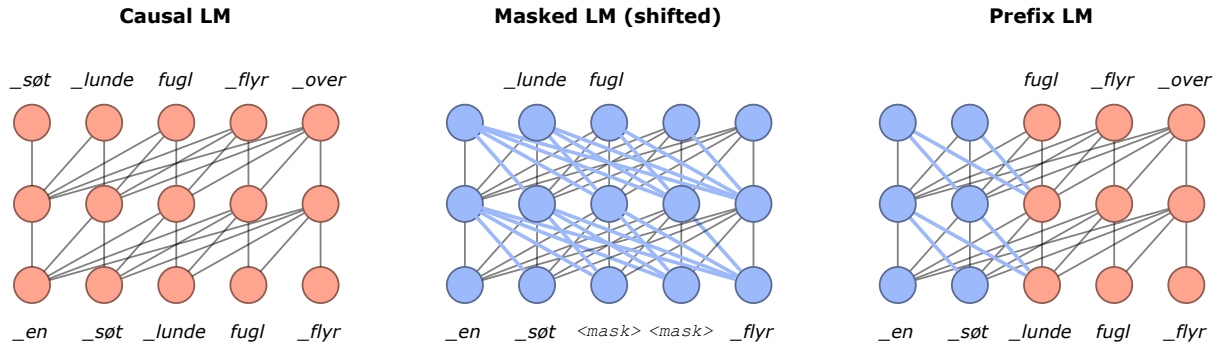


Figure 3: **Inference modes of NorMistral-11B** The hybrid masked-causal pretraining allows the model to be more flexible during inference. It can not only serve as a unidirectional causal language model (left), but also as a fully bidirectional masked language model (middle), or as a partially bidirectional prefix language model (right). The diagrams illustrate possible attention connections.

of causal models (Samuel, 2024). Following this observation, we train a model that can be flexibly used as a masked or causal language model.

Training objective We combine two training objectives during pretraining: the standard causal language modeling one as well as masked next-token prediction (MNTP; BehnamGhader et al., 2024; Lv et al., 2023), a variation of masked language modeling where the next token is predicted rather than the current one (see Masked LM (shifted) in Figure 3). This has been used by Charpentier and Samuel (2024), with evidence of providing better causal modeling quality and increased finetuning performance. We trained with 90% causal LM and 10% MNTP. This ratio is rather conservative – to teach the model bidirectional processing without drifting too much from its original training objective.

3.3 Experimental Setup

We compare the performance of NorMistral-11B with publicly available LMs using NorEval,⁴ an open-source framework for evaluating Norwegian generative LMs built on lm-evaluation-harness (Gao et al., 2024). The evaluation is run in k -shot scenarios with $k \in \{0, 1, 16\}$ on ten benchmarks. We report the maximum k for each benchmark across a set of prompts, which depends on the availability of a training/development set for demonstration examples and on the average length of these examples.

Baselines We use seven pretrained LMs of comparable size accessed via the Transformers li-

brary (Wolf et al., 2020) as our baselines: NorwAI-Mistral-7B, NorwAI-Llama2-7B, normistral-7b-warm, NorGPT-3B (Liu et al., 2024a), Viking-7B, Viking-13B, and Mistral-Nemo-12B.

Benchmarks The models are evaluated only on datasets created by native speakers. We consider the following language understanding and generation tasks: ① reading comprehension (NorQuAD; Ivanova et al., 2023 & Belebele; Bandarkar et al., 2024), ② sentiment analysis (NoReC; Velldal et al., 2018), ③ commonsense reasoning (NorCommonsenseQA; Mikhailov et al., 2025), ④ world knowledge (NRK-Quiz-QA & NorOpenBookQA; Mikhailov et al., 2025), ⑤ summarization (NorSumm; Touileb et al., 2025), ⑥ grammatical error correction (ASK-GEC; Jentoft, 2023), ⑦ language identification (SLIDE; <https://github.com/lrgoslo/slide>), and ⑧ translation (Tatoeba; Tiedemann, 2020). NorEval provides a set of task-specific 4–6 prompts written by Norwegian native speakers, which allows to account for prompt sensitivity (Lu et al., 2024). More details about each task with a complete list of prompts are given in Appendix B.

4 Results

We report the aggregated evaluation results in Table 2 and fine-grained evaluation results in Appendix B. Overall, we see a positive indication of NorMistral-11B being a strong Norwegian model as it outperforms other evaluated systems on the majority of tasks.

Comparison to the base model Even though Mistral-Nemo-12B is an English-centric model, it

⁴github.com/lrgoslo/noreval

Benchmark	Language	NorMistral-11B	NorwAI-Mistral-7B	NorwAI-Llama2-7B	NorMistral-7b-warm	NorGPT-3B	Viking-7B	Viking-13B	Mistral-Nemo-12B
READING COMPREHENSION									
Belebele (0-shot)	Bokmål	56.7	33.4	38.0	37.4	26.8	27.6	28.2	62.8
NorQuAD (1-shot)	Bokmål	76.7	63.0	39.2	64.8	3.0	48.4	57.1	76.5
SENTIMENT ANALYSIS									
NoReC (sentence-level; 16-shot)	Bokmål	90.5	88.6	86.0	84.9	49.7	77.9	79.2	86.9
NoReC (document-level; 1-shot)	Bokmål	91.2	81.2	79.2	82.9	51.5	80.4	86.8	89.2
COMMONSENSE REASONING									
NorCommonsenseQA (0-shot)	Bokmål	61.0	54.2	49.7	51.3	34.7	44.9	51.1	46.9
NorCommonsenseQA (0-shot)	Nynorsk	51.6	43.2	37.9	43.2	29.5	39.0	40.0	33.7
WORLD KNOWLEDGE									
NRK-Quiz-QA (0-shot)	Bokmål	63.7	55.2	52.3	57.9	33.1	44.2	51.0	47.4
NRK-Quiz-QA (0-shot)	Nynorsk	71.9	65.2	64.3	65.9	37.3	51.1	54.8	47.2
NorOpenBookQA (16-shot)	Bokmål	77.9	52.3	52.3	49.0	29.5	48.7	47.0	86.9
NorOpenBookQA (16-shot)	Nynorsk	77.8	45.6	38.9	41.1	34.4	27.8	36.7	86.7
SUMMARIZATION									
NorSumm (0-shot)	Bokmål	45.0	12.2	10.7	16.5	33.8	31.9	36.3	44.9
NorSumm (0-shot)	Nynorsk	32.6	10.3	10.4	8.6	24.3	25.7	28.8	30.9
GRAMMATICAL ERROR CORRECTION									
ASK-GEC (16-shot)	Bokmål	52.6	53.2	51.4	48.7	1.8	51.1	52.4	43.9
LANGUAGE IDENTIFICATION									
SLIDE (16-shot)	Bokmål, Nynorsk, Danish, Swedish	98.2	95.7	93.5	98.1	40.3	77.2	84.4	87.3
TRANSLATION									
Tatoeba (from English; 16-shot)	Bokmål	58.8	58.7	57.9	57.2	1.8	59.7	60.0	49.6
Tatoeba (from English; 16-shot)	Nynorsk	48.0	47.4	47.4	44.7	2.6	45.6	45.6	35.7
Tatoeba (from English; 16-shot)	Northern Sámi	50.4	27.5	28.5	18.5	0.0	7.8	11.6	6.5

Table 2: **Performance of NorMistral-11B** This table compares the performance of NorMistral-11B to the performance of other dense generative models that support Norwegian. All models are evaluated with the same fully-causal in-context-learning setup without any parameter updates. The best results are in bold; higher values are always better. The performance is evaluated by accuracy (Belebele, NorCommonsenseQA, NorOpenbookQA & NRK-Quiz-QA), F₁ score (NorQuAD & NoReC), ROUGE-L (Lin, 2004; NorSumm), ERRANT F_{0.5} (Bryant et al., 2017; ASK-GEC), accuracy (SLIDE), and BLEU (Papineni et al., 2002; Tatoeba). We report the maximum performance score across all prompts. The random guessing baselines are 20% for NorCommonSenseQA, 25% for Belebele and NorOpenBookQA, 28% / 27% for NRK-Quiz-QA NOB / NNO, and 48.5% / 48.4% for NoReC sentence-level / document-level.

performs well on the Norwegian benchmarks even before any continual pretraining. While we see a clear increase in performance after further training when evaluated on native Norwegian datasets, there is a notable decrease in performance on Bebebe (a well-known multilingual dataset) and NorOpen-BookQ (an adaptation of a popular English benchmark). This aspect requires a further study, but overall, we believe that the results clearly show the benefit of three-stage continual pretraining.

Bokmål, Nynorsk and Sámi performance We evaluate the models on all target languages: Bokmål, Nynorsk and Northern Sámi. Relative to other models, the performance gains of NorMistral-11B stay consistent across these three languages.

It is possible to estimate the difference in performance on Nynorsk compared to Bokmål when focusing on NorSumm, a dataset that is perfectly balanced and parallel for the two variants of Norwegian. The substantially higher score for Bokmål indicates that the much smaller amount of Nynorsk in the training corpus (even after upsampling) limits the downstream performance on this language variant.

The results on the English-to-Sámi translation suggest that our model was able to learn aspects of this language even though it made only 0.5% of the training corpus. However, any stronger claim about the level of understanding of Sámi would require a substantially more robust benchmarking suite than what is currently available.

4.1 Using NorMistral in practice

Large language models can be utilized in many different ways. We used the most direct and straightforward one for comparing Norwegian models – in-context learning – but there is a broader spectrum of methods with varying complexity-to-performance trade-offs. We evaluate the most common methods in Table 3 using NorQuAD:

In-context learning This is the most popular method of using large language models, mostly because it does not require any further training (Brown et al., 2020). Using just one sample from the training set as a demonstration can substantially improve the output quality on NorQuAD. More demonstrations can improve the performance further, but at the cost of reduced inference speed.

Quantization In order to reduce the large memory cost of large language models, a popular

Method	F ₁	EM	Runtime train / eval
0-shot (causal)	59.7	33.5	0 / 6 min
1-shot (causal)	76.7	55.3	0 / 8 min
8-shot (causal)	79.6	60.8	0 / 23 min
0-shot (4-bit, causal)	59.2	33.5	0 / 6 min
0-shot (8-bit, causal)	59.1	33.7	0 / 6 min
Full finetuning (causal)	90.4	79.2	57 / 6 min
Full finetuning (prefix)	92.2	80.3	57 / 6 min
LoRA finetuning (causal)	89.9	77.1	18 / 6 min
LoRA finetuning (prefix)	91.3	79.0	18 / 6 min

Table 3: **Evaluation methods** NorMistral-11B can be flexibly used in many different ways for solving downstream tasks. We compare them on NorQuAD, a dataset for extractive question answering. NorMistral can be finetuned as a standard causal language model and also as a partially bidirectional prefix language model. We also show the total training and evaluation time for each method (run on AMD MI250X GPUs). We use the two standard metrics for extractive question answering: F₁ score and exact-match accuracy (EM).

method is reducing the precision of their parameters. Specifically, we test 8-bit and 4-bit quantization (Dettmers et al., 2022; Dettmers and Zettlemoyer, 2023). There is no noticeable decrease of performance on NorQuAD when lowering the precision from the original 16 bits. Note that some GPUs can also increase their throughput at the lowered precision.

Full finetuning The best-performing strategy is to do supervised finetuning of all learnable parameters. This method is also the most difficult to set up, the large memory requirements necessitate distributed training with some model sharding. However, after finetuning, this method clearly outperforms all other ones without any additional cost. Interestingly, when finetuned with partially-bidirectional attention masks (as a prefix LM), the model even exceeds the estimated human performance on NorQuAD – 91.1 F₁ score and 78.1 EM accuracy (Ivanova et al., 2023).

LoRA finetuning Further training NorMistral on a downstream task is more demanding, but it is the preferred way for achieving the best performance – as long as there is a sizeable training set

available. Low-rank adaptation (LoRA) reduces the computational cost of finetuning by freezing all original model parameters and training only small low-rank adaptors (Hu et al., 2022). The resulting model is 10 F₁ percentage points better than the best few-shot prompt while running almost 4 times faster because of shorter context lengths. Because of its hybrid pretraining (Section 3.2), NorMistral can also be finetuned as a partially-bidirectional prefix language model, which further improves its performance by 1.4 points without any additional computational cost.

5 Methodological comparisons

We have conducted an initial comparative study of different training methods before settling on the pretraining process from Section 3 and training NorMistral-11B. The results are presented in Table 4, where different models are evaluated on a representative subset of available Norwegian benchmarks: extractive question answering (1-shot NorQuAD), binary sentence-level polarity classification (16-shot NoReC), world knowledge (0-shot NRK-Quiz-QA) and machine translation (16-shot English-to-Bokmål Tatoeba).

Architectural choice There are many promising improvements of the original GPT neural architecture (Radford et al., 2018) – we considered two recent and well-studied architectures: BLOOM (Scao et al., 2023) and Llama (Touvron et al., 2023), which is also used for training the Mistral models (Jiang et al., 2023). We adopted the training hyperparameters suggested by the respective papers and trained two models with 7 billion parameters on the same Norwegian corpus and with the same Norwegian tokenizer. Table 4 clearly shows that the Llama architecture is preferred for our training corpus and Norwegian benchmarks.

From scratch vs. warm-starting The central research question of this paper is how to train a good large language model for relatively small languages. Here we test our proposed three-stage continual pretraining and compare it against a model trained from scratch. For a fair comparison, we train two 7-billion-parameter models on the same Norwegian corpus (the Norwegian Colossal Corpus by Kummervold et al., 2021), and with the same architecture and tokenizer. Note that we do not consider existing methods that do not adapt the

Training method	NorQuAD 1-shot	NoReC 16-shot	NRK 0-shot	Tatoeba 16-shot
TRANSFORMER ARCHITECTURE				
BLOOM	43.6	67.6	44.6	52.2
Llama / Mistral	43.7	80.3	48.2	53.4
CONTINUAL TRAINING				
init. from scratch	43.7	80.3	48.2	53.4
three-stage continual	64.8	84.9	57.9	57.2
HYBRID TRAINING OBJECTIVE				
causal-only	67.0	86.0	59.0	58.8
hybrid masked-causal	69.3	87.5	55.4	58.2
TRAINING STEPS				
0 steps (base model)	76.5	86.9	47.4	49.6
0 steps (adapted tokenizer)	73.5	89.4	44.2	51.4
10,000 steps	69.3	87.5	55.4	58.2
20,000 steps	70.5	89.2	57.7	58.8
30,000 steps	66.2	82.3	59.0	58.5
40,000 steps	68.5	87.0	61.1	58.9
50,000 steps	70.4	88.7	60.2	58.7
60,000 steps	76.7	90.5	63.7	58.8

Table 4: **Comparison of training methods** The methods are compared on NorQuAD with F₁ score, sentence-level Bokmål NoReC with F₁ score, Bokmål NRK-Quiz-QA with accuracy, and on English-to-Bokmål Tatoeba with BLEU.

subword vocabulary – like simple continual training or adapter tuning (Yong et al., 2023) – because they necessarily lead to inefficient inference (Table 1). The results in Table 4 demonstrate that the knowledge transfer from an English-centric model works and the model is able to be adapted to new languages.

Hybrid masked-causal modeling Interestingly, we do not observe an overall increase in performance after training with the ‘dual’ training objective, as opposed to the observations by Charpentier and Samuel (2024). However, we believe that this can be explained by continued training – the hybrid masked-causal training is used for a negligible number of steps compared to the fully-causal pre-training of the base Mistral model.

Number of training steps Finally, we compare the performance of model checkpoints saved at different points of training. We can make several observations from the results: ① they confirm the data-scaling laws by Muennighoff et al. (2023) as the model continues to improve even after (at least) four repetitions of the Norwegian data; ②

tokenizer adaptation (the first two stages of our training method) is a simple and efficient way of adapting a model to a new language without losing performance; ③ the three-stage continual pretraining does not affect all downstream tasks equally – while it usually leads to monotonical improvement, there are some tasks (NorQuAD) that experience an initial decrease in performance. Further investigation is needed to determine if this drop is significant and if it can be avoided by a more careful switch to a new language distribution at the start of training.

6 Related work

Norwegian language models There have been several prior efforts on creating language models for Norwegian. When it comes to creating openly available generative decoder-only models for Norwegian, most of the main efforts are listed in [Section 3.3](#) and used in our experiments. However, one other notable mention is NB-GPT-J-6B – a fine-tuned version of the English GPT-J-6B model.⁵ Released by the National Library of Norway in 2022, it was the first large generative language model trained for Norwegian.

There have also been several efforts on developing smaller transformer models, e.g., based on the BERT encoder architecture ([Devlin et al., 2019](#)) and the T5 encoder-decoder architecture ([Raffel et al., 2020](#)). The NorBERT family of models were first released by [Kutuzov et al. \(2021\)](#) and have by now reached their third iteration of releases ([Samuel et al., 2023](#)) and come in several different sizes; ranging from 15M parameters for the XS model to 323M for NorBERT3 Large. [Samuel et al. \(2023\)](#) also introduced the NorT5 family of models, ranging 32M to 808M parameters. Whereas the above-mentioned models were all trained from scratch for Norwegian, [Kummervold et al. \(2021\)](#) trained NB-BERT (base and large) by fine-tuning the pre-trained mBERT model on Norwegian data, also reusing the tokenizer. A similar approach was followed for the North-T5 models.⁶

Language models for Northern Sámi As for Northern Sámi, [Paul et al. \(2024\)](#) has recently experimented with targeting this language. However, their models have not been published nor did they evaluate them on any downstream tasks; we are thus not able to compare them to our model.

⁵<https://huggingface.co/NbAiLab/nb-gpt-j-6B>

⁶<https://huggingface.co/north>

Continual training techniques Adaptation of pretrained language models to new domains by continual training has a long history ([Gururangan et al., 2020](#)). Our three-stage continual pretraining is designed specifically for adapting language models to a new language – by entirely replacing the original tokenizer, we can get an efficient model (by compressing the textual input into a short sequence of tokens) without the need of any extra parameters. Simple continual pretraining works well performance-wise but the training and inference computation cost is high ([Ibrahim et al., 2024](#)). A substantially more efficient approach is to introduce a new tokenizer and replace the embedding layers (first two stages of our approach), as proposed by ([Marchisio et al., 2023](#); [de Vries and Nissim, 2021](#)). Similarly, [Csaki et al. \(2023\)](#) only use the first and last stage of our method – they extend the vocabulary by 5 000 new tokens and then train the full model. On the other hand, [Kim et al. \(2024\)](#) pursue a more careful approach, the most similar to our training method. They first extend the subword vocabulary with extra tokens and then meticulously train the new and old parameters in eight subsequent stages.

7 Conclusion

We presented NorMistral-11B, a new large language model for Norwegian Bokmål, Nynorsk, and Northern Sámi. We proposed a novel three-stage continual pretraining approach that efficiently adapts existing models to other languages while maintaining high performance and increasing their inference speed. This approach involves training a new tokenizer, realigning embedding weights, and then training the full model. We also demonstrated the benefits of hybrid masked-causal pretraining, which allows the model to be used flexibly as either a causal or bidirectional model. Our extensive evaluation shows that NorMistral-11B achieves the state-of-the-art performance across a wide range of Norwegian tasks, while also showing promising results for Northern Sámi. This suggests that our approach could be beneficial for developing large language models for other smaller languages. To facilitate further research and development, we have released NorMistral-11B, the three 7B models trained for [Section 5](#), training code, and a new Northern Sámi corpus at <https://github.com/ltgoslo/norallm>.

Limitations

Limitations of the base language model Since NorMistral-11B is continually pretrained on the existing Mistral-Nemo-12B weights, the model is partially dependent on the training data of the original Mistral model. The exact composition of this training data is not known, which to some extent limits more detailed studies of this model. Specifically, the original model might have been trained on contaminated data, which could explain its high-scores on well-known evaluation tasks such as Belebele.

Computational cost As mentioned in Section 3, training NorMistral-11B took more than 55 000 GPU/hours. This is a significant amount. We have not yet estimated the CO₂ footprint of the full training, but it was conducted on the LUMI supercomputer which is powered exclusively with renewable electricity and deployed in one of the most eco-efficient data centers in the world.⁷

Evaluation of Northern Sámi knowledge Finally, our evaluation for Northern Sámi is limited to English-Sámi translation, which is obviously insufficient. Unfortunately, we lack more advanced or diverse benchmarks for low-resource languages like this one. We hope to see further development in this direction by the NLP community.

Acknowledgments

The computations were performed on resources provided through Sigma2 – the national research infrastructure provider for high-performance computing and large-scale data storage in Norway. We acknowledge Norway and Sigma2 for awarding this project access to the LUMI supercomputer, owned by the EuroHPC Joint Undertaking, hosted by CSC (Finland) and the LUMI consortium through project 465000498.

The efforts described in this paper were jointly funded by the University of Oslo and the HPLT project (High Performance Language Technologies; coordinated by Charles University).

The Norwegian part of our training corpus – Mímir-core – has been cleaned and graciously provided to us ahead of time by the National Library of Norway.

⁷<https://www.lumi-supercomputer.eu/sustainable-future/>

References

- Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. 2023. [GQA: Training generalized multi-query transformer models from multi-head checkpoints](#). In *The 2023 Conference on Empirical Methods in Natural Language Processing*.
- Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabza. 2024. [The belebele benchmark: a parallel reading comprehension dataset in 122 language variants](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775, Bangkok, Thailand. Association for Computational Linguistics.
- Adrien Barbaresi. 2021. [Trafilatura: A web scraping library and command-line tool for text discovery and extraction](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 122–131, Online. Association for Computational Linguistics.
- Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. [LLM2vec: Large language models are secretly powerful text encoders](#). In *First Conference on Language Modeling*.
- 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. Curran Associates, Inc.
- Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. [Automatic annotation and evaluation of error types for grammatical error correction](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 793–805, Vancouver, Canada. Association for Computational Linguistics.
- Lucas Georges Gabriel Charpentier and David Samuel. 2024. [GPT or BERT: why not both?](#)
- 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,

- Sashank Tsveyashchenko, 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 Sankaranarayanan 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. 2024. Palm: scaling language modeling with pathways. *J. Mach. Learn. Res.*, 24(1).
- 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.
- Zoltan Csaki, Pian Pawakapan, Urmish Thakker, and Qiantong Xu. 2023. [Efficiently adapting pretrained language models to new languages](#). In *Workshop on Efficient Natural Language and Speech Processing (ENLSP) at NeurIPS 2023*.
- Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022. [GPT3.int8\(\): 8-bit matrix multiplication for transformers at scale](#). In *Advances in Neural Information Processing Systems*.
- Tim Dettmers and Luke Zettlemoyer. 2023. [The case for 4-bit precision: k-bit inference scaling laws](#). In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 7750–7774. PMLR.
- 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.
- Ethan Ewer, Daewon Chae, Thomas Zeng, Jinkyu Kim, and Kangwook Lee. 2024. [ENTP: Encoder-only next token prediction](#).
- Philip Gage. 1994. [A new algorithm for data compression](#). *The C Users Journal archive*, 12:23–38.
- Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac’h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024. [A framework for few-shot language model evaluation](#).
- Ona de Gibert, Graeme Nail, Nikolay Arefyev, Marta Bañón, Jelmer van der Linde, Shaoxiong Ji, Jaime Zaragoza-Bernabeu, Mikko Aulamo, Gema Ramírez-Sánchez, Andrey Kutuzov, Sampo Pyysalo, Stephan Oepen, and Jörg Tiedemann. 2024. [A new massive multilingual dataset for high-performance language technologies](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 1116–1128, Torino, Italia. ELRA and ICCL.
- Giellatekno and Divvun. 2016. [SIKOR North Saami corpus](#).
- Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor O.K. Li. 2018. [Universal neural machine translation for extremely low resource languages](#). 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 344–354, New Orleans, Louisiana. 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, Online. Association for Computational Linguistics.
- Alexander Hägele, Elie Bakouch, Atli Kosson, Loubna Ben allal, Leandro Von Werra, and Martin Jaggi. 2024. [Scaling laws and compute-optimal training beyond fixed training durations](#). In *Workshop on Efficient Systems for Foundation Models II @ ICML2024*.
- Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Thomas Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karén Simonyan, Erich Elsen, Oriol Vinyals, Jack Rae, and Laurent Sifre. 2022. [An empirical analysis of compute-optimal large language model training](#). In *Advances in Neural Information Processing Systems*, volume 35, pages 30016–30030. Curran Associates, Inc.
- 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*.
- Adam Ibrahim, Benjamin Thérien, Kshitij Gupta, Mats Leon Richter, Quentin Gregory Anthony, Eugene Belilovsky, Timothée Lesort, and Irina Rish. 2024. [Simple and scalable strategies to continually pre-train large language models](#). *Transactions on Machine Learning Research*.
- Ayyoob ImaniGooghari, Peiqin Lin, Amir Hossein Kargaran, Silvia Severini, Masoud Jalili Sabet, Nora Kassner, Chunlan Ma, Helmut Schmid, André Martins, François Yvon, and Hinrich Schütze. 2023. [Glot500: Scaling multilingual corpora and language models to 500 languages](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1082–1117, Toronto, Canada. Association for Computational Linguistics.
- Sardana Ivanova, Fredrik Andreassen, Matias Jentoft, Sondre Wold, and Lilja Øvrelid. 2023. [NorQuAD: Norwegian question answering dataset](#). In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 159–168, Tórshavn, Faroe Islands. University of Tartu Library.
- Matias Jentoft. 2023. [Grammatical error correction with byte-level language models](#). Master’s thesis, University of Oslo.
- Albert Qiaochu Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Giana Lengyel, Guillaume Lample, Lucile Saulnier, L’elio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. [Mistral 7b](#). *ArXiv*, abs/2310.06825.
- Amir Hossein Kargaran, Ayyoob Imani, François Yvon, and Hinrich Schuetze. 2023. [GlotLID: Language identification for low-resource languages](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6155–6218, Singapore. Association for Computational Linguistics.
- Seungduk Kim, Seungtaek Choi, and Myeongho Jeong. 2024. [Efficient and effective vocabulary expansion towards multilingual large language models](#).
- Per Kummervold, Freddy Wetjen, and Javier de la Rosa. 2022. [The Norwegian colossal corpus: A text corpus for training large Norwegian language models](#). In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3852–3860, Marseille, France. European Language Resources Association.
- Per E Kummervold, Javier De la Rosa, Freddy Wetjen, and Svein Arne Brygfeld. 2021. [Operationalizing a national digital library: The case for a Norwegian transformer model](#). In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 20–29, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- Andrey Kutuzov, Jeremy Barnes, Erik Velldal, Lilja Øvrelid, and Stephan Oepen. 2021. [Large-scale contextualised language modelling for Norwegian](#). In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 30–40, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- 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.
- Peng Liu, Lemei Zhang, Terje Farup, Even W. Lauvra, Jon Espen Ingvaldsen, Simen Eide, Jon Atle Gulla, and Zhirong Yang. 2024a. [NLEBench+NorGLM: A comprehensive empirical analysis and benchmark dataset for generative language models in Norwegian](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5543–5560, Miami, Florida, USA. Association for Computational Linguistics.
- Xiaoran Liu, Hang Yan, Chenxin An, Xipeng Qiu, and Dahua Lin. 2024b. [Scaling laws of RoPE-based extrapolation](#). In *The Twelfth International Conference on Learning Representations*.
- Ilya Loshchilov and Frank Hutter. 2019. [Decoupled weight decay regularization](#). In *International Conference on Learning Representations*.
- Anton Lozhkov, Raymond Li, Loubna Ben Allal, Federico Cassano, Joel Lamy-Poirier, Nouamane Tazi, Ao Tang, Dmytro Pykhtar, Jiawei Liu, Yuxiang Wei, Tianyang Liu, Max Tian, Denis Kocetkov, Arthur Zucker, Younes Belkada, Zijian Wang, Qian Liu, Dmitry Abulkhanov, Indraneil Paul, Zhuang Li, WenDing Li, Megan Risdal, Jia Li, Jian Zhu, Terry Yue Zhuo, Evgenii Zheltonozhskii, Nii Osa Osa Dade, Wenhao Yu, Lucas Krauß, Naman Jain, Yixuan Su, Xuanli He, Manan Dey, Edoardo Abati, Yekun Chai, Niklas Muennighoff, Xiangru Tang, Muhtasham Oblokulov, Christopher Akiki, Marc Marone, Chenghao Mou, Mayank Mishra, Alex Gu, Binyuan Hui, Tri Dao, Armel Zebaze, Olivier Dehaene, Nicolas Patry, Canwen Xu, Julian McAuley, Han Hu, Torsten Scholak, Sebastien Paquet, Jennifer Robinson, Carolyn Jane Anderson, Nicolas Chapados, Mostofa Patwary, Nima Tajbakhsh, Yacine Jernite, Carlos Muñoz Ferrandis, Lingming Zhang, Sean Hughes, Thomas Wolf, Arjun Guha, Leandro von Werra, and Harm de Vries. 2024. [StarCoder 2 and The Stack v2: The next generation](#).
- Sheng Lu, Hendrik Schuff, and Iryna Gurevych. 2024. [How are prompts different in terms of sensitivity?](#) 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 5833–5856, Mexico City, Mexico. Association for Computational Linguistics.
- Risto Luukkonen, Jonathan Burdge, Elaine Zosa, Aarne Talman, Ville Komulainen, Vaino Hatanpää, Peter Sarlin, and Sampo Pyysalo. 2024. [Poro 34b and the blessing of multilinguality](#). *ArXiv*, abs/2404.01856.
- Risto Luukkonen, Ville Komulainen, Jouni Luoma, Anni Eskelinen, Jenna Kanerva, Hanna-Mari Kupari, Filip Ginter, Veronika Laippala, Niklas Muennighoff, Aleksandra Piktus, Thomas Wang, Nouamane Tazi, Teven Scao, Thomas Wolf, Osmo Suominen, Samuli Sairanen, Mikko Meriöksä, Jyrki Heinonen, Aija Vahtola, Samuel Antao, and Sampo Pyysalo. 2023. [FinGPT: Large generative models for a small language](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2710–2726, Singapore. Association for Computational Linguistics.
- Ang Lv, Kaiyi Zhang, Shufang Xie, Quan Tu, Yuhan Chen, Ji-Rong Wen, and Rui Yan. 2023. [Are we falling in a middle-intelligence trap? An analysis and mitigation of the reversal curse](#). *CoRR*, abs/2311.07468.
- Petter Mæhlum, David Samuel, Rebecka Maria Norman, Elma Jelin, Øyvind Andresen Bjertnæs, Lilja Øvrelid, and Erik Velldal. 2024. [It’s difficult to be neutral – human and LLM-based sentiment annotation of patient comments](#). In *Proceedings of the First Workshop on Patient-Oriented Language Processing (CL4Health) @ LREC-COLING 2024*, pages 8–19, Torino, Italia. ELRA and ICCL.
- Kelly Marchisio, Patrick Lewis, Yihong Chen, and Mikel Artetxe. 2023. [Mini-model adaptation: Efficiently extending pretrained models to new languages via aligned shallow training](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5474–5490, Toronto, Canada. Association for Computational Linguistics.
- Michael McCloskey and Neal J. Cohen. 1989. [Catastrophic interference in connectionist networks: The sequential learning problem](#). In Gordon H. Bower, editor, *Psychology of Learning and Motivation*, volume 24 of *Psychology of Learning and Motivation*, pages 109–165. Academic Press.
- Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. [Can a suit of armor conduct electricity? a new dataset for open book question answering](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391, Brussels, Belgium. Association for Computational Linguistics.
- Vladislav Mikhailov, Petter Mæhlum, Victoria Ovedie Chruickshank Langø, Erik Velldal, and Lilja Øvrelid. 2025. A collection of question answering datasets for Norwegian. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, Tallinn, Estonia.
- Niklas Muennighoff, Alexander M Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. 2023. [Scaling data-constrained language models](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- 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](#).
- Toan Q. Nguyen and Julian Salazar. 2019. [Transformers without tears: Improving the normalization of self-attention](#). In *Proceedings of the 16th International Conference on Spoken Language Translation*, Hong Kong. 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.
- Ronny Paul, Himanshu Buckchash, Shantipriya Parida, and Dilip K. Prasad. 2024. [Towards a more inclusive AI: Progress and perspectives in large language model training for the sámí language](#). *ArXiv*, abs/2405.05777.
- Guilherme Penedo, Hynek Kydlíček, Loubna Ben al-lal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. [The FineWeb datasets: Decanting the web for the finest text data at scale](#). In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- Maja Popović. 2017. [chrF++: words helping character n-grams](#). In *Proceedings of the Second Conference on Machine Translation: Research Papers*, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.
- Matt Post. 2018. [A call for clarity in reporting BLEU scores](#). In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.
- Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. [Improving language understanding by generative pre-training](#). *OpenAI Blog*.
- 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(1).

- Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. [ZeRO: memory optimizations toward training trillion parameter models](#). In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '20. IEEE Press.
- 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.
- Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. [Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters](#). In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 3505–3506, New York, NY, USA. Association for Computing Machinery.
- Javier de la Rosa, Vladislav Mikhailov, Lemei Zhang, Freddy Wetjen, David Samuel, Peng Liu, Rolv-Arild Braaten, Petter Mæhlum, Magnus Breder Birkenes, Andrey Kutuzov, Tita Enstad, Svein Arne Brygfeld, Jon Atle Gulla, Stephan Oepen, Erik Velldal, Wilfred Østgulen, Lilja Øvrelid, and Aslak Sira Myhre. 2025. The Impact of Copyrighted Material on Large Language Models: A Norwegian Perspective. In *Proceedings of the 25th Nordic Conference on Computational Linguistics (NoDaLiDa)*.
- David Samuel. 2024. [BERTs are generative in-context learners](#). In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- David Samuel, Andrey Kutuzov, Samia Touileb, Erik Velldal, Lilja Øvrelid, Egil Rønningstad, Elina Sigdel, and Anna Palatkina. 2023. [NorBench – a benchmark for Norwegian language models](#). In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 618–633, Tórshavn, Faroe Islands. University of Tartu Library.
- Teven Le Scao, Angela Fan, Christopher Akiki, Elie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, and et al. 2023. [BLOOM: A 176B-parameter open-access multilingual language model](#).
- Noam M. Shazeer. 2020. [GLU variants improve transformer](#). *ArXiv*, abs/2002.05202.
- Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2020. [Megatron-LM: Training multi-billion parameter language models using model parallelism](#).
- Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. 2021. [Roformer: Enhanced transformer with rotary position embedding](#).
- Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. [CommonsenseQA: A question answering challenge targeting commonsense knowledge](#). 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 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.
- Jörg Tiedemann. 2020. [The tatoeba translation challenge – realistic data sets for low resource and multilingual MT](#). In *Proceedings of the Fifth Conference on Machine Translation*, pages 1174–1182, Online. Association for Computational Linguistics.
- Samia Touileb, Vladislav Mikhailov, Marie Ingeborg Kroka, Øvrelid Lilja, and Erik Velldal. 2025. Benchmarking abstractive summarisation: A dataset of human-authored summaries of norwegian news articles. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, Tallin, Estonia.
- 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*, abs/2302.13971.
- 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.
- Erik Velldal, Lilja Øvrelid, Eivind Alexander Bergem, Cathrine Stadsnes, Samia Touileb, and Fredrik Jørgensen. 2018. [NoReC: The Norwegian review corpus](#). In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Wietse de Vries and Malvina Nissim. 2021. [As good as new. how to successfully recycle English GPT-2 to make models for other languages](#). In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 836–846, Online. Association for Computational Linguistics.
- Hai Wang, Dian Yu, Kai Sun, Jianshu Chen, and Dong Yu. 2019. [Improving pre-trained multilingual model with vocabulary expansion](#). In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 316–327, Hong Kong, China. 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.

Zheng Xin Yong, Hailey Schoelkopf, Niklas Muenighoff, Alham Fikri Aji, David Ifeoluwa Adelani, Khalid Almubarak, M Saiful Bari, Lintang Sutawika, Jungo Kasai, Ahmed Baruwa, Genta Winata, Stella Biderman, Edward Raff, Dragomir Radev, and Vasilina Nikoulina. 2023. [BLOOM+1: Adding language support to BLOOM for zero-shot prompting](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11682–11703, Toronto, Canada. Association for Computational Linguistics.

Biao Zhang and Rico Sennrich. 2019. [Root Mean Square Layer Normalization](#). In *Advances in Neural Information Processing Systems 32*, Vancouver, Canada.

A Inference efficiency of three-stage continual pretraining

In order to provide evidence for our claim that three-stage continual pretraining is necessary to increase the inference efficiency, we measure the actual inference speed on downstream tasks. Note that we specifically focus on the first stage of our pretraining recipe – creating a brand new tokenizer for the target domain. Since in-context-learning evaluation can be done in two modes – classification or generation – we measure the inference speed on both of them. Since the model quality might influence the number of generated tokens, we constrain the generation to only output tokens from the gold answers.

Speedup due to a new tokenizer We compare the speed of the original language model, Mistral-Nemo-12B, with the speed of our model that was initialized from it, NorMistral 11B. The results in Table 5 show that completely changing the tokenizer results in a noticeable speed up in both tests.

Other evaluated models For completeness, the inference speed of other models used in this paper are included as well; even though they have different number of non-embedding parameters or even completely different architectures. These additional measurements also show the benefit of replacing the entire vocabulary instead of only extending it with additional tokens.

SENTENCE-LEVEL NoReC (16-SHOT)

Model	Vocabulary	Note	Average length	Time / sample	Slowdown
NorMistral-11B	51 200	<i>our new Norwegian tokenizer</i>	522 tokens	0.23 s	1×
Mistral-Nemo-12B	131 072	<i>original English-centric tokenizer</i>	640 tokens	0.30 s	1.30×
NorwAI-Mistral-7B	67 993	<i>extends an English tokenizer</i>	591 tokens	0.18 s	0.78×
NorwAI-Llama2-7B	67 993	<i>extends an English tokenizer</i>	591 tokens	0.15 s	0.65×
NorMistral-7B-warm	32 768	<i>new Norwegian tokenizer</i>	569 tokens	0.17 s	0.74×
NorGPT-3B	64 000	<i>new Norwegian tokenizer</i>	552 tokens	0.08 s	0.35×
Viking-7B	131 072	<i>new Nordic tokenizer</i>	512 tokens	0.14 s	0.61×
Viking-13B	131 072	<i>new Nordic tokenizer</i>	512 tokens	0.26 s	1.13×

NORQUAD (8-SHOT)

Model	Vocabulary	Note	Average length	Time / sample	Slowdown
NorMistral-11B	51 200	<i>our new Norwegian tokenizer</i>	4 909 tokens	3.10 s	1×
Mistral-Nemo-12B	131 072	<i>original English-centric tokenizer</i>	6 171 tokens	4.13 s	1.33×
NorwAI-Mistral-7B	67 993	<i>extends an English tokenizer</i>	5 206 tokens	2.28 s	0.73×
NorwAI-Llama2-7B	67 993	<i>extends an English tokenizer</i>	5 206 tokens	2.10 s	0.68×
NorMistral-7B-warm	32 768	<i>new Norwegian tokenizer</i>	5 012 tokens	2.22 s	0.71×
NorGPT-3B	64 000	<i>new Norwegian tokenizer</i>	4 604 tokens	—	—
Viking-7B	131 072	<i>new Nordic tokenizer</i>	4 810 tokens	1.93 s	0.62×
Viking-13B	131 072	<i>new Nordic tokenizer</i>	4 810 tokens	—	—

Table 5: **Inference speed with different tokenization strategies** We measure the average sequence length that a model needs to preprocess per sample, as well as the average processing time per sample. These statistics are measured on a classification task (NoReC) as well as on a generative task (NorQuAD). Some models were not able to process the dataset, either because of not supporting long-enough input sequences or because of out-of-memory errors.

B Evaluation details

We provide a complete description of the evaluation design in this appendix. We provide inference details and prompts as well as full non-aggregated results here. Further information can be found at <https://github.com/lrgoslo/noreval> and <https://github.com/lrgoslo/norallm>.

B.1 Belebele

Belebele is a reading comprehension benchmark for evaluating the natural language understanding of language models (Bandarkar et al., 2024).

Inference setup The model is given a test example formatted according to a prompt template and ranks the answer candidates based on their probabilities. The most probable answer candidate is selected as the resulting answer.

Performance metric There are four possible answers for each passage-question pair. We measure the performance with a simple accuracy.

Prompt templates We used the following five prompt templates from NorEval.

Prompt A:

```
1 Tekst: {$passage}
2 Spørsmål: {$question}
3 A: {$answer_1}
4 B: {$answer_2}
5 C: {$answer_3}
6 D: {$answer_4}
7 Svar: {$prediction:A/B/C/D}
```

Prompt B:

```
1 Bakgrunn: {$passage}
2 Spørsmål: {$question}
3 Svaralternativer:
4 - {$answer_1}
5 - {$answer_2}
6 - {$answer_3}
7 - {$answer_4}
8 Svar: {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}}
```

Prompt C:

```
1 {$question}
2 Hvilket av følgende mulige svar er det riktige?
3 A: {$answer_1}
4 B: {$answer_2}
5 C: {$answer_3}
6 D: {$answer_4}
7 Svar: {$prediction:A/B/C/D}
```

Prompt D:

```
1 Svar på følgende spørsmål: {$question}
2 Svaret skal baseres på følgende tekst:
3 {$passage}
4 Velg et svar fra denne listen:
5 - {$answer_1}
6 - {$answer_2}
7 - {$answer_3}
8 - {$answer_4}
9 Svar: {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}}
```

Prompt E:

```

1  {$passage}
2
3  {$question}
4
5  A: {$answer_1}
6  B: {$answer_2}
7  C: {$answer_3}
8  D: {$answer_4}
9
10 Er det riktige svaret A, B, C, eller D? {$prediction:A/B/C/D}

```

Full results The complete evaluation results on Belebele (Bokmål) are given in Table 6. Note that the random-guessing baseline on this task achieves accuracy of 25%.

Prompt template	0-shot				
	A	B	C	D	E
NorMistral-11B	45.2	56.7	32.6	31.1	22.8
NorwAI-Mistral-7B	29.6	33.4	27.2	24.8	22.9
NorwAI-Llama2-7B	29.6	38.0	26.4	25.9	21.2
NorMistral-7B-warm	22.9	37.4	23.2	27.0	23.0
NorGPT-3B	22.2	26.8	22.9	25.7	22.9
Viking-7B	23.8	27.6	25.4	26.1	22.8
Viking-13B	27.3	27.3	28.2	25.1	22.8
Mistral-Nemo-12B	60.6	62.8	38.1	28.4	27.0

Table 6: **Complete results on Belebele question answering (Bokmål)** We show the detailed results for each evaluated model and prompt template. The best results for each column are boldfaced, the overall best result is highlighted in blue.

B.2 NorQuAD

The second benchmark for reading comprehension, NorQuAD by Ivanova et al. (2023), follows the scheme of extractive question-answering from SQuAD (Rajpurkar et al., 2016).

Inference setup The model is given a test example formatted according to a prompt template and generates an answer via the greedy-search decoding strategy.

Performance metrics The performance metrics are exact match (the percentage of predictions that exactly match the gold answer) and F₁-score (the average N-gram overlap between the prediction and the gold answer treated as bag-of-words).

Prompt templates We used the following five prompt templates from NorEval.

Prompt A:

```

1  Tittel: {$title}
2
3  Tekst: {$passage}
4
5  Spørsmål: {$question}
6
7  Svar: {$prediction}

```

Prompt B:

```
1 Tittel: {$title}
2
3 Tekst: {$passage}
4
5 Gitt teksten over, hva er svaret på følgende spørsmål? "{$question}"
6
7 Svar: {$prediction}
```

Prompt C:

```
1 Tittel: {$title}
2
3 Tekst: {$passage}
4
5 Svar på følgende: {$question}
6
7 Svar: {$prediction}
```

Prompt D:

```
1 Tittel: {$title}
2
3 Tekst: {$passage}
4
5 Hvordan kan man svare på spørsmålet "{$question}", gitt teksten over?
6
7 Svar: {$prediction}
```

Prompt E:

```
1 Tittel: {$title}
2
3 Tekst: {$passage}
4
5 Gitt teksten over, besvar følgende spørsmål: "{$question}"
6
7 Svar: {$prediction}
```

Full results The complete evaluation results on NorQuAD (Bokmål) can be found in [Table 7](#), both F_1 scores and exact-match accuracies.

B.3 Sentiment analysis

Sentiment analysis can serve as a good indicator of language understanding when evaluating language models. We use NoReC as a source of manually-annotated data for sentiment analysis ([Velldal et al., 2018](#)). While it offers fine-grained 6-class sentiment labels, we simplify the task to binary sentiment analysis, which works more reliably for in-context learning ([Mæhlum et al., 2024](#)).

Inference setup The model is given a test example formatted according to a prompt template and ranks the answer candidates based on their probabilities. The most probable answer candidate is selected as the resulting answer.

Performance metrics The dataset is slightly unbalanced and so we use the macro-average F_1 -score to assess the performance.

F ₁ SCORE										
Prompt template	0-shot					1-shot				
	A	B	C	D	E	A	B	C	D	E
NorMistral-11B	35.4	32.8	37.9	16.5	31.8	54.4	55.3	53.0	50.6	52.8
NorwAI-Mistral-7B	28.4	22.0	28.6	8.5	21.6	41.3	40.7	41.5	37.7	42.6
NorwAI-Llama2-7B	23.1	18.0	24.2	7.4	16.7	34.5	39.2	36.4	35.2	37.7
NorMistral-7B-warm	24.8	21.0	23.7	3.2	17.6	37.1	41.9	40.7	36.0	41.3
NorGPT-3B	1.1	1.1	0.2	0.4	0.6	0.0	0.0	0.0	0.0	0.0
Viking-7B	15.0	20.3	16.9	7.6	20.3	28.8	29.9	27.3	26.3	29.7
Viking-13B	19.1	22.5	20.8	11.9	22.5	35.8	35.8	35.8	33.1	35.6
Mistral-Nemo-12B	27.3	34.3	29.2	17.2	31.6	49.4	56.4	49.4	53.8	53.4

EXACT MATCH										
Prompt template	0-shot					1-shot				
	A	B	C	D	E	A	B	C	D	E
NorMistral-11B	35.4	32.8	37.9	16.5	31.8	54.4	55.3	53.0	50.6	52.8
NorwAI-Mistral-7B	28.4	22.0	28.6	8.5	21.6	41.3	40.7	41.5	37.7	42.6
NorwAI-Llama2-7B	23.1	18.0	24.2	7.4	16.7	34.5	39.2	36.4	35.2	37.7
NorMistral-7B-warm	24.8	21.0	23.7	3.2	17.6	37.1	41.9	40.7	36.0	41.3
NorGPT-3B	1.1	1.1	0.2	0.4	0.6	0.0	0.0	0.0	0.0	0.0
Viking-7B	15.0	20.3	16.9	7.6	20.3	28.8	29.9	27.3	26.3	29.7
Viking-13B	19.1	22.5	20.8	11.9	22.5	35.8	35.8	35.8	33.1	35.6
Mistral-Nemo-12B	27.3	34.3	29.2	17.2	31.6	49.4	56.4	49.4	53.8	53.4

Table 7: **Complete results on extractive question answering with NorQuAD** We show the detailed results for each evaluated model, few-shot setting and prompt template. The best results for each column are boldfaced, the overall best result for each few-shot setting is highlighted in blue.

B.3.1 Sentence-level NoReC

The converted dataset with binary sentiment labels can be found at https://huggingface.co/datasets/litg/norec_sentence.

Prompt templates We used the following five prompt templates from NorEval.

Prompt A:

```
1  Tekst: {$text}
2  Sentiment: {$prediction:positiv/negativ}
```

Prompt B:

```
1  {$text}
2  Er denne setningen "positiv" eller "negativ"? {$prediction:positiv/negativ}
```

Prompt C:

```
1  {$text}
2  Hva slags sentiment uttrykker anmelderen? {$prediction:positiv/negativ}
```

Prompt D:

```
1  {$text}
2  Er anmeldelsen "positiv" eller "negativ"? {$prediction:positiv/negativ}
```

Prompt E:

```

1  {$text}
2  Er denne setningen positiv eller negativ? {$prediction:positiv/negativ}

```

Full results The complete evaluation results on sentence-level NoReC are given in Table 8. The random-guessing baseline achieves 48.5% on this task.

Prompt template	0-shot					1-shot					16-shot				
	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E
NorMistral-11B	73.9	68.8	68.8	69.1	68.8	88.5^{±1.3}	65.4 ^{±2.0}	79.8 ^{±1.7}	76.3 ^{±1.8}	72.4 ^{±1.9}	90.2^{±1.2}	91.8^{±1.1}	90.2 ^{±1.2}	91.1^{±1.2}	91.6^{±1.2}
NorwAI-Mistral-7B	69.8	55.7	72.7	53.3	63.0	76.0 ^{±1.8}	55.1 ^{±2.1}	81.1 ^{±1.6}	77.2 ^{±1.7}	56.1 ^{±2.1}	89.0 ^{±1.3}	86.4 ^{±1.4}	90.4^{±1.2}	87.3 ^{±1.4}	87.8 ^{±1.4}
NorwAI-Llama2-7B	67.2	54.9	69.1	36.7	58.5	72.0 ^{±1.9}	64.3 ^{±2.0}	65.9 ^{±2.0}	63.6 ^{±2.0}	65.5 ^{±2.0}	88.2 ^{±1.3}	83.5 ^{±1.5}	88.9 ^{±1.3}	82.8 ^{±1.6}	87.5 ^{±1.4}
NorMistral-7B-warm	75.0	68.4	61.9	54.7	69.0	81.6 ^{±1.6}	69.1 ^{±1.9}	74.6 ^{±1.8}	71.5 ^{±1.9}	69.3 ^{±1.9}	86.6 ^{±1.4}	72.4 ^{±1.9}	85.9 ^{±1.4}	77.2 ^{±1.7}	72.6 ^{±1.8}
NorGPT-3B	72.4	41.2	47.3	71.4	67.9	66.0 ^{±2.0}	61.2 ^{±2.0}	61.9 ^{±2.0}	64.3 ^{±2.0}	59.3 ^{±2.0}	58.0 ^{±2.0}	48.9 ^{±2.1}	65.2 ^{±2.0}	48.7 ^{±2.1}	51.1 ^{±2.1}
Viking-7B	70.5	69.0	70.8	59.0	67.4	79.4 ^{±1.7}	70.8 ^{±1.9}	74.4 ^{±1.8}	73.6 ^{±1.8}	55.6 ^{±2.1}	81.8 ^{±1.6}	77.4 ^{±1.7}	73.8 ^{±1.8}	76.2 ^{±1.8}	82.5 ^{±1.6}
Viking-13B	69.1	69.1	68.1	50.8	68.1	78.9 ^{±1.7}	69.0 ^{±1.9}	79.9 ^{±1.7}	71.5 ^{±1.9}	69.0 ^{±1.9}	84.0 ^{±1.5}	77.4 ^{±1.7}	83.0 ^{±1.6}	80.4 ^{±1.6}	79.2 ^{±1.7}
Mistral-Nemo-12B	71.9	68.4	68.8	68.4	69.0	84.4 ^{±1.5}	77.4^{±1.7}	84.6^{±1.5}	82.0^{±1.6}	80.3^{±1.6}	87.0 ^{±1.4}	89.0 ^{±1.3}	88.2 ^{±1.3}	87.5 ^{±1.4}	88.9 ^{±1.3}

Table 8: Complete results on sentence-level sentiment analysis with NoReC We show the detailed results for each evaluated model, few-shot setting and prompt template. As the few-shot demonstrations are sampled randomly, we repeat them five times and show the mean accuracy as well as the standard deviation (rendered as superscript). The best results for each column are boldfaced, the overall best result for each few-shot setting is highlighted in blue.

B.3.2 Document-level NoReC

The converted dataset with binary sentiment labels can be found at https://huggingface.co/datasets/litg/norec_document.

Prompt templates We used the following five prompt templates from NorEval for testing all language models on document-level sentiment analysis:

Prompt A:

```

1  Tekst: {$text}
2  Sentiment: {$prediction:positiv/negativ}

```

Prompt B:

```

1  Tekst: {$text}
2  Er anmeldelsen "positiv" eller "negativ"? {$prediction:positiv/negativ}

```

Prompt C:

```

1  Er polariteten til følgende anmeldelse positiv eller negativ?
2  Anmeldelse: {$text}
3  Anmeldelsen er {$prediction:positiv/negativ}

```

Prompt D:

```

1  Anmeldelse: {$text}
2  Er anmelderen positiv eller negativ? {$prediction:positiv/negativ}

```

Prompt E:

```

1  Anmeldelse: {$text}
2  Vil du oppsummere anmeldelsen som "bra" eller "dårlig"? {$prediction:bra/dårlig}

```

Full results The complete evaluation results on document-level NoReC are provided in Table 9. The random-guessing baseline achieves 48.4% on this task.

Prompt template	0-shot					1-shot				
	A	B	C	D	E	A	B	C	D	E
NorMistral-11B	69.4	67.0	66.6	68.9	67.0	88.4 ^{±0.6}	92.5^{±0.5}	86.0 ^{±0.6}	92.5^{±0.5}	87.7 ^{±0.6}
NorwAI-Mistral-7B	69.1	66.1	67.0	66.5	63.8	82.8 ^{±0.7}	70.4 ^{±0.8}	84.6 ^{±0.7}	77.1 ^{±0.8}	84.6 ^{±0.7}
NorwAI-Llama2-7B	71.9	39.8	67.0	65.2	65.2	76.2 ^{±0.8}	73.2 ^{±0.8}	78.7 ^{±0.8}	83.8 ^{±0.7}	80.1 ^{±0.7}
NorMistral-7B-warm	74.8	55.6	67.2	67.4	67.6	84.3 ^{±0.7}	73.9 ^{±0.8}	84.3 ^{±0.7}	75.7 ^{±0.8}	73.4 ^{±0.8}
NorGPT-3B	67.7	52.4	67.0	67.7	67.0	58.1 ^{±0.9}	54.0 ^{±0.9}	55.5 ^{±0.9}	54.8 ^{±0.9}	55.0 ^{±0.9}
Viking-7B	75.3	56.6	68.3	65.9	67.0	84.5 ^{±0.7}	78.4 ^{±0.8}	73.9 ^{±0.8}	74.5 ^{±0.8}	73.4 ^{±0.8}
Viking-13B	69.0	66.8	67.3	68.3	65.0	83.2 ^{±0.7}	72.5 ^{±0.8}	89.2 ^{±0.6}	84.5 ^{±0.7}	83.2 ^{±0.7}
Mistral-Nemo-12B	78.5	68.0	67.0	67.1	67.0	91.2^{±0.5}	89.8 ^{±0.6}	90.5^{±0.5}	89.8 ^{±0.6}	89.3^{±0.6}

Table 9: **Complete results on document-level sentiment analysis with NoReC** We show the detailed results for each evaluated model, few-shot setting and prompt template. As the few-shot demonstrations are sampled randomly, we repeat them five times and show the mean accuracy as well as the standard deviation (rendered as superscript). The best results for each column are boldfaced, the overall best result for each few-shot setting is highlighted in blue.

B.4 NorCommonsenseQA

Accurately predicting the correct answers on this datasets requires different types of commonsense knowledge. The creating of the Norwegian NorCommonsenseQA has been inspired by the English CommonsenseQA dataset (Talmor et al., 2019). The data can be found at <https://huggingface.co/datasets/litg/norcommonsenseqa>.

Inference setup The model is given a test example formatted according to a prompt template and ranks the answer candidates based on their probabilities. The most probable answer candidate is selected as the resulting answer.

Performance metric There are five possible answers for each question. We measure the performance with a simple accuracy.

Prompt templates We used the following five prompt templates from NorEval. The templates are adapted to the Bokmål and Nynorsk versions of this dataset.

Prompt A (Bokmål and Nynorsk):

```

1   Spørsmål: {$question}
2
3   Svar: {$prediction:{$answer_1}/{ $answer_2}/{ $answer_3}/{ $answer_4}/{ $answer_5}}
```

Prompt B (Bokmål):

```

1   {$question}
2   Hvilket av følgende mulige svar er det riktige?
3   A: {$answer_1}
4   B: {$answer_2}
5   C: {$answer_3}
6   D: {$answer_4}
7   E: {$answer_5}
8   Svar: {$prediction:A/B/C/D/E}
```

Prompt B (Nynorsk):

```
1  {$question}
2  Kva av følgende moglege svar er det rette?
3  A: {$answer_1}
4  B: {$answer_2}
5  C: {$answer_3}
6  D: {$answer_4}
7  E: {$answer_5}
8  Svar: {$prediction:A/B/C/D/E}
```

Prompt C (Bokmål):

```
1  Gitt alternativene under, hva er svaret på følgende spørsmål: {$question}
2
3  Alternativer:
4  - {$answer_1}
5  - {$answer_2}
6  - {$answer_3}
7  - {$answer_4}
8  - {$answer_5}
9
10 Svar: {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}/{$answer_5}}
```

Prompt C (Nynorsk):

```
1  Gitt alternativa under, kva er svaret på følgende spørsmål: {$question}
2
3  Alternativ:
4  - {$answer_1}
5  - {$answer_2}
6  - {$answer_3}
7  - {$answer_4}
8  - {$answer_5}
9
10 Svar: {$prediction:A/B/C/D/E}
```

Prompt D (Bokmål):

```
1  {$question}
2  Velg riktig svar blant disse alternativene:
3  - {$answer_1}
4  - {$answer_2}
5  - {$answer_3}
6  - {$answer_4}
7  - {$answer_5}
8
9  Svar: {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}/{$answer_5}}
```

Prompt D (Nynorsk):

```
1  {$question}
2  Vel rett svar blant desse alternativa:
3  - {$answer_1}
4  - {$answer_2}
5  - {$answer_3}
6  - {$answer_4}
7  - {$answer_5}
8
9  Svar: {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}/{$answer_5}}
```

Prompt E (Bokmål):

```

1  {$question}
2  A: {$answer_1}
3  B: {$answer_2}
4  C: {$answer_3}
5  D: {$answer_4}
6  E: {$answer_5}
7
8  Er det riktige svaret A, B, C, D, eller E?
9
10 Svar: {$prediction:A/B/C/D/E}

```

Prompt E (Nynorsk):

```

1  {$question}
2  A: {$answer_1}
3  B: {$answer_2}
4  C: {$answer_3}
5  D: {$answer_4}
6  E: {$answer_5}
7
8  Er det rette svaret A, B, C, D, eller E?
9
10 Svar: {$prediction:A/B/C/D/E}

```

Full results The complete evaluation results on NorCommonsenseQA (Bokmål and Nynorsk) are provided in Table 10. For reference, the random-guessing baseline achieves 20% on this task (for both language variants).

Prompt template	Bokmål (0-shot)					Nynorsk (0-shot)				
	A	B	C	D	E	A	B	C	D	E
NorMistral-11B	61.0	56.9	23.1	51.8	45.3	44.2	51.6	36.8	46.3	30.5
NorwAI-Mistral-7B	30.8	49.7	20.3	22.3	28.5	43.2	20.0	23.2	27.4	23.2
NorwAI-Llama2-7B	37.2	54.2	23.2	25.8	33.6	37.9	18.9	17.9	27.4	18.9
NorMistral-7B-warm	30.4	51.3	20.5	21.4	29.2	43.2	18.9	15.8	30.5	20.0
NorGPT-3B	26.4	34.7	22.1	20.1	23.5	29.5	20.0	16.8	25.3	25.3
Viking-7B	26.1	44.9	19.1	20.5	23.2	38.9	21.1	25.3	23.2	23.2
Viking-13B	24.7	51.1	18.1	19.1	24.0	40.0	13.7	24.2	20.0	16.8
Mistral-Nemo-12B	43.4	44.1	43.7	38.9	31.7	33.7	33.7	25.3	27.4	25.3

Table 10: **Complete results on commonsense reasoning evaluated on NorCommonsenseQA (Bokmål and Nynorsk)** We show the detailed results for each evaluated model and prompt template. The best results for each column are boldfaced, the overall best result is highlighted in blue.

B.5 NRK-Quiz-QA

This question-answering dataset focuses on knowledge about Norway and its culture. The data can be found at https://huggingface.co/datasets/litg/nrk_quiz_qa.

Inference setup The model is given a test example formatted according to a **Prompt template** and ranks the answer candidates based on their probabilities. The most probable answer candidate is selected as the resulting answer.

Performance metric There is a limited number of possible answers for each question. We measure the performance with a simple accuracy.

Prompt templates We used the following five prompt templates from NorEval for testing all language models on question answering with NRK-Quiz-QA. Note that the examples in this dataset have a variable number of answer options, we show the prompt templates for four options as an example. The templates are adapted to the Bokmål and Nynorsk versions of this dataset.

Prompt A (Bokmål and Nynorsk):

```
1   Spørsmål: {$question}
2
3   Svar: {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}}
```

Prompt B (Bokmål):

```
1   {$question}
2
3   Svaralternativer:
4   - {$answer_1}
5   - {$answer_2}
6   - {$answer_3}
7   - {$answer_4}
8
9   Hva er riktig svar?
10
11  Svar: {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}}
```

Prompt B (Nynorsk):

```
1   {$question}
2   {$question}
3
4   Svaralternativer:
5   - {$answer_1}
6   - {$answer_2}
7   - {$answer_3}
8   - {$answer_4}
9
10  Kva er rett svar?
11
12  Svar: {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}}
```

Prompt C (Bokmål):

```
1   {$question}
2   A: {$answer_1}
3   B: {$answer_2}
4   C: {$answer_3}
5   D: {$answer_4}
6
7   Er det riktige svaret A, B, C, eller D?
8
9   Svar: {$prediction:A/B/C/D}
```

Prompt C (Nynorsk):

```
1   {$question}
2   A: {$answer_1}
3   B: {$answer_2}
4   C: {$answer_3}
5   D: {$answer_4}
6
7   Er det rette svare A, B, C, eller D?
```

```

8
9 Svar: {$prediction:A/B/C/D}

```

Prompt D (Bokmål and Nynorsk):

```

1 Spørsmål: {$question}
2 A: {$answer_1}
3 B: {$answer_2}
4 C: {$answer_3}
5 D: {$answer_4}
6
7 Svar: {$prediction:A/B/C/D}

```

Prompt E (Bokmål):

```

1 {$question}
2 Velg riktig svar blant disse alternativene:
3 - {$answer_1}
4 - {$answer_2}
5 - {$answer_3}
6 - {$answer_4}
7
8 Svar: {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}}

```

Prompt E (Nynorsk):

```

1 {$question}
2 Vel rett svar blant desse alternativa:
3 - {$answer_1}
4 - {$answer_2}
5 - {$answer_3}
6 - {$answer_4}
7
8 Svar: {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}}

```

Full results The complete evaluation results on NRK-Quiz-QA (Bokmål and Nynorsk) are in Table 11. The random-guessing baseline achieves 28% accuracy on the Bokmål version of this task and 27% on the Nynorsk version.

Prompt template	Bokmål (0-shot)					Nynorsk (0-shot)				
	A	B	C	D	E	A	B	C	D	E
NorMistral-11B	63.7	50.5	38.6	41.1	50.6	71.9	56.5	46.4	41.9	57.1
NorwAI-Mistral-7B	55.2	43.4	34.6	34.8	46.6	65.2	50.6	35.8	35.6	53.2
NorwAI-Llama2-7B	52.3	39.2	26.0	30.1	40.3	64.3	44.1	25.3	31.8	44.1
NorMistral-7B-warm	57.9	39.8	27.7	32.5	40.7	65.9	41.3	28.8	32.7	41.1
NorGPT-3B	33.1	28.2	26.3	26.1	27.9	37.3	29.6	25.0	24.7	30.5
Viking-7B	44.3	29.9	26.1	28.8	31.9	51.1	31.2	26.8	30.8	34.7
Viking-13B	51.0	31.8	27.8	30.2	31.6	54.8	34.5	28.0	30.2	31.9
Mistral-Nemo-12B	47.0	46.1	41.8	47.4	46.6	47.2	43.6	41.4	45.7	42.8

Table 11: **Complete results on Norwegian-specific and world knowledge evaluated on NRK-Quiz-QA (Bokmål and Nynorsk)** We show the detailed results for each evaluated model and prompt template. The best results for each column are boldfaced, the overall best result is highlighted in blue.

B.6 NorOpenBookQA

Inspired by the English OpenBookQA (Mihaylov et al., 2018), this task follows the open book exams for testing human understanding of a subject. Correctly answering a question should require multi-step reasoning, common and commonsense knowledge, and rich text comprehension. The data can be found at <https://huggingface.co/datasets/lgt/noropenbookqa>.

Inference setup The model is given a test example formatted according to a prompt template and ranks the answer candidates based on their probabilities. The most probable answer candidate is selected as the resulting answer.

Performance metric There are four possible for answers for each passage-question pair. We measure the performance with a simple accuracy.

Prompt templates We used the following five prompt templates from NorEval for testing all language models on question answering with NorOpenBookQA: The templates are adapted to the Bokmål and Nynorsk versions of this dataset.

Prompt A (Bokmål and Nynorsk):

```
1  {$fact}
2  {$question} {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}}
```

Prompt B (Bokmål):

```
1  Faktatekst: {$fact}
2  Spørsmål til teksten: {$question}
3
4  Svaralternativer:
5  - {$answer_1}
6  - {$answer_2}
7  - {$answer_3}
8  - {$answer_4}
9
10 Hva er riktig svar? {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}}
```

Prompt B (Nynorsk):

```
1  Faktatekst: {$fact}
2  Spørsmål til teksten: {$question}
3
4  Svaralternativer:
5  - {$answer_1}
6  - {$answer_2}
7  - {$answer_3}
8  - {$answer_4}
9
10 Kva er rett svar? {$prediction:{$answer_1}/{$answer_2}/{$answer_3}/{$answer_4}}
```

Prompt C (Bokmål):

```
1  {$fact}
2  {$question}
3  A: {$answer_1}
4  B: {$answer_2}
5  C: {$answer_3}
6  D: {$answer_4}
7
8  Er det riktige svaret A, B, C, eller D?
9
10 Svar: {$prediction:A/B/C/D}
```

Prompt C (Nynorsk):

```
1  {{$fact}}
2  {{$question}}
3  A: {{$answer_1}}
4  B: {{$answer_2}}
5  C: {{$answer_3}}
6  D: {{$answer_4}}
7
8  Er det rette svare A, B, C, eller D?
9
10 Svar: {{$prediction:A/B/C/D}}
```

Prompt D (Bokmål and Nynorsk):

```
1  Bakgrunn: {{$fact}}
2
3  Spørsmål: {{$question}}
4  A: {{$answer_1}}
5  B: {{$answer_2}}
6  C: {{$answer_3}}
7  D: {{$answer_4}}
8
9  Svar: {{$prediction:A/B/C/D}}
```

Prompt E (Bokmål):

```
1  Ta utgangspunkt i følgende fakta når du svarer på spørsmålet: {{$fact}}
2
3  {{$question}}
4  Velg riktig svar blant disse alternativene:
5  - {{$answer_1}}
6  - {{$answer_2}}
7  - {{$answer_3}}
8  - {{$answer_4}}
9
10 Svar: {{$prediction:{{$answer_1}}/{{$answer_2}}/{{$answer_3}}/{{$answer_4}}}}
```

Prompt E (Nynorsk):

```
1  Ta utgangspunkt i følgende fakta når du svarar på spørsmålet: {{$fact}}
2
3  {{$question}}
4  Vel rett svar blant desse alternativa:
5  - {{$answer_1}}
6  - {{$answer_2}}
7  - {{$answer_3}}
8  - {{$answer_4}}
9
10 Svar: {{$prediction:{{$answer_1}}/{{$answer_2}}/{{$answer_3}}/{{$answer_4}}}}
```

Full results The complete evaluation on NorOpenBookQA (Bokmål and Nynorsk) is in [Table 12](#). Note that randomly guessing the answers achieves 25% on this task.

B.7 Summarization (NorSumm)

NorSumm by [Touileb et al. \(2025\)](#) is a benchmark for abstractive summarization of Norwegian news articles. It offers another perspective on the level of Norwegian language understanding of different language models. An important feature of this dataset is that its Bokmål and Nynorsk variants are parallel.

BOKMÅL

Prompt template	0-shot					1-shot					16-shot				
	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E
NorMistral-11B	44.6	55.7	54.0	67.8	65.8	45.6 ^{±2.9}	74.5 ^{±2.5}	68.8 ^{±2.7}	76.2 ^{±2.5}	70.1 ^{±2.7}	51.3^{±2.9}	75.8 ^{±2.5}	77.9 ^{±2.4}	75.8 ^{±2.5}	76.5 ^{±2.5}
NorwAI-Mistral-7B	47.7	35.6	34.6	35.9	43.3	46.3 ^{±2.9}	51.7 ^{±2.9}	34.9 ^{±2.8}	35.2 ^{±2.8}	50.0 ^{±2.9}	50.0 ^{±2.9}	52.3 ^{±2.9}	40.9 ^{±2.9}	44.3 ^{±2.9}	51.7 ^{±2.9}
NorwAI-Llama2-7B	45.6	32.6	25.8	32.6	43.3	45.0 ^{±2.9}	51.7 ^{±2.9}	33.2 ^{±2.7}	37.6 ^{±2.8}	46.3 ^{±2.9}	45.0 ^{±2.9}	52.3 ^{±2.9}	32.2 ^{±2.7}	34.2 ^{±2.8}	49.7 ^{±2.9}
NorMistral-7B-warm	46.6	35.6	28.9	32.6	44.3	46.6^{±2.9}	51.7 ^{±2.9}	32.9 ^{±2.7}	34.2 ^{±2.8}	46.6 ^{±2.9}	48.3 ^{±2.9}	49.0 ^{±2.9}	42.3 ^{±2.9}	40.3 ^{±2.8}	45.0 ^{±2.9}
NorGPT-3B	32.6	31.2	24.2	22.5	33.2	28.5 ^{±2.6}	28.2 ^{±2.6}	23.8 ^{±2.5}	24.2 ^{±2.5}	28.5 ^{±2.6}	29.5 ^{±2.6}	27.9 ^{±2.6}	26.5 ^{±2.6}	26.8 ^{±2.6}	28.2 ^{±2.6}
Viking-7B	41.9	26.5	20.8	28.5	27.9	45.3 ^{±2.9}	25.8 ^{±2.5}	26.8 ^{±2.6}	24.5 ^{±2.5}	30.9 ^{±2.7}	48.7 ^{±2.9}	26.5 ^{±2.6}	28.2 ^{±2.6}	23.8 ^{±2.5}	31.9 ^{±2.7}
Viking-13B	44.6	27.5	21.1	25.5	31.9	45.6 ^{±2.9}	33.2 ^{±2.7}	25.2 ^{±2.5}	27.2 ^{±2.6}	38.6 ^{±2.8}	47.0 ^{±2.9}	38.9 ^{±2.8}	29.9 ^{±2.7}	26.2 ^{±2.6}	36.9 ^{±2.8}
Mistral-Nemo-12B	43.6	60.7	58.1	71.5	68.5	44.0 ^{±2.9}	82.6^{±2.2}	82.2^{±2.2}	82.9^{±2.2}	76.2^{±2.5}	49.7 ^{±2.9}	82.9^{±2.2}	82.2^{±2.2}	85.9^{±2.0}	80.9^{±2.3}

NYNORSK

Prompt template	0-shot					1-shot					16-shot				
	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E
NorMistral-11B	33.3	56.7	56.7	56.7	65.6	28.9 ^{±4.8}	70.0 ^{±4.9}	68.9 ^{±4.9}	71.1 ^{±4.8}	68.9 ^{±4.9}	40.0^{±5.2}	72.2 ^{±4.7}	76.7 ^{±4.5}	77.8 ^{±4.4}	77.8 ^{±4.4}
NorwAI-Mistral-7B	30.0	37.8	32.2	27.8	38.9	31.1 ^{±4.9}	34.4 ^{±5.0}	28.9 ^{±4.8}	27.8 ^{±4.7}	36.7 ^{±5.1}	37.8 ^{±5.1}	45.6 ^{±5.3}	41.1 ^{±5.2}	44.4 ^{±5.3}	42.2 ^{±5.2}
NorwAI-Llama2-7B	25.6	30.0	32.2	27.8	28.9	28.9 ^{±4.8}	37.8 ^{±5.1}	32.2 ^{±5.0}	32.2 ^{±5.0}	36.7 ^{±5.1}	27.8 ^{±4.7}	37.8 ^{±5.1}	32.2 ^{±5.0}	32.2 ^{±5.0}	38.9 ^{±5.2}
NorMistral-7B-warm	26.7	28.9	34.4	40.0	36.7	27.8 ^{±4.7}	32.2 ^{±5.0}	40.0 ^{±5.2}	38.9 ^{±5.2}	43.3 ^{±5.3}	30.0 ^{±4.9}	38.9 ^{±5.2}	41.1 ^{±5.2}	40.0 ^{±5.2}	41.1 ^{±5.2}
NorGPT-3B	20.0	27.8	34.4	30.0	25.6	20.0 ^{±4.2}	25.6 ^{±4.6}	24.4 ^{±4.6}	23.3 ^{±4.5}	20.0 ^{±4.2}	18.9 ^{±4.1}	23.3 ^{±4.5}	34.4 ^{±5.0}	26.7 ^{±4.7}	23.3 ^{±4.5}
Viking-7B	22.2	20.0	18.9	27.8	15.6	30.0 ^{±4.9}	22.2 ^{±4.4}	32.2 ^{±5.0}	25.6 ^{±4.6}	27.8 ^{±4.7}	27.8 ^{±4.7}	22.2 ^{±4.4}	27.8 ^{±4.7}	23.3 ^{±4.5}	27.8 ^{±4.7}
Viking-13B	30.0	34.4	17.8	23.3	31.1	32.2 ^{±5.0}	27.8 ^{±4.7}	17.8 ^{±4.1}	33.3 ^{±5.0}	33.3 ^{±5.0}	36.7 ^{±5.1}	26.7 ^{±4.7}	22.2 ^{±4.4}	18.9 ^{±4.1}	26.7 ^{±4.7}
Mistral-Nemo-12B	30.0	52.2	52.2	58.9	60.0	35.6^{±5.1}	71.1^{±4.8}	77.8^{±4.4}	78.9^{±4.3}	71.1^{±4.8}	33.3 ^{±5.0}	82.2^{±4.1}	82.2^{±4.1}	86.7^{±3.6}	81.1^{±4.1}

Table 12: **Complete results on world knowledge evaluated on NorOpenBookQA (Bokmål and Nynorsk)** We show the detailed results for each evaluated model, few-shot setting and prompt template. The best results for each column are boldfaced, the overall best result for each few-shot setting is highlighted in blue.

Inference setup The model is given a test example formatted according to a **Prompt template** and generates the answer via a greedy search decoding strategy.

Performance metric We use Rouge-L as the standard metric for summarization (Lin, 2004). ROUGE-L uses longest common subsequence matching, allowing it to identify matching content even when ordered differently in generated and reference summaries.

Prompt templates We used the following six prompt templates from NorEval for testing all language models on summarization with NorSumm. The templates are adapted to the Bokmål and Nynorsk versions of this dataset.

Prompt A (Bokmål):

- 1 Skriv en oppsummering av følgende artikkel med kun noen få punkter: {\$article}
- 2 Oppsummering: {\$prediction}

Prompt A (Nynorsk):

- 1 Skriv ei oppsummering av følgande artikkel med berre nokre få punkt: {\$article}
- 2 Oppsummering: {\$prediction}

Prompt B (Bokmål):

- 1 Oppsummer følgende artikkel med noen få setninger: `{ $article }`
- 2 Oppsummering: `{ $prediction }`

Prompt B (Nynorsk):

- 1 Oppsummer følgande artikkel med nokre få setningar: `{ $article }`
- 2 Oppsummering: `{ $prediction }`

Prompt C (Bokmål):

- 1 `{ $article }`
- 2 Skriv en kort og presis oppsummering av teksten over. Språket må være klart og lett å forstå.
 - Sørg for å ikke introdusere feil. Oppsummeringen må dekke følgende spørsmål: hvem, hva,
 - hvor, når, og hvorfor er denne saken viktig å vite om. Oppsummeringen må være engasjerende
 - og fremheve nøkkelinformasjon fra artikkelen. Oppsummeringen skal inneholde maksimalt 700
 - tegn, inkludert mellomrom. `{ $prediction }`

Prompt C (Nynorsk):

- 1 `{ $article }`
- 2 Skriv ein kort og presis oppsummering av teksten over. Språket må vere klart og lett å forstå.
 - Sørg for å ikkje introdusere feil. Oppsummeringa må dekkje følgande spørsmål: kven, kva,
 - kor, når, og kvifor er denne saka viktig å vite om. Oppsummeringa må vere engasjerande og
 - framheve nøkkelinformasjon frå artikkelen. Oppsummeringa skal innehalde maksimalt 700 tegn,
 - inkludert mellomrom. `{ $prediction }`

Prompt D (Bokmål):

- 1 Gi et kortfattet sammendrag av følgende tekst: `{ $article }` `{ $prediction }`

Prompt D (Nynorsk):

- 1 Gje eit kortfatta samandrag av følgande tekst: `{ $article }` `{ $prediction }`

Prompt E (Bokmål):

- 1 Lag en kort oppsummering som sammenfatter den følgende teksten i noen få punkter:
- 2 `{ $article }`
- 3
- 4 Oppsummering: `{ $prediction }`

Prompt E (Nynorsk):

- 1 Lag ein kort oppsummering som samanfattar den følgande teksten i nokre få punkt:
- 2 `{ $article }`
- 3
- 4 Oppsummering: `{ $prediction }`

Prompt F (Bokmål):

- 1 Hele artikkelen:
- 2 `{ $article }`
- 3
- 4 Hovedpunkter: `{ $prediction }`

Prompt F (Nynorsk):

- 1 Heile artikkelen:
- 2 `{ $article }`
- 3
- 4 Hovudpunkt: `{ $prediction }`

Prompt template	Bokmål (0-shot)						Nynorsk (0-shot)					
	A	B	C	D	E	F	A	B	C	D	E	F
NorMistral-11B	17.6	20.5	34.9	45.0	42.5	5.7	15.4	17.9	25.4	32.4	32.6	6.8
NorwAI-Mistral-7B	12.2	0.0	0.0	2.6	0.0	0.0	10.3	0.0	0.0	3.5	0.0	0.0
NorwAI-Llama2-7B	10.7	0.0	0.0	7.8	3.5	0.0	10.4	0.0	0.0	4.4	5.9	0.0
NorMistral-7B-warm	7.8	0.0	0.0	16.5	9.0	0.0	8.6	0.0	0.0	6.9	4.1	0.0
NorGPT-3B	8.8	8.5	31.6	33.8	25.2	2.8	7.4	10.9	21.6	24.3	20.0	4.0
Viking-7B	11.2	5.5	16.5	29.8	31.9	0.0	10.5	3.0	16.4	25.7	24.2	0.0
Viking-13B	11.1	1.7	6.0	23.7	36.3	0.0	9.8	0.0	4.3	19.6	28.8	0.4
Mistral-Nemo-12B	13.4	26.4	41.5	35.6	44.9	2.9	12.4	18.2	30.0	30.3	30.9	3.6

Table 13: **Complete results on NorSumm summarization (Bokmål and Nynorsk versions)** We show the detailed results for each evaluated model and prompt template. The best results for each column are boldfaced, the overall best result is highlighted in blue.

Full results The complete evaluation on NorSumm (both Bokmål and Nynorsk variants) is provided in Table 13.

B.8 Grammatical error correction (ASK-GEC)

This task tests how do language models understand more low-level features of the Norwegian language. We use the ASK-GEC dataset from Jentoft (2023) that is based on corrected essays of Norwegian language learners.

Inference setup The model is given a test example formatted according to a prompt template; given this input, it then generates the answer via a greedy-search decoding strategy.

Performance metric We use the $F_{0.5}$ -score to measure the amount of successfully fixed correction-spans. These spans are heuristically identified by the ERRANT system (Bryant et al., 2017). More details about using this metric for Norwegian grammatical error corrections can be found in Jentoft (2023).

Prompt templates We used the following five prompt templates for grammatical error correction:

Prompt A:

```
1 Tekst: {$text}
2 Korreksjon: {$prediction}
```

Prompt B:

```
1 Tekst: {$text}
2 Rettet versjon: {$prediction}
```

Prompt C:

```
1 Skriv om følgende tekst slik at den blir grammatisk korrekt: {$text}
2 Korreksjon: {$prediction}
```

Prompt D:

```
1 Original versjon: {$text}
2 Korrekturlest og rettet versjon: {$prediction}
```

Prompt E:

```
1 Rett opp grammatiske feil i denne teksten: {$text}
2 Korreksjon: {$prediction}
```

Prompt template	0-shot					1-shot					16-shot				
	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E
NorMistral-11B	2.8	16.6	39.9	26.8	38.8	38.6	41.8	43.3	45.2	44.7	52.6	52.3	50.4	51.5	51.4
NorwAI-Mistral-7B	0.0	0.0	22.2	0.0	0.0	37.2	39.1	45.3	42.8	46.1	51.9	52.4	52.5	53.2	52.7
NorwAI-Llama2-7B	0.0	0.2	13.3	0.0	27.1	38.5	40.5	46.2	44.2	45.0	51.1	51.3	51.2	51.4	51.1
NorMistral-7B-warm	2.1	0.0	11.1	33.6	18.7	34.4	38.0	42.8	41.2	41.1	48.0	48.2	48.7	48.2	48.5
NorGPT-3B	0.2	0.0	0.2	1.4	0.3	0.4	0.4	0.4	0.4	0.4	0.9	1.3	1.8	1.2	1.5
Viking-7B	2.8	1.2	23.1	0.0	11.4	29.9	37.0	40.6	39.9	38.9	50.7	51.0	50.4	51.2	50.1
Viking-13B	3.1	0.0	37.8	25.1	34.8	42.6	43.5	45.7	44.8	46.0	52.4	52.0	51.9	52.4	51.8
Mistral-Nemo-12B	14.7	18.6	36.5	16.9	12.3	38.8	36.8	37.5	38.6	39.6	43.9	43.7	42.7	43.7	43.1

Table 14: **Complete results on grammatical error correction** We show the detailed results for each evaluated model, few-shot setting and prompt template. The best results for each column are boldfaced, the overall best result for each few-shot setting is highlighted in blue.

Full results The complete evaluation on ASK-GEC is provided in Table 15.

B.9 Language identification (SLIDE)

We use the Scandinavian language identification and evaluation (SLIDE) from <https://github.com/lgtoslo/slide>. This dataset consists of sentences manually annotated with the language they are written in: Norwegian Bokmål, Nynorsk, Danish or Swedish (we filtered out the examples that are not written in any Scandinavian language). The sentences can be annotated with multiple language labels if applicable.

Inference setup This task is solved as classification – the label with the highest probability given the prompt (estimated by the evaluated language model) is chosen as the predicted label. The few-shot demonstrations are randomly sampled from the SLIDE validation set.

Performance metric We test whether a language model is able to correctly predict one of the (potentially multiple) languages a sentence is written in. We thus adopt the *loose accuracy* metric from SLIDE, where a single-label prediction is considered to be correct if is in the set of gold language labels.

Prompt templates We used the following five prompt templates for testing all language models on grammatical error correction. The few-shot demonstrations are separated by double newlines \n\n.

Prompt A:

```
1 Tekst: {$text}
2 Korreksjon: {$prediction}
```

Prompt B:

```
1 Tekst: {$text}
2 Rettet versjon: {$prediction}
```

Prompt C:

```
1 Skriv om følgende tekst slik at den blir grammatisk korrekt: {$text}
2 Korreksjon: {$prediction}
```

Prompt D:

```
1 Original versjon: {$text}
2 Korrekturlest og rettet versjon: {$prediction}
```

Prompt E:

```
1 Rett opp grammatiske feil i denne teksten:  $\{ \$text \}$ 
2 Korreksjon:  $\{ \$prediction \}$ 
```

Full results The complete evaluation on language identification is given in ?? . Note that the majority baseline on this task is 40.3% loose accuracy and the random baseline is 28.2%.

Prompt template	0-shot					1-shot					16-shot				
	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E
NorMistral-11B	74.0	41.5	55.7	62.6	53.1	78.8	60.8± 0.7	80.6± 0.2	80.6± 0.4	58.7	97.9	95.6± 0.2	94.6	97.0± 0.3	98.2± 0.1
NorwAI-Mistral-7B	65.8	54.0	38.8	69.6	66.9	74.8	39.2	47.9	76.9	64.5	95.1	69.9	90.4	92.0	95.7
NorwAI-Llama2-7B	69.8	37.4	42.9	49.4	59.0	65.4	33.0	40.8	54.5	43.1	93.5	74.1	67.0	77.4	87.5
NorMistral-7B-warm	87.5	47.7	42.2	65.4	61.6	85.7± 0.4	40.3	63.5	72.2	73.2± 0.4	98.1± 0.1	92.2	96.2± 0.3	92.1	97.3
NorGPT-3B	36.6	24.0	49.9	43.9	42.6	37.7	35.6	32.4	32.4	32.4	39.0	27.9	40.0	40.3	40.2
Viking-7B	74.4	42.7	41.0	34.7	32.8	46.2	37.1	35.2	39.5	36.0	77.2	47.4	44.3	58.5	52.6
Viking-13B	71.5	59.5	41.0	41.9	32.4	55.1	36.4	37.8	43.8	34.1	84.4	62.0	56.8	79.1	65.3
Mistral-Nemo-12B	68.3	41.7	50.3	48.5	40.7	63.8	56.0	74.3	58.6	45.8	85.9	84.6	86.6	87.3	86.1

Table 15: **Complete results on Scandinavian language identification** We show the detailed results for each evaluated model, few-shot setting and prompt template. As the few-shot demonstrations are sampled randomly, we repeat them five times and show the mean accuracy as well as the standard deviation (rendered as superscript). The best results for each column are boldfaced, the overall best result for each few-shot setting is highlighted in blue.

B.10 Translation

Inference setup This task is solved as generation via prefix prompting – the model is given a prompt without the $\$prediction$ suffix and then it autoregressively generates a prediction until outputting a newline. We use simple greedy search to generate the output.

Performance metric We measure the translation quality with SacreBLEU scores (Post, 2018) with signature BLEU+case.mixed+numrefs.1+smooth.exp+tok.intl+version.1.2.20 as the main metric. We also provide chrF++ scores as an additional metric (Popović, 2017).

B.10.1 English to Bokmål translation

Prompt templates We used the following four prompt templates for testing all language models on translation to Northern Sámi.

Prompt A:

```
1 Engelsk:  $\{ \$text \}$ 
2 Bokmål:  $\{ \$prediction \}$ 
```

Prompt B:

```
1 Oversett følgende setning til Bokmål:  $\{ \$text \}$ 
2 Bokmål:  $\{ \$prediction \}$ 
```

Prompt C:

```
1 Gi en oversettelse til Bokmål for denne setningen:  $\{ \$text \}$ 
2 Bokmål:  $\{ \$prediction \}$ 
```

Prompt D:

```
1 Hva blir " $\{ \$text \}$ " på Bokmål?
2 Bokmål:  $\{ \$prediction \}$ 
```

Full results The complete evaluation on translation to Bokmål is given in Table 18 (BLEU scores in the top sub-table and chrF++ scores below).

BLEU SCORES

Prompt template	0-shot				1-shot				16-shot			
	A	B	C	D	A	B	C	D	A	B	C	D
NorMistral-11B	54.9	51.6	43.1	44.5	58.2 \pm 0.6	58.5\pm0.6	58.2 \pm 0.6	58.1\pm0.6	58.6 \pm 0.6	58.8 \pm 0.6	58.3 \pm 0.6	58.5 \pm 0.6
NorwAI-Mistral-7B	56.1	31.0	52.8	50.9	58.2 \pm 0.6	58.1 \pm 0.6	58.1 \pm 0.6	57.7 \pm 0.7	58.4 \pm 0.6	58.7 \pm 0.6	58.5 \pm 0.6	58.2 \pm 0.6
NorwAI-Llama2-7B	35.5	23.9	23.6	44.8	55.8 \pm 0.7	57.0 \pm 0.6	57.3 \pm 0.6	56.1 \pm 0.7	57.8 \pm 0.6	57.9 \pm 0.6	57.8 \pm 0.6	57.5 \pm 0.6
NorMistral-7B-warm	54.7	53.1	0.0	51.6	55.9 \pm 0.7	56.5 \pm 0.6	56.6 \pm 0.6	56.2 \pm 0.7	57.0 \pm 0.7	57.0 \pm 0.6	57.2 \pm 0.7	56.4 \pm 0.6
NorGPT-3B	0.1	0.1	0.1	0.2	0.0 \pm 0.0	0.2 \pm 0.0	0.1 \pm 0.0	0.4 \pm 0.0	0.2 \pm 0.0	1.8 \pm 0.1	0.7 \pm 0.0	0.9 \pm 0.0
Viking-7B	53.4	35.0	42.1	1.4	54.2 \pm 0.6	58.3 \pm 0.7	57.2 \pm 0.7	56.7 \pm 0.7	58.7 \pm 0.7	59.6 \pm 0.6	59.7 \pm 0.6	59.4 \pm 0.7
Viking-13B	24.2	58.2	14.9	1.9	58.6\pm0.7	58.3 \pm 0.7	58.7\pm0.8	57.5 \pm 0.7	59.5\pm0.6	60.0\pm0.6	59.9\pm0.6	59.9\pm0.7
Mistral-Nemo-12B	44.3	46.1	44.3	45.5	48.5 \pm 0.6	48.9 \pm 0.6	48.9 \pm 0.6	48.9 \pm 0.6	49.3 \pm 0.6	49.5 \pm 0.6	49.2 \pm 0.6	49.5 \pm 0.6

CHRF++ SCORES

Prompt template	0-shot				1-shot				16-shot			
	A	B	C	D	A	B	C	D	A	B	C	D
NorMistral-11B	71.7	71.4	67.2	69.3	73.6\pm0.4	73.8 \pm 0.4	73.8\pm0.4	73.4\pm0.4	73.8 \pm 0.4	74.0 \pm 0.4	73.8 \pm 0.4	73.7 \pm 0.4
NorwAI-Mistral-7B	71.1	55.5	68.8	70.8	73.0 \pm 0.4	73.2 \pm 0.4	73.2 \pm 0.4	72.6 \pm 0.5	73.4 \pm 0.4	73.7 \pm 0.4	73.6 \pm 0.4	73.4 \pm 0.4
NorwAI-Llama2-7B	59.3	40.8	39.9	64.1	71.3 \pm 0.4	72.1 \pm 0.4	72.4 \pm 0.4	71.2 \pm 0.5	72.7 \pm 0.4	72.9 \pm 0.4	72.9 \pm 0.4	72.6 \pm 0.4
NorMistral-7B-warm	69.6	67.5	9.6	66.3	70.9 \pm 0.4	71.6 \pm 0.4	71.6 \pm 0.4	71.3 \pm 0.4	72.0 \pm 0.4	72.3 \pm 0.4	72.2 \pm 0.4	72.0 \pm 0.4
NorGPT-3B	9.8	8.1	9.7	9.5	4.4 \pm 0.1	4.8 \pm 0.1	4.4 \pm 0.1	6.4 \pm 0.1	4.4 \pm 0.1	12.3 \pm 0.3	7.9 \pm 0.1	8.9 \pm 0.1
Viking-7B	70.8	51.8	63.6	12.5	70.6 \pm 0.4	72.9 \pm 0.7	72.3 \pm 0.5	71.9 \pm 0.5	73.3 \pm 0.5	74.3 \pm 0.4	74.4 \pm 0.4	74.1 \pm 0.5
Viking-13B	59.4	72.8	46.8	15.5	73.3 \pm 0.5	74.0\pm0.4	73.6 \pm 0.5	72.6 \pm 0.5	74.2\pm0.4	74.6\pm0.4	74.5\pm0.4	74.5\pm0.4
Mistral-Nemo-12B	61.4	64.2	61.7	63.4	65.8 \pm 0.5	66.5 \pm 0.4	66.4 \pm 0.4	66.6 \pm 0.4	66.8 \pm 0.4	67.0 \pm 0.4	66.9 \pm 0.4	67.0 \pm 0.4

Table 16: **Complete results on translation from English to Norwegian Bokmål** We show the detailed results for each evaluated model, few-shot setting and prompt template. As the few-shot demonstrations are sampled randomly, we repeat them five times and show the mean accuracy as well as the standard deviation (rendered as superscript). The best results for each column are boldfaced, the overall best result for each few-shot setting is highlighted in blue.

B.10.2 English to Nynorsk translation

Prompt templates We used the following five prompt templates for testing all language models on translation to Nynorsk.

Prompt A:

```
1 Engelsk: {$text}
2 Nynorsk: {$prediction}
```

Prompt B:

```
1 Omsett følgende setning til Nynorsk: {$text}
2 Nynorsk: {$prediction}
```

Prompt C:

```
1 Gje ei Nynorsk omsetjing av denne setninga: {$text}
2 Nynorsk: {$prediction}
```

Prompt D:

```
1 Kva blir "{text}" på Nynorsk?
2 Nynorsk: {prediction}
```

Full results The complete evaluation on translation to Nynorsk is given in Table 18 (BLEU scores in the top sub-table and chrF++ scores below).

BLEU SCORES

Prompt template	0-shot				1-shot				16-shot			
	A	B	C	D	A	B	C	D	A	B	C	D
NorMistral-11B	36.2	6.9	20.1	39.3	46.3 ^{±1.6}	45.4 ^{±1.5}	45.2 ^{±1.4}	44.7 ^{±1.5}	46.5 ^{±1.6}	48.0^{±1.6}	46.1 ^{±1.6}	47.3^{±1.6}
NorwAI-Mistral-7B	46.0	44.7	42.3	40.0	46.7^{±1.6}	46.7^{±1.6}	46.6 ^{±1.7}	45.9^{±1.6}	47.4 ^{±1.8}	46.5 ^{±1.8}	46.1 ^{±1.6}	46.5 ^{±1.7}
NorwAI-Llama2-7B	43.9	23.2	0.0	28.8	45.9 ^{±1.7}	46.7 ^{±1.8}	46.7^{±1.7}	45.2 ^{±1.7}	47.4^{±1.7}	47.2 ^{±1.7}	47.3^{±1.8}	47.0 ^{±1.8}
NorMistral-7B-warm	43.5	31.2	15.2	11.7	43.7 ^{±1.7}	44.6 ^{±1.8}	43.5 ^{±1.6}	43.5 ^{±1.8}	43.6 ^{±1.8}	44.7 ^{±1.7}	43.9 ^{±1.7}	44.2 ^{±1.6}
NorGPT-3B	1.5	2.4	0.8	1.6	0.1 ^{±0.0}	0.2 ^{±0.0}	0.1 ^{±0.0}	0.4 ^{±0.1}	0.2 ^{±0.0}	0.7 ^{±0.1}	2.6 ^{±0.5}	0.8 ^{±0.1}
Viking-7B	26.7	44.3	16.5	1.9	45.0 ^{±1.7}	44.4 ^{±1.6}	43.7 ^{±1.7}	42.3 ^{±1.6}	44.5 ^{±1.6}	43.9 ^{±1.5}	44.5 ^{±1.6}	45.6 ^{±1.6}
Viking-13B	42.5	31.6	11.1	1.7	45.2 ^{±1.7}	45.2 ^{±1.7}	44.8 ^{±1.7}	42.4 ^{±1.6}	45.2 ^{±1.7}	45.1 ^{±1.6}	45.5 ^{±1.7}	45.6 ^{±1.7}
Mistral-Nemo-12B	33.0	33.2	33.9	29.2	33.6 ^{±1.5}	34.7 ^{±1.4}	33.9 ^{±1.3}	35.1 ^{±1.5}	35.6 ^{±1.6}	35.4 ^{±1.7}	35.3 ^{±1.7}	35.7 ^{±1.6}

chrF++ SCORES

Prompt template	0-shot				1-shot				16-shot			
	A	B	C	D	A	B	C	D	A	B	C	D
NorMistral-11B	62.1	32.4	53.3	62.2	65.1 ^{±1.1}	64.6 ^{±1.1}	64.4 ^{±1.0}	63.9 ^{±1.1}	65.2 ^{±1.1}	66.5^{±1.2}	65.7^{±1.1}	65.9^{±1.1}
NorwAI-Mistral-7B	64.9	62.6	60.8	63.4	65.4^{±1.1}	64.9^{±1.1}	64.8 ^{±1.1}	64.5^{±1.1}	65.7^{±1.2}	65.1 ^{±1.2}	65.0 ^{±1.1}	65.1 ^{±1.1}
NorwAI-Llama2-7B	63.5	41.9	3.4	45.7	64.1 ^{±1.2}	64.8 ^{±1.1}	64.8^{±1.1}	63.5 ^{±1.1}	65.7^{±1.2}	65.8 ^{±1.1}	65.7^{±1.2}	65.2 ^{±1.2}
NorMistral-7B-warm	62.3	48.1	32.6	29.0	62.7 ^{±1.3}	63.6 ^{±1.3}	63.0 ^{±1.2}	63.0 ^{±1.3}	63.6 ^{±1.2}	64.6 ^{±1.1}	64.4 ^{±1.1}	63.9 ^{±1.2}
NorGPT-3B	15.5	16.6	12.2	16.9	4.9 ^{±0.2}	4.4 ^{±0.2}	3.3 ^{±0.2}	7.7 ^{±0.5}	4.0 ^{±0.2}	9.2 ^{±0.5}	15.0 ^{±1.1}	8.0 ^{±0.4}
Viking-7B	56.9	63.6	47.8	15.1	64.1 ^{±1.2}	64.4 ^{±1.1}	64.6 ^{±1.1}	62.8 ^{±1.2}	64.5 ^{±1.1}	64.3 ^{±1.0}	64.5 ^{±1.1}	65.2 ^{±1.1}
Viking-13B	62.9	60.4	44.0	14.9	64.7 ^{±1.1}	64.8 ^{±1.1}	64.4 ^{±1.1}	61.9 ^{±1.2}	64.6 ^{±1.1}	64.9 ^{±1.1}	65.0 ^{±1.1}	65.3 ^{±1.1}
Mistral-Nemo-12B	55.4	54.2	55.5	53.8	56.3 ^{±1.1}	57.0 ^{±1.1}	56.4 ^{±1.0}	56.9 ^{±1.1}	57.3 ^{±1.2}	57.2 ^{±1.2}	57.2 ^{±1.2}	57.5 ^{±1.2}

Table 17: **Complete results on translation from English to Nynorsk** We show the detailed results for each evaluated model, few-shot setting and prompt template. As the few-shot demonstrations are sampled randomly, we repeat them five times and show the mean accuracy as well as the standard deviation (rendered as superscript). The best results for each column are boldfaced, the overall best result for each few-shot setting is highlighted in blue.

B.10.3 English to Northern Sámi translation

We source the data from the English-Sámi parallel corpus from Tatoeba (Tiedemann, 2020), specifically the latest v2023-04-12 revision available on HuggingFace at <https://hf.co/datasets/Helsinki-NLP/tatoeba>. We deduplicate this corpus (both on the source and target side) and remove the empty entries – obtaining 53 examples in total.

Prompt templates We used the following five prompt templates for testing all language models on translation to Northern Sámi.

Prompt A:

```
1 Eangalsgiella: {text}
2 Davvisámegiella: {prediction}
```

Prompt B:

```
1 Engelsk: {$text}
2 Samisk: {$prediction}
```

Prompt C:

```
1 Oversett følgende setning til nordsamisk: {$text}
2 Nordsamisk: {$prediction}
```

Prompt D:

```
1 Gi en oversettelse til nordsamisk for denne setningen: {$text}
2 Nordsamisk: {$prediction}
```

Prompt E:

```
1 Hva blir "{$text}" på nordsamisk?
2 Nordsamisk: {$prediction}
```

Full results The complete evaluation on translation to Sámi is given in [Table 18](#) (BLEU scores in the top sub-table and chrF++ scores below).

BLEU SCORES

Prompt template	0-shot					1-shot					16-shot				
	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E
NorMistral-11B	31.5	31.0	14.8	23.2	14.7	24.8 ^{±5.1}	44.1^{±1.2}	15.7 ^{±1.6}	20.2 ^{±9.4}	33.7^{±8.2}	45.5^{±2.2}	48.8^{±2.4}	49.0^{±2.1}	48.8^{±2.3}	50.4^{±0.9}
NorwAI-Mistral-7B	21.2	25.0	24.0	16.7	20.7	26.6^{±1.7}	24.7 ^{±2.2}	24.4 ^{±1.6}	24.2^{±2.4}	25.0 ^{±1.1}	24.8 ^{±2.7}	25.1 ^{±2.6}	26.2 ^{±1.8}	26.8 ^{±2.0}	27.5 ^{±2.3}
NorwAI-Llama2-7B	16.9	10.7	3.3	0.0	15.4	24.7 ^{±2.1}	23.5 ^{±2.7}	24.5^{±2.1}	22.6 ^{±2.5}	22.5 ^{±1.8}	26.6 ^{±2.3}	27.6 ^{±2.0}	27.9 ^{±1.9}	27.6 ^{±2.1}	28.5 ^{±1.6}
NorMistral-7B-warm	12.2	5.7	0.0	0.0	0.0	11.2 ^{±3.1}	10.5 ^{±1.9}	12.2 ^{±3.2}	10.5 ^{±2.0}	8.4 ^{±0.5}	14.9 ^{±2.2}	17.2 ^{±1.9}	17.9 ^{±1.5}	15.9 ^{±1.2}	18.5 ^{±2.5}
NorGPT-3B	0.0	0.0	0.0	0.0	0.0	0.0 ^{±0.0}	0.0 ^{±0.0}	0.0 ^{±0.0}	0.0 ^{±0.0}	0.0 ^{±0.0}	0.0 ^{±0.0}	0.0 ^{±0.0}	0.0 ^{±0.0}	0.0 ^{±0.0}	0.0 ^{±0.0}
Viking-7B	0.0	0.0	0.0	0.0	0.0	0.0 ^{±0.0}	1.0 ^{±2.3}	0.7 ^{±1.5}	0.7 ^{±1.6}	0.5 ^{±1.1}	4.6 ^{±2.6}	6.8 ^{±1.8}	6.7 ^{±2.6}	7.8 ^{±1.6}	4.3 ^{±3.9}
Viking-13B	0.0	0.0	0.0	0.0	0.0	0.3 ^{±0.7}	3.3 ^{±3.2}	2.3 ^{±2.1}	4.3 ^{±4.0}	1.8 ^{±1.9}	9.8 ^{±1.1}	11.6 ^{±1.9}	11.5 ^{±1.9}	11.4 ^{±2.0}	11.2 ^{±1.2}
Mistral-Nemo-12B	0.0	0.0	0.0	0.0	0.0	0.0 ^{±0.0}	1.4 ^{±2.0}	0.8 ^{±1.8}	0.9 ^{±2.0}	0.9 ^{±1.9}	3.9 ^{±2.5}	6.5 ^{±2.2}	4.9 ^{±3.2}	5.7 ^{±3.8}	6.1 ^{±1.4}

chrF++ SCORES

Prompt template	0-shot					1-shot					16-shot				
	A	B	C	D	E	A	B	C	D	E	A	B	C	D	E
NorMistral-11B	51.8	58.8	49.5	58.2	47.3	55.5 ^{±2.4}	66.7^{±0.8}	54.6 ^{±1.8}	56.9 ^{±5.4}	63.2 ^{±3.7}	64.9 ^{±1.2}	67.2 ^{±0.6}	67.9 ^{±1.0}	67.7 ^{±1.4}	69.6^{±0.9}
NorwAI-Mistral-7B	44.9	45.8	53.9	50.6	49.6	50.7 ^{±1.3}	50.8 ^{±1.5}	50.8 ^{±0.8}	51.5 ^{±1.1}	50.4 ^{±0.7}	51.0 ^{±1.7}	51.7 ^{±1.7}	52.3 ^{±1.7}	52.4 ^{±1.2}	52.8 ^{±1.6}
NorwAI-Llama2-7B	37.3	30.0	46.8	44.1	39.3	45.6 ^{±2.0}	44.2 ^{±3.4}	48.1 ^{±2.3}	46.6 ^{±1.7}	44.0 ^{±1.4}	48.4 ^{±1.9}	50.2 ^{±1.3}	50.4 ^{±1.6}	50.2 ^{±1.4}	50.4 ^{±1.1}
NorMistral-7B-warm	27.4	22.5	21.4	26.8	23.8	32.3 ^{±2.4}	30.9 ^{±1.9}	32.7 ^{±2.2}	32.7 ^{±1.5}	29.9 ^{±0.7}	38.1 ^{±3.7}	37.7 ^{±3.5}	38.9 ^{±1.5}	37.2 ^{±1.5}	39.4 ^{±2.1}
NorGPT-3B	2.7	3.0	3.0	2.8	2.8	2.8 ^{±0.1}	2.8 ^{±0.1}	2.8 ^{±0.1}	2.8 ^{±0.1}	2.9 ^{±0.1}	2.9 ^{±0.1}	2.6 ^{±0.1}	3.0 ^{±0.1}	3.0 ^{±0.1}	3.1 ^{±0.1}
Viking-7B	10.4	11.5	11.5	11.7	8.1	10.3 ^{±0.9}	13.0 ^{±1.7}	12.3 ^{±1.5}	12.7 ^{±1.1}	12.4 ^{±1.3}	18.0 ^{±1.3}	19.6 ^{±0.7}	19.1 ^{±1.3}	19.7 ^{±1.0}	19.5 ^{±1.6}
Viking-13B	11.3	10.8	8.0	10.6	6.2	10.9 ^{±0.7}	14.9 ^{±1.4}	16.2 ^{±0.8}	16.1 ^{±0.7}	15.2 ^{±1.5}	24.5 ^{±2.5}	24.5 ^{±2.0}	24.9 ^{±2.2}	24.0 ^{±2.9}	25.2 ^{±1.7}
Mistral-Nemo-12B	11.5	12.1	12.2	12.4	10.5	14.2 ^{±2.0}	14.2 ^{±0.9}	15.3 ^{±1.1}	14.8 ^{±1.3}	14.9 ^{±0.9}	21.9 ^{±0.8}	22.3 ^{±1.7}	22.6 ^{±1.5}	22.4 ^{±1.6}	23.0 ^{±1.5}

Table 18: Complete results on translation from English to Northern Sámi We show the detailed results for each evaluated model, few-shot setting and prompt template. As the few-shot demonstrations are sampled randomly, we repeat them five times and show the mean accuracy as well as the standard deviation (rendered as superscript). The best results for each column are boldfaced, the overall best result for each few-shot setting is highlighted in blue.

Rethinking Low-Resource MT: The Surprising Effectiveness of Fine-Tuned Multilingual Models in the LLM Age

Barbara Scalvini

University of the Faroe Islands
barbaras@setur.fo

Annika Simonsen

University of Iceland
ans72@hi.is

Iben Nyholm Debess

University of the Faroe Islands
ibennd@setur.fo

Hafsteinn Einarsson

University of Iceland
hafsteinne@hi.is

Abstract

This study challenges the current paradigm shift in machine translation, where large language models (LLMs) are gaining prominence over traditional neural machine translation models. We focus on English-to-Faroese translation. We compare the performance of fine-tuned multilingual models, LLMs (GPT-SW3, Llama 3.1), and closed-source models (Claude 3.5, GPT-4). Our findings show that a finetuned NLLB model outperforms most LLMs, including larger models, in both automatic and human evaluations. We also demonstrate the effectiveness of using LLM-generated synthetic data for fine-tuning. While closed-source models like Claude 3.5 perform best overall, the competitive performance of smaller, finetuned models suggests a nuanced approach to low-resource machine translation. Our results highlight the potential of specialized multilingual models and the importance of language-specific knowledge. We discuss implications for resource allocation in low-resource settings and suggest future directions, including targeted data creation and comprehensive evaluation methods.

1 Introduction

The recent rise of LLMs has introduced new possibilities in machine translation (Lyu et al., 2024, 2023). LLMs demonstrated impressive performance across various language pairs, often through the use of in-context learning (Brown et al., 2020). These new opportunities often come at a price in terms of computational resources: LLMs have massive requirements in terms of pre-training data and high-end hardware. Hardware requirements can sometimes be mitigated by using closed-source LLM APIs (e.g., OpenAI API).

However, this approach introduces issues related to transparency and license limitations.

These limitations and high requirements disproportionately affect low-resource languages and communities. For such languages, lack of resources can often extend beyond data scarcity and effectively imply lack of computational infrastructure and expertise, rendering the use of APIs offered by tech giants the only available option. This is the case for Faroese, an Insular Scandinavian language and official language of the Faroe Islands.

Neural machine translation (NMT) models are less demanding in terms of computational resources. However, due to their more limited reasoning capabilities compared to LLMs, they often underperform in low-resource settings. Nonetheless, there are potential strategies to leverage the linguistic knowledge of an LLM in conjunction with lightweight MT models to optimize performance while minimizing resource requirements. One such approach is to use LLMs to augment parallel datasets, allowing a lighter MT model to be trained on this synthetic data (Yang and Nicolai, 2023).

In NLP, efficiency encompasses various factors like data requirements, model size, training costs, and performance metrics. This paper focuses on the relationship between model performance and size, a crucial consideration for real-world applications. We explore different approaches to English-to-Faroese machine translation, investigating how various techniques balance translation quality with model compactness. Our research aims to shed light on the trade-offs between performance and model size in this specific language pair. We will compare the following approaches, in the context of English to Faroese MT:

- Using LLMs in a few-shot learning setting.
- Fine-tuning LLMs for translation (English-

to-Faroese).

- Using a multilingual NMT out of the box.
- Fine-tuning a multilingual model on English-Faroese parallel data.
- Fine-tuning a multilingual model on English-to-Faroese parallel data and LLM-generated synthetic parallel data.

These strategies will be compared based on automatic and human evaluation. We will be comparing the following open-source LLMs: Llama 3.1, (Meta) (Dubey et al., 2024) in its 8B version, and GPT-SW3, a generative model for the Nordic languages, primarily Swedish, (Ekgren et al., 2022, 2024), in its 1.3, 6.7 and 40B version.

Their performance will be compared to closed-source models such as Claude 3.5 Sonnet (Anthropic, 2024) by Anthropic, GPT-4 Turbo (OpenAI et al., 2024) and GPT-4o (OpenAI, 2024) by OpenAI. We compare the LLMs with No Language Left Behind (NLLB) (Team, 2024), an open-source NMT multilingual model covering, among other under-resourced languages, Faroese. All new models produced via fine-tuning in this paper are now publicly available.¹

2 Background and related work

2.1 LLMs for translation

The emergence of LLMs has challenged the dominance of sequence-to-sequence transformer-based models in the field of machine translation (MT) (Lyu et al., 2024; Hendy et al., 2023; Robinson et al., 2023). LLMs like initially observed for GPT-3 can perform translations with minimal input through in-context learning (ICL), significantly reducing the data requirements typically needed for the training process. This ability to achieve state-of-the-art results with minimal data has highlighted the potential of LLMs as a promising solution for low-resource translation. A few studies have investigated methods to

enhance LLMs’ MT capabilities in low-resource settings, employing techniques such as layer adaptation and fine-tuning (Tran et al., 2024), retrieval-augmented prompting (Merx et al., 2024), integration with rule-based systems (Coleman et al., 2024), and synthetic parallel data generation with an LLM (Yang and Nicolai, 2023). Additionally, LLMs have demonstrated remarkable performance as evaluators of translation quality, achieving near-human accuracy, although these results have been primarily studied in high-resource languages (Karpinska and Iyyer, 2023; Fernandes et al., 2023; Huang et al., 2024; Kocmi and Federmann, 2023). However, the effectiveness of LLMs in low-resource contexts, such as Faroese, remains relatively underexplored. Some studies suggest that LLM-driven translation may be less competitive for low-resource languages (Robinson et al., 2023), when compared to their higher resource counterparts.

2.2 Machine Translation for Faroese

In recent years, a few notable efforts have focused on improving coverage for Faroese in machine translation (MT). A key initiative was the creation of Sprotin’s parallel corpus (Mikkelsen, 2021), which includes around 100,000 short human-translated English-Faroese sentences. This corpus supported Faroese’s integration into Microsoft Translator and an Icelandic Machine Translation platform called Vélþýðing, by the Icelandic company Miðeind. The rise of multilingual MT models has led to initiatives like Google’s MADLAD 400 (Kudugunta et al., 2023) and Meta’s No Language Left Behind (NLLB) (Team, 2024), targeting low-resource languages such as Faroese. Since July 2024, Faroese has also been included in Google Translate (Bapna et al., 2022). The linguistic proximity of Faroese to its higher-resource relatives, the Scandinavian languages, makes it an ideal candidate for transfer learning (Snæbjarnarson et al., 2023). GPT-SW3, an LLM trained on English and Scandinavian languages, has demonstrated significant potential for understanding Faroese (Scalvini and Debess, 2024). Likewise, GPT-4 has shown promising results in Faroese sentiment analysis (Debess et al., 2024) and Faroese-to-English translation (Simonsen and Einarsson, 2024).

¹https://huggingface.co/barbaroo/llama3.1_translate_8B,
https://huggingface.co/barbaroo/gptsw3_translate_1.3B,
https://huggingface.co/barbaroo/gptsw3_translate_6.7B,
https://huggingface.co/barbaroo/nllb_200_1.3B_en_fo,
https://huggingface.co/barbaroo/nllb_200_600M_en_fo

3 Methods

3.1 Experiments

In this study, we evaluate machine translation performance for English into low-resource Faroese of various models: 5 LLM models (GPT-SW3, Llama 3.1, GPT-4 Turbo, GPT-4o, Claude 3.5 Sonnet) and one multilingual MT model covering Faroese in its pre-training phase, NLLB. We chose NLLB as representative of multilingual MT because it demonstrated the highest potential in earlier studies (Simonsen, 2024). Since the goal of this paper is to analyze which settings are best for open-source MT in a low-resource scenario, we mostly preferred smaller, less computationally costly versions of the models. We utilize NLLB in its 600M and 1.3B parameters, and fine-tune LLMs that have sizes below 10B parameters, as these would be the ones most likely to be fine-tuned and deployed on common, commercial hardware. In order to investigate different modalities to exploit LLM language capabilities in machine translation, we fine-tune the MT model, NLLB, on LLM generated parallel sentences, in addition to the available human made corpus. This approach is presented as an alternative to either directly deploying the LLM in a few-shot manner, or instruct fine-tuning it directly for the desired translation direction. We evaluate these models both automatically and by human evaluation, for which we build an openly available evaluation platform online². The performance of these open-source models is also benchmarked against that of three of the most popular closed-source models (GPT-4 Turbo, GPT-4o and Claude 3.5 Sonnet), for comparison.

3.2 Datasets

Faroese, as a low-resource language, lacks substantial parallel datasets for machine translation. The most comprehensive resource is the Sprotin corpus (Mikkelsen, 2021), though it may miss Faroese-specific cultural elements since it was translated from English. Recent studies have explored using LLMs to generate synthetic parallel datasets, like the `fo_en_synthetic`³ dataset (Scalvini and Debess, 2024), created through back-translation with GPT-SW3, contain-

ing 70,000 sentences from the BLARK corpus (Simonsen et al., 2022).

The inclusion of Faroese in Meta’s No Language Left Behind (NLLB) initiative (Team, 2024) enabled the language’s integration into the FLORES-200 benchmark for machine translation. Currently, FLORES-200 is the only available evaluation benchmark for Faroese translation, making it our choice for the automatic comparison of model performance. While FLORES-200 is a well-established benchmark in the field, it has known limitations, such as its domain composition and a narrow representation of cultural elements, given that it was originally translated from English (Simonsen and Einarsson, 2024). To address this, we manually compiled a small dataset of 200 English sentences for human evaluation. The dataset consists of 68 sentences sourced from documents produced by the University of the Faroe Islands (Strategic Plan 2025-2030), 56 from the webpage of the Nordic Council⁴ and 92 sentences from international news outlets such as BBC, CNN, and Al Jazeera. The dataset is publicly available on Hugging Face, together with all synthetic translations produced in the context of this paper.⁵ All sentences were guaranteed to be created within a specific recent time period, ensuring that none of the data had been used in the training of any models included in the study. The inclusion of sentences from Faroese and Nordic-related contexts aimed to better represent Faroese-specific cultural elements, which are typically underrepresented in datasets despite being highly relevant to the end users of Faroese machine translation products. For example, using sentences from locally relevant contexts included concepts and named entities that actually have a Faroese translation, as they are Faroese or Nordic by origin (e.g. the local institution ‘Statistics Faroe Islands’ - *Hagstova Føroya*). This is opposed to many concepts or entities in sentences from international sources, where the translation of such can be difficult due to the entities not having a direct Faroese translation, as they are often irrelevant to Faroese society (e.g. the concept of a ‘US Governor’, which has no Faroese equivalent). These foreign concepts make evaluation more complex. Furthermore, using locally or regionally sourced data together with internationally sourced data enables evaluating con-

²<https://github.com/Haffil12/error-span-labelling>

³https://huggingface.co/datasets/barbaroo/fo_en_synthetic

⁴<https://www.norden.org/en>

⁵https://huggingface.co/datasets/barbaroo/news_en_fo

tent for real-use Faroese scenarios.

3.3 Prompting LLMs for English to Faroese translation

All LLMs used in this study were prompted in a few-shot fashion. Each translation query consisted of a prompt presenting the model with 5 randomly selected examples of English to Faroese translation. Examples were selected from a small subset of the Sprotin corpus comprising of 25 manually selected parallel sentences. These sentences were selected by a Faroese linguist based on the following criteria: 1) the meaning of the sentence is fully preserved in its translation 2) all words have unambiguous meaning, 3) they present simple syntax (declarative sentences or interrogative sentences, excluding subordinate clauses or sentences), 4) there are no typographical and inflectional errors. Two different prompting strategies were used for open-source (GPT-SW3 and Llama) and closed-source models (GPT-4o, Claude 3.5 Sonnet). This distinction was made in order to provide each model with an optimal prompting format.

3.4 Open-source models

We used the base versions of the Llama 3.1 and GPT-SW3 models. To facilitate model comprehension, we framed the prompt as a language completion task. Each example was structured as follows:

The English sentence {english_sentence} is translated to Faroese as {faroese_sentence}

The query followed the same format but omitted the Faroese translation:

The English sentence {english_sentence} is translated to Faroese as

This approach minimized the number of failed translation outputs.

3.5 Closed-source models

Closed-source models (GPT-4 Turbo, GPT-4o and Claude 3.5 Sonnet) were prompted via their respective APIs. The prompt structure was then adapted to the API format, with a system prompt containing the few-shot examples and the instructions of the task (*When I give you a sentence in English, you translate it into Faroese. Only answer with a translation.*) and a translation prompt containing the translation query.

3.6 Fine-tuning of models for English to Faroese translation

All open-source models in this study, except GPT-SW3 40B, were also fine-tuned for English-to-Faroese translation. For the LLMs, fine-tuning was conducted for three epochs with early stopping, using the Sprotin corpus. We adopted the Alpaca prompting format for both Llama and GPT-SW3, which includes an instruction ("Translate this sentence from English to Faroese"), an input (the English sentence), and an output (the Faroese sentence). Training was performed in 8-bit precision to reduce computational resource requirements. Two versions of NLLB, with 0.6 billion and 1.3 billion parameters, were also fine-tuned for English-to-Faroese translation. The training was carried out in two settings: (1) using only the Sprotin corpus and (2) using a combination of the Sprotin corpus and the `fo_en_synthetic` dataset. These different settings were chosen to demonstrate the potential benefits of incorporating LLM-generated parallel sentences to improve translation quality. The complete training configuration can be found in our GitHub repository.⁶

3.7 Evaluation

Automatic evaluation is performed using the metrics BLEU, ChrF and BERTscore. We do not use more advanced neural metrics, as these are not currently available for Faroese.

For human evaluation, we adopted the recently developed Error Span Annotation (ESA) metric proposed by Kocmi et al. (2024). ESA combines elements from two established methods: the overall scoring approach of Direct Assessment (DA) and the error severity span markings from Multidimensional Quality Metrics (MQM). In their study, Kocmi et al. (2024) compared ESA to MQM and DA across several MT systems. Their findings demonstrated that ESA offers a more cost-effective and time-efficient alternative to MQM without compromising evaluation quality. The ESA operates with a dual error system, which is less complex to the annotator compared to the multiple error categories and subcategories of MQM.

We created an annotation user interface based on the task description in Kocmi et al. (2024). Figure 1 shows an example from the interface. The

⁶https://github.com/barbaroo/finetune_translation

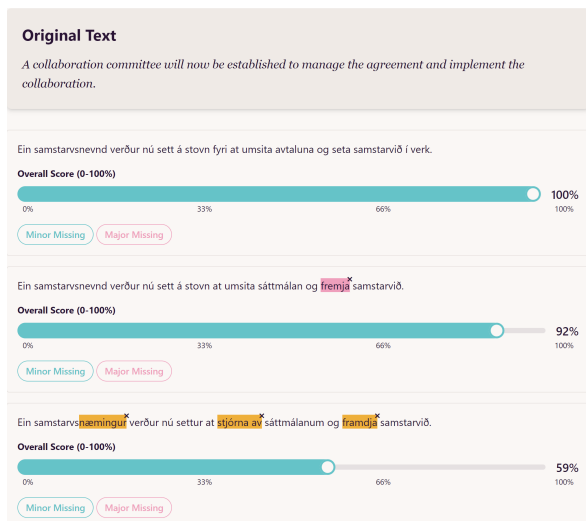


Figure 1: The annotation interface. Annotators were presented with the original text along with four translations (three shown here). The annotators mark any segment and are prompted to label it minor (pink) or major (orange). The annotators assign an overall score (1-100) to each translation (blue). For each translation, the annotators can optionally mark missing elements as major or minor.

annotation process was the following: the annotator is presented with the original English sentence along with four Faroese translations. The annotator then marks all the errors in the Faroese translations and to each error assigns one of the two severity levels: **major** or **minor**. Additionally, there is a label for omission errors, called *minor/major missing*. After marking the errors, the annotators assign each translation with an overall score from 0 to 100. The overall score reflects translation quality in a broad sense, covering adequacy, fluency and comprehension.

3.8 Annotator Guidelines

For the human evaluation, we had two human annotators, both linguists and native speakers of Faroese. The annotators developed the annotation guidelines together, using the original guidelines from Kocmi et al. (2024) as a starting point and adjusting it to fit the specific task. The full guidelines are shown below.

Approach

Annotators identified and marked error spans in translations, assigning severity levels (major or minor) to each. They then provided an independent, holistic overall score that could consider fac-

tors beyond marked errors, such as fluency. Major errors include significant meaning changes, mistranslations, foreign words, untranslated named entities, and synthetic words (constructed well-structured and sensible words, that are however not recognized in human language use). Minor errors encompass slight meaning alterations, style issues, grammatical mistakes, spelling errors, and punctuation problems.

Other

- Grammatical errors spanning over multiple words are marked as a single error
- If the source sentence has an error, annotators consider this original error in their evaluation of the translations
- If the source sentence is erroneous to an extent where translation output is completely off, all 4 sentences are given 0% and no errors are marked.

Scoring

This method provides two scores: an ESA overall score (0-100) and the ESA_{spans} (number and severity of errors). The ESA_{spans} is calculated as segment score, $SEG, SCORE = -1 * N_{MINOR} - 4.8 * N_{MAJOR}$, as suggested by Kocmi et al. (2024). As the evaluations of overall score and errors are meant to be performed independently, these scores can be treated separately.

4 Results

4.1 Automatic evaluation

The results for automatic evaluation on the FLORES-200 benchmark for all models can be found in Table 1. For all three different scores, we can see how closed-source Claude yields the best results. However, NLLB 1.3B, in its fine-tuned version (Sprotin + fo_en_synthetic) scores second overall and first among open-source models. A representation of the CHRF score with respect to model size, for all models under 10B is shown in Figure 2. As we can see the top left corner, representing the best performing models with respect to their hardware requirements, is dominated by fine-tuned NLLB models. NLLB 1.3 fine-tuned with the Sprotin corpus alone does yield a better performance with respect to fine-tuned LLMs, and with respect to GPT-4o as well. The performance is anyway sensibly increased (1

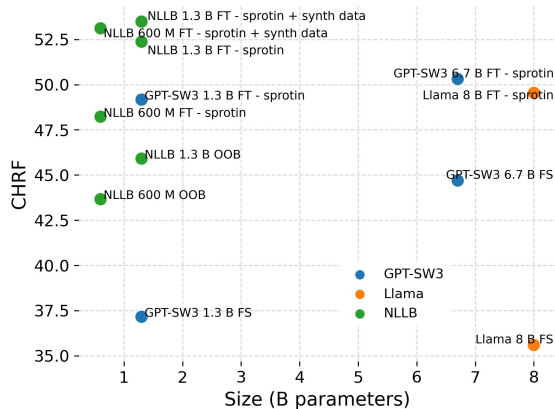


Figure 2: Translation performance for all models (with fewer than 10 billion parameters) in the automatic evaluation, quantified by the CHRf score. The performance is plotted against the model size, expressed in billions of parameters.

ChrF point and 3 BLEU points) by adding LLM-generated synthetic data. Llama 3.1 8 B does yield the worst performance in a few-shot setting, demonstrating however great potential for improvement after fine-tuning, beating out of the box NLLB and GPT-SW3 1.3 B.

4.2 Human evaluation

When picking models for human evaluation, we picked the best models from each category according to the automatic evaluation (see Table 1). We picked the following four models: GPT-SW3 6.7B - Sprotin, Llama 3.1 8B - Sprotin, NLLB 1.3B - Sprotin + `fo_en_synthetic` and we also picked the best performing closed-source model, Claude 3.5 Sonnet. The results from the human evaluation, in terms of ESA - overall quality score - and ESA_{spans} scores, are displayed in Table 2. Claude 3.5 Sonnet shows the best performance of the four, with NLLB getting the best results for the open-source models. GPT-SW3, despite the smaller size, does beat Llama 3.1 in both human and automatic evaluation, showing that family language specific knowledge is an advantage for models of comparable sizes.

Figure 3 shows the average ESA score for the two annotators separately, showing that the two annotators agree on how the models should be ranked in terms of translation quality. The ESA_{spans} score can be deconstructed into different error types, as shown in Figure 4. Here we see the

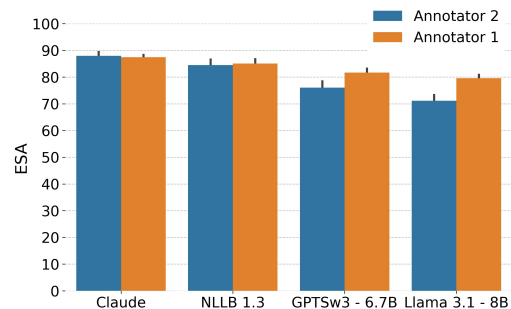


Figure 3: Average overall quality score (ESA) per model, assigned by the two annotators. "Average overall quality score (ESA) per model, as assigned by the two annotators. All models in the plot are shown in their fine-tuned versions (GPT-SW3 6.7B - Sprotin, Llama 3.1 8B - Sprotin, NLLB 1.3B - Sprotin + `fo_en_synthetic`), except for Claude."

two best performing models, Claude and NLLB 1.3, have comparable number of minor and major errors, with Claude performing better when it comes to preserving content (missing content, major and minor). NLLB and Claude do display comparable performance across the metrics. While the ESA scores assigned to the two models are statistically distinct ($p = 0.017$, as calculated by Mann-Whitney U test), the same cannot be said for the ESA_{spans} scores ($p = 0.465$). GPT-SW3 6.7B seems to struggle the most with preserving content due to the greatest number of missing content errors. However, it is performing largely better than Llama 3.1 8B when it comes to number of errors.

4.2.1 Annotator agreement

Figure 5 shows the distribution of ESA scores from both annotators. While mostly overlapping, the distributions have different variances (Levene test, $p = 1.34 \times 10^{-28}$). Krippendorff's alpha indicates moderate to strong agreement for absolute ESA (0.58) and ESA_{spans} (0.67) scores. We also converted scores to rankings for each translation query, assigning equal ranks for tied scores. Kendall's W analysis of these rankings showed moderate to strong inter-annotator agreement (ESA: 0.514, ESA_{spans} : 0.518), further supporting the reliability of our annotations.

4.3 Common Error Patterns

From a qualitative perspective the annotators report some common error patterns that emerged in

Model	BLEU	CHRF	BERTScore (f1)
GPT-SW3 40 B	0.173 \pm 0.005	48.3 \pm 0.4	0.9472 \pm 0.0005
GPT-SW3 6.7 B	0.119 \pm 0.004	44.7 \pm 0.4	0.9373 \pm 0.0005
GPT-SW3 1.3 B	0.084 \pm 0.004	37.1 \pm 0.4	0.9279 \pm 0.0006
GPT-SW3 6.7 B* - Sprotin	0.183 \pm 0.006	50.3 \pm 0.4	0.951 \pm 0.001
GPT-SW3 1.3 B* - Sprotin	0.179 \pm 0.005	49.2 \pm 0.4	0.947 \pm 0.001
Llama 3.1 8 B	0.062 \pm 0.003	35.6 \pm 0.3	0.9311 \pm 0.0005
Llama 3.1 8 B* - Sprotin	0.175 \pm 0.005	49.5 \pm 0.4	0.9487 \pm 0.0005
NLLB 600 M	0.129 \pm 0.005	43.7 \pm 0.4	0.9428 \pm 0.0005
NLLB 600 M* - Sprotin	0.171 \pm 0.005	48.2 \pm 0.5	0.9458 \pm 0.0006
NLLB 600 M* - Sprotin + fo_en_synthetic	0.200 \pm 0.006	53.1 \pm 0.4	0.9524 \pm 0.0005
NLLB 1.3 B	0.161 \pm 0.005	45.9 \pm 0.4	0.9459 \pm 0.0005
NLLB 1.3 B* - Sprotin	0.209 \pm 0.006	52.4 \pm 0.4	0.9516 \pm 0.0005
NLLB 1.3 B* - Sprotin + fo_en_synthetic	0.212 \pm 0.006	53.5 \pm 0.4	0.9530 \pm 0.0005
GPT-4 Turbo	0.193 \pm 0.006	52.7 \pm 0.4	0.9518 \pm 0.0005
GPT-4o	0.191 \pm 0.005	51.7 \pm 0.4	0.9509 \pm 0.0005
Claude 3.5 Sonnet	0.226 \pm 0.006	55.3 \pm 0.4	0.9546 \pm 0.0005

Table 1: Model performance metrics, calculated over the FLORES-200 dataset. All scores pertaining to LLMs were obtained in a few shot setting, with the exception of those that were fine-tuned (*). The mention of *Sprotin* and *fo_en_synthetic* indicate which datasets was the model fine-tuned on. The error term represents the standard error of the mean for 1012 translations.

Model	ESA	ESA _{spans}	N (ESA = 0)
Claude 3.5 Sonnet	87.7 \pm 0.5	-2.3 \pm 0.1	0
NLLB 1.3B - Sprotin + fo_en_synthetic	84.8 \pm 0.7	-2.3 \pm 0.1	3
Llama 3.1 8B - Sprotin	75.3 \pm 0.6	-6.3 \pm 0.2	0.5*
GPT-SW3 6.7B - Sprotin	78.8 \pm 0.7	-4.6 \pm 0.2	2

Table 2: Comparison of Models based on human evaluation. The table portrays ESA and ESA_{spans} scores, and number of failed translations, expressed in terms of number of translations that received a 0 as ESA score, N (ESA = 0). The * indicates that only one of the two annotators assigned a 0 score, therefore we do not assign N = 1, but N = 0.5. The error term represents the standard error of the mean for 215 translations.

the annotation process. Taking a closer look at linguistic errors, morphological errors seem more common with inflectional errors in adjectives being prevalent. Errors in translating named entities were also frequent, as the models struggle with identifying the correct entities in Faroese. An interesting observation is the occurrences of a type of error, where the models make up new words, that are structurally well-formed for Faroese and semantically appropriate to various extents, but are complete neologisms and not recognised in natural Faroese language use, spoken or written. These words were typically compound words, like the example of "artificial intelligence" being translated into *telduheimsniðgððskapur*. Finally, all

models tend to translate word-for-word, which leads to literal translations of idioms and fixed phrases. Error patterns like these can suggest effective focus areas when creating parallel data for improving the models.

5 Discussion

Our study on English to Faroese machine translation reveals several important findings that provide new insights into the relative strengths of different approaches to low-resource language translation, including large language models and specialized multilingual models. Surprisingly, the fine-tuned NLLB model outperformed most LLMs, including GPT-4 and GPT-SW3 40B, in both au-

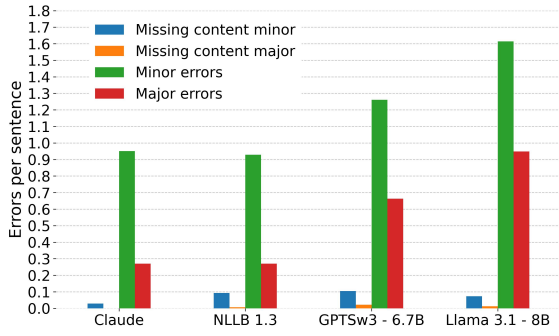


Figure 4: Average error type per model, as defined by the ESA framework: minor error, major error, minor missing content and major missing content. All models in the plot are shown in their fine-tuned versions (GPT-SW3 6.7B - Sprotin, Llama 3.1 8B - Sprotin, NLLB 1.3B - Sprotin + fo_en_synthetic), except for Claude.

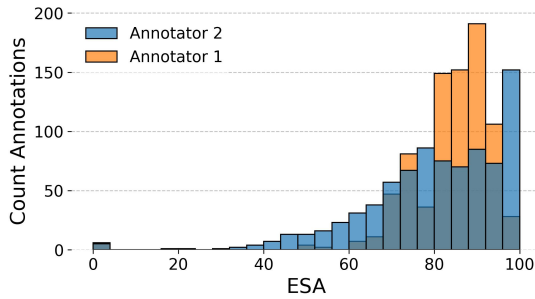


Figure 5: Distribution of overall quality scores (ESA) given by the annotators.

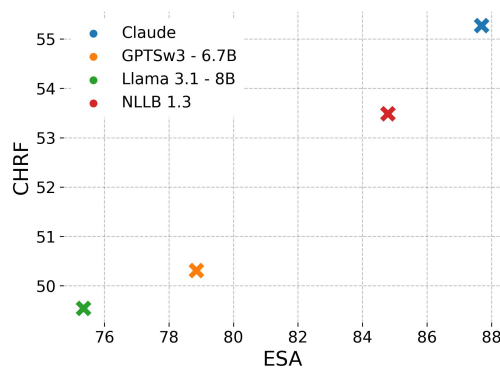


Figure 6: Scatterplot of CHRF scores versus overall quality scores (ESA). All models in the plot are shown in their fine-tuned versions (GPT-SW3 6.7B - Sprotin, Llama 3.1 8B - Sprotin, NLLB 1.3B - Sprotin + fo_en_synthetic), except for Claude.

tomatic and human evaluations. This suggests that specialized multilingual models, when fine-tuned appropriately, can be highly effective, often achieving comparable or even superior performance to larger LLMs for specific language pairs. The success of NLLB highlights the importance of domain-specific training and more compact, efficient models, which can be especially valuable in low-resource settings where computational power may be limited. Furthermore, the performance of GPT-SW3, despite its smaller size compared to Llama 3.1, underscores the critical role of language-specific knowledge in translation tasks. These findings have significant implications for resource allocation and model selection in low-resource language translation.

While automatic and human evaluations generally aligned on model rankings, there were key differences in perceived quality. This reveals the limitations of relying solely on automatic metrics, especially for low-resource languages. Human evaluations showed that while Claude 3.5 Sonnet and NLLB 1.3B had similar error counts, Claude performed better in content preservation and received a higher overall ESA score, suggesting that evaluators may prioritize factors like fluency and naturalness beyond just error quantity.

The improvement in NLLB's performance when fine-tuned on both the Sprotin corpus and LLM-generated synthetic data (fo_en_synthetic) highlights the potential of leveraging LLMs to augment training data for low-resource languages (Yang and Nicolai, 2023). This strategy could enhance translation quality in resource-constrained settings. However, despite these gains, all evaluated models still exhibit significant errors, falling short of human-quality translation, which calls for further research. These findings suggest that fine-tuning smaller, specialized models may offer a more cost-effective solution than relying on large LLMs, and that targeted data creation, informed by common error patterns, could further boost performance. Additionally, the discrepancies between automatic and human evaluations emphasize the need for more nuanced evaluation methods for low-resource language translation.

Future work should focus on iterative improvement techniques such as back-translation, exploring methods to distill knowledge from larger LLMs to smaller, more deployable models, and

creating more diverse and representative parallel datasets for low-resource languages like Faroese.

6 Conclusion

Our study on English to Faroese machine translation offers a nuanced perspective on the effectiveness of different approaches to low-resource language pairs, highlighting how fine-tuned models like NLLB can rival or outperform larger LLMs for low-resource languages. This suggests that focusing on fine-tuning smaller models and creating targeted synthetic datasets may be more effective and resource-efficient. Despite improvements, all models still fall short of human-quality translation, emphasizing the need for further research on error patterns, data augmentation, and better evaluation methods. Advancing low-resource translation likely calls for a tailored combination of specialized models with effective data augmentation strategies.

7 Limitations

One possible limitation of our study is that we did not consider how much Faroese text these models were exposed to during pre-training. We excluded this information because, for some models, it is not publicly available: we do not have access to closed-source training data, and detailed documentation on the data sources for Llama 3.1 had not been released as of December 2024. GPT-SW3 does not officially cover Faroese, although it is possible that some Faroese text was misclassified as Icelandic within the training data. Conversely, NLLB was trained on approximately 2.8 million Faroese–English bitext sentences (Schwenk et al., 2020; Fan et al., 2020), which are now available on Opus (Tiedemann, 2012). The amount of Faroese these models have seen certainly influences their final performance; however, quantifying this exposure is difficult for most LLMs, making such comparisons challenging.

References

- Anthropic. 2024. Claude 3.5 sonnet. <https://www.anthropic.com>. Proprietary software, closed-source.
- Ankur Bapna, Isaac Caswell, Julia Kreutzer, Orhan Firat, Daan van Esch, Aditya Siddhant, Mengmeng Niu, Pallavi Baljekar, Xavier Garcia, Wolfgang Macherey, Theresa Breiner, Vera Axelrod, Jason Riesa, Yuan Cao, Mia Xu Chen, Klaus Macherey, Maxim Krikun, Pidong Wang, Alexander Gutkin, Apurva Shah, Yanping Huang, Zhifeng Chen, Yonghui Wu, and Macduff Hughes. 2022. Building machine translation systems for the next thousand languages. *arXiv preprint arXiv:2205.03983*.
- 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. Curran Associates, Inc.
- Jared Coleman, Bhaskar Krishnamachari, Ruben Rosales, and Khalil Iskarous. 2024. LLM-assisted rule based machine translation for low/no-resource languages. In *Proceedings of the 4th Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP 2024)*, pages 67–87, Mexico City, Mexico. Association for Computational Linguistics.
- Iben Nyholm Debess, Annika Simonsen, and Hafsteinn Einarsson. 2024. Good or bad news? Exploring GPT-4 for sentiment analysis for Faroese on a public news corpora. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7814–7824, Torino, Italia. ELRA and ICCL.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiohu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade

Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnston, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yearly, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Rapparthi, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuwei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Sto-

jkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, DingKang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelen, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabza, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang,

- Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vitor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Ariel Ekgren, Amaru Cuba Gyllensten, Evangelia Gogoulou, Alice Heiman, Severine Verlinden, Joey Öhman, Fredrik Carlsson, and Magnus Sahlgren. 2022. Lessons learned from GPT-SW3: Building the first large-scale generative language model for Swedish. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3509–3518, Marseille, France. European Language Resources Association.
- Ariel Ekgren, Amaru Cuba Gyllensten, Felix Stollenwerk, Joey Öhman, Tim Isbister, Evangelia Gogoulou, Fredrik Carlsson, Judit Casademont, and Magnus Sahlgren. 2024. GPT-SW3: An autoregressive language model for the Scandinavian languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7886–7900, Torino, Italia. ELRA and ICCL.
- Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Edouard Grave, Michael Auli, and Armand Joulin. 2020. Beyond english-centric multilingual machine translation. *CoRR*, abs/2010.11125.
- Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André Martins, Graham Neubig, Ankush Garg, Jonathan Clark, Markus Freitag, and Orhan Firat. 2023. The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1066–1083, Singapore. Association for Computational Linguistics.
- Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. How good are gpt models at machine translation? a comprehensive evaluation. *arXiv preprint arXiv:2302.09210*.
- Xu Huang, Zhirui Zhang, Xiang Geng, Yichao Du, Jiajun Chen, and Shujian Huang. 2024. Lost in the source language: How large language models evaluate the quality of machine translation. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 3546–3562, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.
- Marzena Karpinska and Mohit Iyyer. 2023. Large language models effectively leverage document-level context for literary translation, but critical errors persist. In *Proceedings of the Eighth Conference on Machine Translation*, pages 419–451, Singapore. Association for Computational Linguistics.
- Tom Kocmi and Christian Federmann. 2023. Large language models are state-of-the-art evaluators of translation quality. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 193–203, Tampere, Finland. European Association for Machine Translation.
- Tom Kocmi, Vilém Zouhar, Eleftherios Avramidis, Roman Grundkiewicz, Marzena Karpinska, Maja Popović, Mrinmaya Sachan, and Mariya Shmatova. 2024. Error span annotation: A balanced approach for human evaluation of machine translation. *arXiv preprint arXiv:2406.11580*.
- Sneha Kudugunta, Isaac Caswell, Biao Zhang, Xavier Garcia, Christopher A. Choquette-Choo, Katherine Lee, Derrick Xin, Aditya Kusupati, Romi Stella, Ankur Bapna, and Orhan Firat. 2023. Madlad-400: A multilingual and document-level large audited dataset. *arXiv preprint arXiv:2309.04662*.
- Chenyang Lyu, Zefeng Du, Jitao Xu, Yitao Duan, Minghao Wu, Teresa Lynn, Alham Fikri Aji, Derek F. Wong, and Longyue Wang. 2024. A paradigm shift: The future of machine translation lies with large language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 1339–1352, Torino, Italia. ELRA and ICCL.
- Chenyang Lyu, Jitao Xu, and Longyue Wang. 2023. New trends in machine translation using large language models: Case examples with chatgpt. *arXiv preprint arXiv:2305.01181*.
- Raphaël Merx, Aso Mahmudi, Katrina Langford, Leo Alberto de Araujo, and Ekaterina Vylomova. 2024. Low-resource machine translation through retrieval-augmented LLM prompting: A study on the Mambai language. In *Proceedings of the 2nd*

Workshop on Resources and Technologies for Indigenous, Endangered and Lesser-resourced Languages in Eurasia (EURALI) @ LREC-COLING 2024, pages 1–11, Torino, Italia. ELRA and ICCL.

Jonhard Mikkelsen. 2021. Sprotin sentences. https://raw.githubusercontent.com/Sprotin/translations/main/sentences_en-fo.strict.csv. Accessed: October 13, 2023.

OpenAI. 2024. Gpt-4o. <https://www.openai.com>. Proprietary software, closed-source.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, 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 Boddonoff, 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, Vin-

nie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mely, Ashvin Nair, Reiichiro Nakano, Rameev 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. *arXiv preprint arXiv:2303.08774*.

Nathaniel Robinson, Perez Ogayo, David R. Mortensen, and Graham Neubig. 2023. ChatGPT MT: Competitive for high- (but not low-) resource languages. In *Proceedings of the Eighth Conference on Machine Translation*, pages 392–418, Singapore. Association for Computational Linguistics.

Barbara Scalvini and Iben Nyholm Debess. 2024. Evaluating the potential of language-family-specific generative models for low-resource data augmentation: A Faroese case study. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 6496–6503, Torino, Italia. ELRA and ICCL.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, and Armand Joulin. 2020. Ccmatrix: Mining billions of high-quality parallel sentences on the web. *arXiv preprint arXiv:1911.04944*.

Annika Simonsen. 2024. Improving Machine Translation for Faroese using ChatGPT-Generated Parallel

Data. Master’s thesis, University of Iceland, Reykjavík.

Annika Simonsen and Hafsteinn Einarsson. 2024. A Human Perspective on GPT-4 Translations: Analysing Faroese to English News and Blog Text Translations. In *Proceedings of the 25th Annual Conference of the European Association for Machine Translation (Volume 1)*, pages 24–36, Sheffield, UK. European Association for Machine Translation (EAMT).

Annika Simonsen, Sandra Saxov Lamhauge, Iben Nyholm Debess, and Peter Juel Henriksen. 2022. Creating a basic language resource kit for Faroese. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4637–4643, Marseille, France. European Language Resources Association.

Vésteinn Snæbjarnarson, Annika Simonsen, Goran Glavaš, and Ivan Vulić. 2023. Transfer to a low-resource language via close relatives: The case study on Faroese. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 728–737, Tórshavn, Faroe Islands. University of Tartu Library.

NLLB Team. 2024. Scaling neural machine translation to 200 languages. *Nature*, 630:841–846.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

Khanh-Tung Tran, Barry O’Sullivan, and Hoang Nguyen. 2024. Irish-based large language model with extreme low-resource settings in machine translation. In *Proceedings of the Seventh Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2024)*, pages 193–202, Bangkok, Thailand. Association for Computational Linguistics.

Wayne Yang and Garrett Nicolai. 2023. Neural machine translation data generation and augmentation using chatgpt. *arXiv preprint arXiv:2307.05779*.

Prompt Engineering Enhances Faroese MT, but Only Humans Can Tell

Barbara Scalvini

University of the Faroe Islands
barbaras@setur.fo

Iben Nyholm Debess

University of the Faroe Islands
ibennd@setur.fo

Annika Simonsen

University of Iceland
ans72@hi.is

Hafsteinn Einarsson

University of Iceland
hafsteinne@hi.is

Abstract

This study evaluates GPT-4’s English-to-Faroese translation capabilities, comparing it with multilingual models on FLORES-200 and Sprotin datasets. We propose a prompt optimization strategy using Semantic Textual Similarity (STS) to improve translation quality. Human evaluation confirms the effectiveness of STS-based few-shot example selection, though automated metrics fail to capture these improvements. Our findings advance LLM applications for low-resource language translation while highlighting the need for better evaluation methods in this context.

1 Introduction

Historically, it has been a challenge to achieve high-quality machine translations (MT) for low-resource languages. The lack of resources has been shown to impact not only the development of high performing MT models, but also the development of high quality automated translation metrics (Callison-Burch et al., 2011; Bojar et al., 2014; Koehn and Knowles, 2017; Ranathunga et al., 2023). Low-resource languages often have to rely on string-based language independent metrics such as BLEU (Papineni et al., 2002) and ChrF (Popović, 2015). However, these methods have shown to perform poorly when compared to neural metrics, as shown by the WMT22 Metrics Shared Task (Freitag et al., 2022). The lack of neural metrics developed for these languages often leaves expensive and slow human evaluation as the only high quality alternative for detecting nuanced improvement in translation quality.

Recent advancements in LLMs offer opportunities to mitigate the effect that low-resources have on translation performance, leveraging few-shot

learning to achieve remarkable performances with minimal data requirements (Brown et al., 2020). However, there is a disparity in the translation performance when it comes to low-resource languages vs high-resource languages (Hendy et al., 2023; Lyu et al., 2023; Bang et al., 2023; Chang et al., 2024). Therefore, optimizing the efficiency of these models in data-constrained environments demands a strategic approach in order to get the best performance. There is still much that is unknown about how prompt engineering and few-shot example selection influences translation performance. Furthermore, LLMs have proven to be a competitive alternative also for what concerns translation evaluation (Kocmi and Federmann, 2023). However, this ability of LLMs to assess translation has not been proven yet in the context of low-resource languages.

We investigate how STS-driven example selection, applied with the translation query, improves translation quality in GPT-4 Turbo (OpenAI et al., 2024), specifically the `gpt-4-1106-preview` release, for Faroese¹, a critically low-resource language. Our findings therefore demonstrate a novel approach to improve the utility of sparse data. Moreover, we demonstrate how current translation metrics cannot adequately capture nuances in translation performance, advocating for the development of more robust evaluation tools. Through this exploration, we provide an assessment of the state of the art of MT for Faroese, and highlight how current automated evaluation metrics cannot appropriately capture nuanced improvements provided by prompt engineering.

Our contributions are the following:

- Creating three synthetic parallel datasets with 1012 sentences each from the FLORES benchmark (NLLB Team et al., 2022),

¹The population of the Faroe Islands is 54.000 (Statistics Faroe Islands, 2024).

translated from English to Faroese by GPT-4 Turbo using zero-shot, random few-shot, and STS-based few-shot techniques respectively².

- Conducting an automated evaluation of these datasets employing BLEU, ChrF, and BERTScore metrics, alongside a GPT-4 Turbo confidence score for the few-shot datasets.
- Ranking of translations from each dataset on a subset of 200 sentences, performed by multiple native Faroese speakers and GPT-4. We further ranked 200 sentences sourced from another dataset to confirm our results.
- Benchmarking GPT-4 Turbo’s English-Faroese translation performance against multilingual translation models covering Faroese such as MADLAD-400 and NLLB-200.

2 Previous Work

2.1 Machine Translation for Faroese

Historically, the limited amount of parallel data available for Faroese has hindered the development of MT tools for the language. However, in recent years, efforts have been made to address this issue and ensure better coverage of Faroese. One such effort led to the creation of the *Sprotin’s parallel corpus* (Mikkelsen, 2021), a collection of around 100K English-Faroese human translated sentences. This corpus facilitated the inclusion of Faroese in *Microsoft Translator* and the development of a Faroese MT model, named *Vélpýðing* (Símonarson et al., 2021), by Miðeind, an Icelandic NLP company. The rise of massively multilingual translation models has sparked several initiatives aimed at including low-resource languages, thanks to their capability for cross-lingual transfer and the exploitation of shared linguistic features. Notably, initiatives such as Google’s MADLAD 400 (Kudugunta et al., 2023) and Meta’s No Language Left Behind (NLLB) (NLLB Team et al., 2022) target specifically low-resource languages, including Faroese. As of July 2024, Faroese is also included in Google Translate, Google’s effort to develop an MT system for over 1,000 languages (Bapna et al.,

2022). The development of these multilingual models still predominantly relies on string-based evaluation metrics like BLEU and ChrF. Despite the widespread criticism and the documented limitations of these metrics (Reiter, 2018; Callison-Burch et al., 2006) they continue to serve as the de facto standard in the field, particularly for low-resource languages, which are for the most part not included in shared tasks aimed at metrics evaluations (Freitag et al., 2022; Mathur et al., 2020). This persistence is likely due to their simplicity, ease of implementation, historical precedent, and, often, lack of affordable alternatives. The recent development of a BERT model for Faroese (Snæbjarnarson et al., 2023) has presented the opportunity to add BERTScore (Zhang et al., 2020), a metric based on contextual embeddings, to the pool of available metrics for Faroese.

2.2 The Rise of LLMs in Machine Translation

With the recent rise of LLMs it became apparent that transformer based MT models are not necessarily the go-to solution anymore when dealing with automatic translation. The few-shot learning capabilities of LLMs opened new avenues for translation with small data. Brown et al. (2020), with their paper titled "Language Models are Few-Shot Learners", demonstrated that GPT-3 could understand and execute tasks, including translation, with minimal examples through in-context learning (ICL). This capacity of LLMs to adapt to specific tasks with just a few guiding examples represents a shift in paradigm from traditional MT methods (Lyu et al., 2024), which often rely on extensive supervised training.

Recently, LLMs have revealed their potential not only as translator but also evaluators of translation (Karpinska and Iyyer, 2023; Fernandes et al., 2023; Huang et al., 2024), reaching state-of-the-art accuracy with respect to human evaluation (Kocmi and Federmann, 2023). However, these results were mostly obtained for high-resource languages, while the potential of LLMs for translating and evaluating translation of low-resource languages remains mostly untapped. In the specific case of Faroese, studies have already been conducted to assess how well LLMs understand the language within the context of MT. Scalvini and Debess (2024) evaluated the language comprehension capabilities of an LLM that targets Nordic

²https://huggingface.co/datasets/barbaroo/FLORES200_translations_GPT4

languages, GPT-SW3, while Debess et al. (2024) and Simonsen and Einarsson (2024) explored GPT-4’s performance in Faroese sentiment analysis and translation from Faroese to English, where it showed good performance.

2.3 The Role of Semantic Textual Similarity in Prompt Engineering.

While some studies have focused on using a zero-shot prompting technique to translate, achieving performance comparable to those of conventional MT systems (Jiao et al., 2023; Chang et al., 2024), the potential of few-shot prompting, particularly in the realm of low-resource languages, invites further exploration. Prior research has predominantly relied on the use of randomly chosen translation examples as prompts. However, emerging studies have explored structured approaches, such as Pattern-Exploiting Training (Schick and Schütze, 2021), K-Nearest-Neighbour (kNN) selection for choosing translation examples from a pool of high-quality candidates (Vilar et al., 2022; Zhu et al., 2023) or choosing examples based on STS (Zhang et al., 2023). Such studies indicate that the quality of translation examples plays a crucial role in the effectiveness of LLMs for MT.

Despite these advancements, the effectiveness of using semantically similar translation examples in MT with LLMs remains an open question. Findings by Vilar et al. (2022) and Zhang et al. (2023) suggest that while example quality is crucial, STS alone does not strongly correlate with improved translation performance. On the other hand, other research, such as the study by Moslem et al. (2023) which utilizes lexical fuzzy matches to find similar translations, points towards significant benefits from employing semantically related examples. It is worth noting that most of this research has focused on high-resource language pairs and previous iterations of LLMs: these results might not therefore directly translate to current LLM versions and low-resource languages. Furthermore, most LLMs are capable of generating grammatically correct output in high-resourced languages, but often fail when zero-shot prompted in languages such as Faroese, making generative language tasks such as translation and summarization challenging. This discrepancy highlights the need for further investigation into the optimal use of example selection strategies in enhancing LLM-based transla-

tion into low-resource languages. Conditioning on grammatically correct and good translation examples has the potential to improve LLM generation quality for low-resourced languages.

3 Methods

3.1 Prompting GPT-4 for Translation

We prompted the GPT-4 Turbo model (`gpt-4-1106-preview`) (OpenAI et al., 2024) for English to Faroese translation in a zero and few-shot setting. This model was selected based on its superior performance in Faroese language generation at the time, as evidenced by preliminary experiments made by the authors of this paper. The prompting strategies used are described below:

- Zero-shot setting.
- Few-shot setting with random selection of 12 parallel sentences from the *Sprotin* dataset (Mikkelsen, 2021). We will refer to such translations as T_{rand} .
- Few-shot setting with selection of 12 parallel sentences from the *Sprotin* dataset based on the highest STS with the translation query (T_{sel}). Note that the translation query is in English, so the similarity search is based on English examples. Their Faroese translated sentences are then used in the few-shot prompt.

The *Sprotin* dataset is, to our knowledge, the largest collection of high quality human translated English-Faroese sentences pairs. STS was quantified by a multilingual model, Multilingual-E5-large (Wang et al., 2024), which was the highest ranking multilingual embedding model at the time according to the MTEB leader board³. The system prompt specified the proficiency of the chat-bot in the Faroese language (*You are an expert in the Faroese language*) and the desired translation quality (*The translations should be of excellent quality*). With these settings, we translated from English to Faroese the test split of the FLORES dataset (NLLB Team et al., 2022), comprised of 1012 sentences.

³<https://huggingface.co/spaces/mteb/leaderboard>

3.2 Comparison with SOTA MT Models

We benchmark GPT-4’s performance against two state of the art multilingual MT models, MADLAD-400 (10B parameters) and NLLB-200 (3B parameters). At the time of writing this paper, the Google Translate API did not allow access to the latest model, covering Faroese. Google Translate was therefore not included in the analysis. The models were used out of the box, without any fine-tuning for the English-Faroese pair, to translate the test split of the FLORES-200 dataset.

3.3 Evaluation on the FLORES-200 Test Set

In order to evaluate and compare translations, several metrics were used: two string-based metrics, BLEU (Papineni et al., 2002) and ChrF score (Popović, 2015), and one neural metric, BERTScore (Zhang et al., 2023), and a human evaluation score. Additionally, GPT-4 was asked to provide a score estimating how confident it was in the translation produced. The BERT model provided to BERTScore for evaluation was, to the best of our knowledge, the only available BERT model specifically catering to Faroese, FoBERT (Snæbjarnarson et al., 2023). We did not find any other neural metric that includes Faroese among its target languages, a situation common to most low and critically low-resource languages. Human evaluation was carried out by three linguists who are native speakers of Faroese. These experts ranked the four Faroese translations - the human translation from FLORES, the zero shot translations, T_{rand} and T_{sel} - blindly from best to worst (1 to 4) (see Figure 1 for an example of the annotation setup in Google Sheets). Annotators were presented with an error type hierarchy to align ranking criteria. According to the hierarchy, sentences with major errors like incomplete translations or lexical errors will be ranked lower than sentences with minor errors such as incorrect inflection or spelling errors. The human evaluation was performed on a subset of 200 translation queries, randomly selected. In this subset, 12 sentences were found which yielded two or more identical translations obtained by different translation methods (zero-shot, T_{rand} , T_{sel} or human reference). These were given the same rank by the annotators⁴. The annotators evaluated the same examples, so that

⁴the ranking could then be 1, 1, 3, 4 if the top ranked sentences are identical or 1, 2, 2, 4 if the second place translations are identical and lastly 1, 2, 3, 3 if the last rankings are identical

inter-annotator agreement could be compared. For comparison, we asked GPT-4 to perform the same ranking task, over the same subset of sentences.

3.4 Replication on Another Source

In order to test the robustness of our human evaluation procedure and its findings, we selected 200 sentences randomly from the Sprotin corpus, and translated them following the three translation strategies presented in Section 3.1, with the aim to reproduce human ranking on this subset. However, the nature of the Sprotin sentences led us to reconsider our strategy: Sprotin is for the most part composed by short, simple, everyday sentences. Such sentences ended up being translated identically across translation strategies, leading to 132 sentences out of 200 having at least two identical translations, 39 having 3 identical sentences, and 21 having all 4 identical entries (three GPT-4 translation strategies plus the human translation). We considered the ranking of these entries a challenging - if not impossible - task, and therefore decided to change selection strategy for test sentences. Preliminary results from this evaluation attempt are discussed in Section 4. Subsequently, we decided to select 200 sentences randomly among longer sentences, as defined by number of tokens in the translation query. The threshold for selection was 18 tokens, as identified by rounding up the average number of tokens in a Sprotin sentence (8.8 tokens) plus 2 standard deviations (8.5). The rationale behind this choice is that longer sentences are more likely to be more linguistically complex and present more opportunities for variation in translation quality. The final subset presented an average of 28.6 tokens, roughly 2 tokens longer than that of FLORES (26.8). This selection thus brought the two subset closer together in terms of average sentence length. These 200 sentences were then translated according to the three different translation strategies, and translations were ranked by two human evaluators from best to worst (1 to 4). Out of 200 translation queries, 4 sentences were not parsed correctly by GPT-4, yielding to incomplete translations. These sentences were excluded from the analysis.

3.5 Annotator Agreement

To assess the degree of agreement among the raters for the ranking tasks, we employed Kendall’s Coefficient of Concordance (W). This

non-parametric statistic is particularly suited for situations where three or more raters are asked to order a set of items, as it measures the extent of agreement among the raters’ rankings (Kendall, 1938). Kendall’s W ranges from 0, indicating no agreement, to 1, denoting perfect concordance.

Each assignment consisted of three Faroese native speakers for FLORES and two native speakers for Sprotin providing rankings for four items. For each assignment, we calculated Kendall’s W to determine the level of rater agreement. We then computed the average of these values across all assignments to obtain an overall measure of agreement. This approach allowed us to quantify the consistency of raters’ evaluations across multiple independent tasks, providing a robust assessment of inter-rater reliability in the context of our study.

4 Results

4.1 Automated Metrics are Blind

Automated metrics such as ChrF, BLEU and BERTScore reveal that GPT-4 produces translations of higher quality with respect to the two MT models, MADLAD-400 and NLLB-200 (see Table 1) on the FLORES dataset. However, when it comes to comparing the different GPT-4 prompting strategies in terms of translation performance, these metrics appear to be "blind" to subtle improvements. By "blind," we mean that the automated metrics are not picking up on the improvement in performance when using the selected method (T_{sel}) over random (T_{rand}) - an improvement that is evident to human evaluators. Statistical comparison between the ChrF, BLEU and BERTScore distributions revealed no statistical difference in translation quality between zero-shot translation, T_{rand} and T_{sel} .

4.2 Human Evaluation on FLORES

Human evaluation revealed a small, but statistically significant difference between T_{rand} and T_{sel} . As Figure 2 shows, human evaluation was consistently ranked first, followed by the STS driven few-shot translation. We aggregated the rankings of T_{rand} and T_{sel} produced by the three evaluators and compared them statistically by Mann–Whitney U test, yielding a p-value of 0.006, indicating that the two distributions are indeed dissimilar. Interestingly, a slightly less substantial difference was found comparing T_{rand} with zero-shot translations ($p = 0.026$), indicating that pro-

viding random examples is a useful approach, but there is still a margin of improvement in translation quality to be exploited by example selection and prompt optimization. To summarize, Table 2 reports how many times (in percentage) each approach received the highest rank. Although the human translation was found to be superior the vast majority of instances, we see that each approach was occasionally ranked first, indicating how nuanced the differences between the approaches can be, when in a low-resource scenario.

4.3 Replication on the Sprotin Subset

As mentioned in Section 3, translating randomly selected sentences from the Sprotin Corpus resulted in many identical translation entries, rendering the set unsuitable for ranking. However, to gain preliminary insight into the performance of the different translation methods on this subset, we counted how often each strategy produced output identical to the human translation. Interestingly, we found that T_{sel} produced the highest number of human-like translations (47), followed by T_{rand} (36) and zero-shot (31). When considering the human reference as the gold standard, these preliminary results mirror the hierarchy observed in the human ranking of the FLORES sentences. The second round of evaluation, concerning the ranking of longer sentences extracted randomly from Sprotin, showed compatible results with our previous findings over the FLORES dataset (Figure 3). We see the human entry being consistently ranked first, obtaining an overall ranking of 1.5, and the zero-shot approach being ranked last overall. The difference between T_{rand} and T_{sel} is however more pronounced in the Sprotin subset than what we observed in FLORES. Statistical comparison of the two distributions yields a p-value of $1.27 e^{-6}$, a strong evidence that the two distributions are in fact distinct, and that the T_{sel} strategy produces statistically better translations. If we take a look at Table 1, displaying how many times each approach was ranked first, in percentage, we find remarkably consistent results between FLORES and Sprotin, a result which supports the robustness of our method and findings.

4.4 Annotator Agreement

The average Kendall’s W value obtained was 0.694 for FLORES and 0.752 for Sprotin, indicating a substantial level of agreement among the raters, which supports the reliability of the ranking

English Sentence	Translation	Evaluation
['Before The Simpsons Simon had worked on several shows in various positions.']	Áðrenn The Simpsons arbeiði Simon við fleiri sýningum í ymiskum störvum.	3
['Before The Simpsons Simon had worked on several shows in various positions.']	Áðrenn The Simpsons starvaðist Simon á ymiskum sjónvarpssendingum í ymsum störvum.	1
['Before The Simpsons Simon had worked on several shows in various positions.']	Áðrenn The Simpsons hövdu Simon arbeiðt við ymiskum sendingum í ymsum störvum.	4
['Before The Simpsons Simon had worked on several shows in various positions.']	Áðrenn The Simpsons hevði Simon arbeiðt við fleiri sendingar í ymiskum starvum.	2

Figure 1: Example of human evaluation setup in a spreadsheet where 4 is the lowest and 1 is the highest rank.

Translation Method	BLEU	ChrF	BERTScore F1
MADLAD-400	13.62 ± 0.53	40.89 ± 0.54	$0.9373 \pm 8 \times 10^{-4}$
NLLB-200	16.79 ± 0.52	48.05 ± 0.39	$0.9474 \pm 5 \times 10^{-4}$
Zero-shot GPT-4	21.36 ± 0.50	52.55 ± 0.39	$0.9516 \pm 5 \times 10^{-4}$
T_{rand} few-shot GPT-4	21.09 ± 0.49	52.36 ± 0.38	$0.9515 \pm 5 \times 10^{-4}$
T_{sel} few-shot GPT-4	21.77 ± 0.50	53.24 ± 0.38	$0.9524 \pm 5 \times 10^{-4}$

Table 1: Translation performance of MADLAD-400, NLLB-200, and GPT-4 on the FLORES-200 dataset for English to Faroese translations.

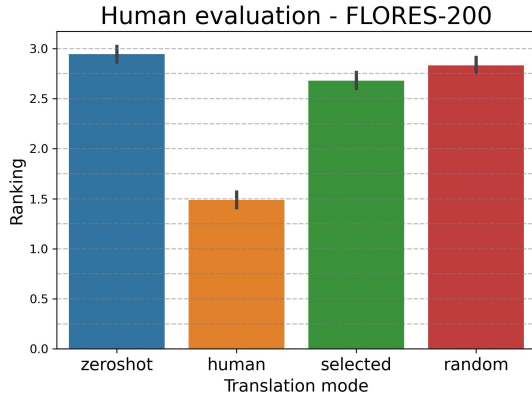


Figure 2: Human evaluation results, for a subset of 200 FLORES sentences. Translations were ranked from best to worst (1 to 4). The T_{rand} (random) and T_{sel} (selected) distributions are statistically different, yielding a p-value of 0.006 by Mann-Whitney U test.

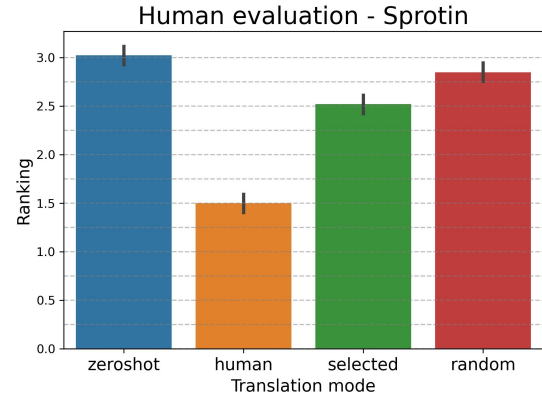


Figure 3: Human evaluation results, for a subset of 200 Sprotin sentences. Translations were ranked from best to worst (1 to 4). The T_{rand} (random) and T_{sel} (selected) distributions are statistically different, yielding a p-value of 1.27×10^{-6} by Mann-Whitney U test.

data used in our analyses.

4.5 GPT-4 is Also Blind

The confidence score provided by GPT-4 was in alignment with human evaluation for what concerns the presence of a statistical difference between T_{rand} and T_{sel} (p value = 1.7×10^{-10}), as can be seen in Figure 5. It is however important to notice how GPT-4 output a confidence score of 0.95 for 93% per cent of translations, a result which is in line with previous findings by Kocmi and Federmann (2023). While these results align with human evaluation, the characteristics of such a dis-

tribution make comparison by statistical analysis less reliable.

To further investigate GPT-4’s understanding of translation nuances, we prompted GPT-4 for translation ranking in a setting that mimics that of human evaluation: the chatbot was asked to rank the 4 translation option from best to worst (1 to 4), on the same set of translated sentences evaluated by human experts. Notably, GPT-4 fails to identify human translation as the best one (Figure 4). Specifically, GPT-4 ranked T_{rand} statistically higher than T_{sel} (p value = 0.026) and human translation (p value = 7.3×10^{-4}). This result there-

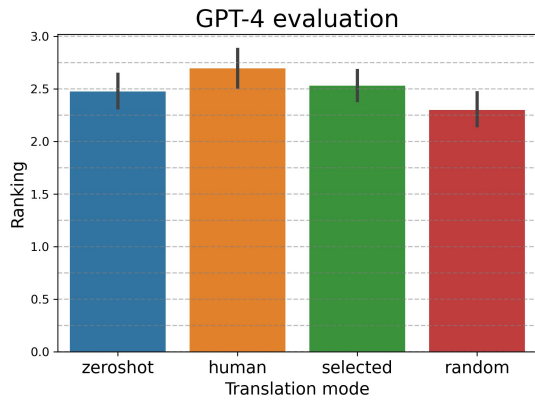


Figure 4: Evaluation rankings assigned by GPT-4, for a subset of 200 FLORES sentences. Translations were ranked from best to worst (1 to 4).

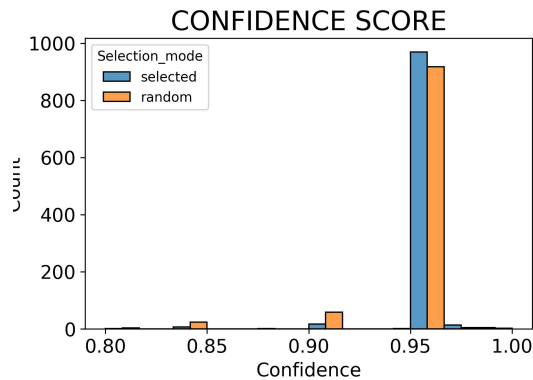


Figure 5: Confidence score assigned by GPT-4 to its own translations of the devtest split of the FLORES-200 dataset. Values for the two distributions are plotted side by side for ease of visualization. The labels 'selected' and 'random' refer respectively to T_{sel} and T_{rand} few-shot translations.

fore shows that GPT-4 is also blind to subtle improvements in translation quality and once again underlines how automated metrics degrade in performance in a low-resource setting.

5 Discussion

5.1 Challenges in Evaluating Low-Resource Language Translation

During our study, we observed how prompt engineering can in fact provide improvements in translation quality into low-resource languages such as Faroese. In order to prove this, we used STS-based few-shot prompting as a proof of concept. While human evaluators were able to detect such improvement, automated scores available for the

language, BLEU, ChrF and BERTScore, failed to do so. That being said, among the automated metrics used, BLEU was most sensitive in detecting the improvement of the selected method (T_{sel}) over the random method (T_{rand}), albeit the difference was small (see Table 1), with overlapping confidence intervals, indicating that it was not able to tell if there was an improvement. In addition to utilizing the above mentioned automated metrics and human evaluation, we also utilize a GPT-4 based confidence score, which is a way to evaluate translation performance from the model’s own perspective. We hypothesize that prompt engineering driven improvements are too nuanced to be detected by currently available automated metrics, including string-based metrics (BLEU, ChrF) and BERTScore. GPT-4’s evaluation also presented critical pitfalls, showing how the model prefers its own output with respect to the human reference. Higher performance automated metrics such as COMET and UNITE (Freitag et al., 2022) are not available for Faroese and for the majority of low-resource languages, as these neural-based metrics require specific resources like large, high-quality datasets for their development. Translation into Faroese and related quality evaluation poses multiple challenges, as Faroese is not only low-resource but also a morphologically rich language. Evaluating MT for morphologically rich languages is notoriously difficult due to the complexity and variability in word forms. These difficulties are well-documented in the literature, with studies highlighting the shortcomings of traditional evaluation metrics when applied to such languages (Freitag et al., 2022). While it is true that LLMs provide new opportunities for low-resource languages, such opportunities cannot be fully taken advantage of for a lack of appropriate methods to assess related improvements. In alignment with statements from Chang et al. (2024) and Sai et al. (2020), our findings highlight how automated metrics do not capture the nuances in quality as human evaluators do. Therefore, we strongly advocate for the development of more robust evaluation tools tailored to low-resource contexts, and in general, for the extension of neural metrics to low-resource languages.

Translation Method	Zero-shot	T_{rand} few-shot	T_{sel} few-shot	Human translation
FLORES - First-Rank (%)	7.83	7.33	11.67	74.33
SPROTIN - First-Rank (%)	7.14	7.65	12.75	74.23

Table 2: Percentage of times the four different translation strategies (human, zero-shot, T_{rand} and T_{sel}) were ranked first during human evaluation. Rankings for all evaluators were aggregated in the final percentage.

5.2 Significance of Semantic Textual Similarity in Few-shot

Our results demonstrate a small yet statistically significant improvement in GPT-4’s translation quality of English to Faroese when using semantically similar examples, as highlighted by human evaluation. This improvement underscores GPT-4’s ability to utilize the context that is provided by semantically similar examples to generate better translations. By using semantically similar examples effectively, our study demonstrates a potential pathway to achieve higher-quality translations without the need for an overly large dataset. Furthermore, We observed a stronger impact of example selection in the Sprotin subset, with respect to FLORES. This might be due to several factors. One possible aspect to consider is the type of language and domains found in FLORES, which are sometimes technical and not representative of every day speech. Therefore, the Sprotin sentences might present a better match to the examples (as they are extracted from the same dataset). Moreover, FLORES is a well known, widely available test dataset for translation, and there is a non negligible possibility of it being already included in GPT-4’s training data. Had the model seen FLORES already, that would limit the impact of the prompting strategy on translation quality. Our findings also contribute to the broader understanding of prompt engineering, specifically in the context of low-resource languages. There is a benefit to selecting STS-based examples. Findings from previous work about the impact of STS are ambiguous (Vilar et al., 2022; Zhang et al., 2023; Moslem et al., 2023). However, they were mostly carried out on high resource languages, for which GPT-4’s performance is generally of high quality. Therefore, we could reasonably expect a smaller margin of improvement, which is harder to detect unambiguously.

5.3 Limitations and Future Works

Our study, while insightful, has certain limitations that pave the way for future research. The focus on a single LLM and language constrains generalizability. Moreover, human evaluation introduces potential biases, particularly in identifying human-written translations. The datasets used lack Faroese cultural elements, and we cannot rule out the possibility of GPT-4 Turbo having been trained on the FLORES dataset. To address these limitations and expand our understanding, future work should explore multiple LLMs, including smaller and domain-specific models, and extend to other low-resource languages. This broader approach could improve the evaluation process and provide insights into the relationship between translation quality and corpus characteristics. Experimenting with an increased number of semantically similar examples and longer paragraphs for translation could enhance quality and offer a more comprehensive evaluation. As open-source models for low-resource languages improve, comparing their performance using our semantic similarity approach could be valuable. Lastly, studying the impact of reference corpus size and domain specificity on STS performance could deepen our understanding in diverse linguistic contexts.

6 Conclusion

This study shows that selecting few-shot learning examples based on STS can improve GPT-4 Turbo’s Faroese translation performance, as confirmed by human evaluation. However, current automated metrics fail to detect these improvements, highlighting a critical issue in low-resource language translation evaluation. While LLMs offer new opportunities for language generation, the inability of automated metrics to capture progress in low-resource contexts could widen digital language representation disparities. This situation necessitates expensive human evaluation, potentially hindering advancements. Therefore, we call

for collaborative efforts to develop metrics specifically designed for low-resource language contexts.

References

- Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. 2023. A Multitask, Multilingual, Multimodal Evaluation of ChatGPT on Reasoning, Hallucination, and Interactivity. *arXiv preprint arXiv:2302.04023*.
- Ankur Bapna, Isaac Caswell, Julia Kreutzer, Orhan Firat, Daan van Esch, Aditya Siddhant, Mengmeng Niu, Pallavi Baljekar, Xavier Garcia, Wolfgang Macherey, Theresa Breiner, Vera Axelrod, Jason Riesa, Yuan Cao, Mia Xu Chen, Klaus Macherey, Maxim Krikun, Pidong Wang, Alexander Gutkin, Apurva Shah, Yanping Huang, Zhifeng Chen, Yonghui Wu, and Macduff Hughes. 2022. Building Machine Translation Systems for the Next Thousand Languages.
- Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, et al. 2014. Findings of the 2014 Workshop on Statistical Machine Translation. In *Proceedings of the ninth workshop on statistical machine translation*, pages 12–58.
- 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. Curran Associates, Inc.
- Chris Callison-Burch, Philipp Koehn, Christof Monz, and Omar F. Zaidan. 2011. Findings of the 2011 Workshop on Statistical Machine Translation. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, WMT ’11, page 22–64, USA. Association for Computational Linguistics.
- Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Re-evaluating the Role of Bleu in Machine Translation Research. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 249–256, Trento, Italy. Association for Computational Linguistics.
- Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2024. A Survey on Evaluation of Large Language Models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45.
- Iben Nyholm Debess, Annika Simonsen, and Hafsteinn Einarsson. 2024. Good or Bad News? Exploring GPT-4 for Sentiment Analysis for Faroese on a Public News Corpora. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7814–7824, Torino, Italia. ELRA and ICCL.
- Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André Martins, Graham Neubig, Ankush Garg, Jonathan Clark, Markus Freitag, and Orhan Firat. 2023. The Devil Is in the Errors: Leveraging Large Language Models for Fine-grained Machine Translation Evaluation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1066–1083, Singapore. Association for Computational Linguistics.
- Markus Freitag, Ricardo Rei, Nitika Mathur, Chiklu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022. Results of WMT22 Metrics Shared Task: Stop Using BLEU – Neural Metrics Are Better and More Robust. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 46–68, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. How Good Are GPT Models at Machine Translation? A Comprehensive Evaluation.
- Xu Huang, Zhirui Zhang, Xiang Geng, Yichao Du, Jiajun Chen, and Shujian Huang. 2024. Lost in the Source Language: How Large Language Models Evaluate the Quality of Machine Translation. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 3546–3562, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.
- Wenxiang Jiao, Wenxuan Wang, Jen-tse Huang, Xing Wang, Shuming Shi, and Zhaopeng Tu. 2023. Is ChatGPT A Good Translator? Yes With GPT-4 As The Engine. *arXiv preprint arXiv:2301.08745*.
- Marzena Karpinska and Mohit Iyyer. 2023. Large Language Models Effectively Leverage Document-level Context for Literary Translation, but Critical Errors Persist. In *Proceedings of the Eighth Conference on Machine Translation*, pages 419–451, Singapore. Association for Computational Linguistics.
- M. G. Kendall. 1938. A New Measure of Rank Correlation. *Biometrika*, 30(1/2):81–93.
- Tom Kocmi and Christian Federmann. 2023. Large Language Models Are State-of-the-Art Evaluators

- of Translation Quality. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 193–203, Tampere, Finland. European Association for Machine Translation.
- Philipp Koehn and Rebecca Knowles. 2017. Six Challenges for Neural Machine Translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39. Association for Computational Linguistics. First Workshop on Neural Machine Translation, NMT 2017 ; Conference date: 04-08-2017 Through 04-08-2017.
- Sneha Kudugunta, Isaac Caswell, Biao Zhang, Xavier Garcia, Christopher A. Choquette-Choo, Katherine Lee, Derrick Xin, Aditya Kusupati, Romi Stella, Ankur Bapna, and Orhan Firat. 2023. MADLAD-400: A Multilingual And Document-Level Large Audited Dataset.
- Chenyang Lyu, Zefeng Du, Jitao Xu, Yitao Duan, Minghao Wu, Teresa Lynn, Alham Fikri Aji, Derek F. Wong, and Longyue Wang. 2024. A Paradigm Shift: The Future of Machine Translation Lies with Large Language Models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 1339–1352, Torino, Italia. ELRA and ICCL.
- Chenyang Lyu, Jitao Xu, and Longyue Wang. 2023. New Trends in Machine Translation using Large Language Models: Case Examples with ChatGPT. *arXiv preprint arXiv:2305.01181*.
- Nitika Mathur, Johnny Wei, Markus Freitag, Qingsong Ma, and Ondřej Bojar. 2020. Results of the WMT20 metrics shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 688–725, Online. Association for Computational Linguistics.
- Jonhard Mikkelsen. 2021. Sprotin sentences. https://raw.githubusercontent.com/Sprotin/translations/main/sentences_en-fo.strict.csv. Accessed: October 13, 2023.
- Yasmin Moslem, Rejwanul Haque, John D Kelleher, and Andy Way. 2023. Adaptive Machine Translation with Large Language Models. *arXiv preprint arXiv:2301.13294*.
- NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No Language Left Behind: Scaling Human-Centered Machine Translation.
- 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 Kopic, 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, Rameev 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 Shepard, 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, Jun-tang Zhuang, William Zhuk, and Barret Zoph. 2024. GPT-4 Technical Report.
- 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.
- Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.
- Surangika Ranathunga, En-Shiun Annie Lee, Marjana Prifti Skenduli, Ravi Shekhar, Mehreen Alam, and Rishemjit Kaur. 2023. Neural Machine Translation for Low-resource Languages: A Survey. *ACM Comput. Surv.*, 55(11).
- Ehud Reiter. 2018. A Structured Review of the Validity of BLEU. *Computational Linguistics*, 44(3):393–401.
- Ananya B. Sai, Akash Kumar Mohankumar, and Mitesh M. Khapra. 2020. A Survey of Evaluation Metrics Used for NLG Systems. ArXiv:2008.12009 [cs].
- Barbara Scalvini and Iben Nyholm Debess. 2024. Evaluating the Potential of Language-family-specific Generative Models for Low-resource Data Augmentation: A Faroese Case Study. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 6496–6503, Torino, Italia. ELRA and ICCL.
- Timo Schick and Hinrich Schütze. 2021. Exploiting Cloze-Questions for Few-Shot Text Classification and Natural Language Inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.
- Haukur Barri Símónarson, Vésteinn Snæbjarnarson, Pétur Orri Ragnarson, Haukur Jónsson, and Vilhjálmur Thorsteinsson. 2021. Miðeind’s WMT 2021 submission. In *Proceedings of the Sixth Conference on Machine Translation*, pages 136–139, Online. Association for Computational Linguistics.
- Annika Simonsen and Hafsteinn Einarsson. 2024. A Human Perspective on GPT-4 Translations: Analysing Faroese to English News and Blog Text Translations. In *Proceedings of the 25th Annual Conference of the European Association for Machine Translation*, volume 1, pages 24–36, Sheffield, United Kingdom. Research And Implementations & Case Studies.
- Vésteinn Snæbjarnarson, Annika Simonsen, Goran Glavaš, and Ivan Vulić. 2023. Transfer to a Low-Resource Language via Close Relatives: The Case Study on Faroese. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 728–737, Tórshavn, Faroe Islands. University of Tartu Library.
- Statistics Faroe Islands. 2024. Population. <https://hagstova.fo/en/population/population/population>. Accessed: May 2024.
- David Vilar, Markus Freitag, Colin Cherry, Jiaming Luo, Viresh Ratnakar, and George Foster. 2022. Prompting paLM for Translation: Assessing Strategies and Performance. *arXiv preprint arXiv:2211.09102*.
- Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Multilingual E5 Text Embeddings: A Technical Report.
- Biao Zhang, Barry Haddow, and Alexandra Birch. 2023. Prompting Large Language Model for Machine Translation: A Case Study. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 41092–41110. PMLR.
- 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.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2023. Multilingual Machine Translation with Large Language Models: Empirical Results and Analysis. *arXiv preprint arXiv:2304.04675*.

Interactive maps for corpus-based dialectology

Yves Scherrer

Dept. of Informatics, University of Oslo
Dept. of Digital Humanities, University of Helsinki

yves.scherrer@ifi.uio.no

Olli Kuparinen

Languages Unit, Tampere University

olli.kuparinen@tuni.fi

Abstract

Traditional data collection methods in dialectology rely on structured surveys, whose results can be easily presented on printed or digital maps. But in recent years, corpora of transcribed dialect speech have become a precious alternative source for data-driven linguistic analysis. For example, topic models can be advantageously used to discover both general dialectal variation patterns and specific linguistic features that are most characteristic for certain dialects. Multilingual (or rather, multilectal) language modeling tasks can also be used to learn speaker-specific embeddings. In connection with this paper, we introduce a website that presents the results of two recent studies in the form of interactive maps, allowing visitors to explore the effects of various parameter settings. The website covers two tasks (topic models and speaker embeddings) and three language areas (Finland, Norway, and German-speaking Switzerland). It is available at <https://www.corcodial.net/>.

1 Introduction

The traditional data collection method in dialectology has relied on structured surveys conducted in a particular language area. The results of such surveys can be presented in maps, typically one map per linguistic feature. These collections of maps, known as dialect atlases, are an important source of information about dialect divisions of different languages. For instance, the dialect atlas of Lauri Kettunen (Kettunen, 1940) still forms the basis of the division of Finnish dialects, even though it was collected almost 100 years ago.

As this example shows, dialect atlases were typically conceived in the first half of the 20th cen-

tury and presented as paper maps. This poses problems of accessibility for modern dialectology, where computational models are often applied on dialect data, e.g. in the subfield of dialectometry (Goebel, 2011). Some atlases have already been digitized and can thus be used in computational analyses (Embleton and Wheeler, 1997; Scherrer and Stoeckle, 2016; Syrjänen et al., 2016). When digitized, the atlases are typically presented as two-dimensional data tables where the columns present linguistic features and the rows locations. Digitized atlases also make interactive visualizations possible (Scherrer, 2023).

In our recent research, we have experimented with topic modeling (Kuparinen and Scherrer, 2024) and representation learning (Kuparinen and Scherrer, 2023) to explore the dialectal divisions arising from corpora instead of atlases. Dialect corpora typically consist of spoken data (mostly interviews) which have been phonetically transcribed. Compared to the straightforward two-dimensional tabular data presented in dialect atlases, corpus data is more difficult to analyze computationally, because individual characteristics of speakers (addressed topics, length of interview, richness of vocabulary, etc.) are mixed with dialect features.

In the following sections, we briefly present the data and experiments, while focusing on the interactive website visualizing the results.

2 Data

We work with three datasets consisting of dialect interviews or conversations, which have been both phonetically transcribed and normalized to a standard variety. The datasets cover the Finnish, Norwegian and Swiss German language areas.

While the topic modeling experiments (Section 3.1) only make use of the phonetic transcriptions, the representation learning study (Section 3.2) is based on a dialect-to-standard normal-

ization task and uses both transcription layers.

2.1 Samples of Spoken Finnish

The Finnish data used in the experiments and visualized on the website comes from the Samples of Spoken Finnish corpus (fi. *Suomen kielten näytteitä*, SKN).¹ The corpus consists of interviews recorded in the 1960s and 1970s in 50 Finnish-speaking locations (Institute for the Languages of Finland, 2021). There are two speakers per location (with one exception) and approximately one hour of speech per person. The interviews were phonetically transcribed by professionals and normalized manually to standard Finnish. In total, the corpus contains 99 interviews and represents traditional Finnish dialects comprehensively.

2.2 Norwegian Dialect Corpus

For Norwegian, we use a subset of the Nordic Dialect Corpus (Johannessen et al., 2009), which contains spoken language data from the North Germanic languages.² The Norwegian part (named Norwegian Dialect Corpus, NDC) is the largest and most thorough in transcription of the different subcorpora. There are 684 interviews (either with a single interviewee or with several) and 438 individual interviewees. For our experiments and visualizations, each data point represents the concatenation of all productions of one interviewee. The recordings were made between 2006 and 2010 and included speakers of different age groups. The recordings were phonetically transcribed and normalized to Bokmål.

2.3 ArchiMob Corpus (Swiss German)

The Swiss German data comes from the ArchiMob corpus (Samardžić et al., 2016; Scherrer et al., 2019), which consists of interviews conducted between 1999 and 2001.³ It contains 43 phonetically transcribed interviews, which are used for the topic modeling experiments. We do not use this corpus for the representation learning experiments, since only six interviews were normalized manually (and the rest automatically).

¹<http://urn.fi/urn:nbn:fi:1b-2021112221>, Licence: CC-BY.

²<http://www.tekstlab.uio.no/scandiasyn/download.html>, Licence: CC BY-NC-SA 4.0.

³<https://www.spur.uzh.ch/en/departments/research/textgroup/ArchiMob.html>, Licence: CC BY-NC-SA 4.0.

3 Experiments

3.1 Topic modeling

The topic modeling experiments are conducted on all three datasets presented in Section 2. We used two topic modeling techniques and five tokenization techniques to explore the dialect divisions of the three focus languages. The used models were non-negative matrix factorization (NMF; Paatero and Tapper 1994) and latent Dirichlet allocation (LDA; Blei et al. 2003), while the tokenizations were complete words, character n-grams from 2 to 4, and Morfessor-based subword tokenization (Smit et al., 2014). A more thorough explanation of the methodology and best results can be found in Kuparinen and Scherrer (2024).

3.2 Representation learning

In the second experiment, we trained a neural machine translation model to “translate” the phonetic transcriptions to standardized spelling. We used a relatively standard setup based on the Transformer architecture (Vaswani et al., 2017) and subword tokenization with BPE (Sennrich et al., 2016). Taking inspiration from multilingual translation modeling (e.g. Johnson et al., 2017), the speaker ID was added as the first token of each utterance on the source side. After training the model, we extracted the learned embeddings of these speaker IDs and used them as input data for three dimensionality reduction algorithms.

The dimensionality reduction algorithms were principal component analysis (PCA; Hotelling 1936), k-means clustering (MacQueen, 1967) and Ward agglomerative clustering (Ward Jr., 1963). The PCA is run with three principal components for visualization purposes (each component represented as a color in the RGB color scheme), while the clustering algorithms are run with the number of clusters ranging from 2 to 20. For further information on the experimental design and a quantitative evaluation of the clustering algorithms, see Kuparinen and Scherrer (2023).

4 Visualization

The website <https://www.corcodial.net/> provides interactive visualizations of the two experiments described in the previous section. The maps are drawn with the Leaflet⁴ mapping toolkit. The map backgrounds use the *Stamen*

⁴<https://leafletjs.com/>

TOPIC MODELING ARCHIMOB - SWISS GERMAN

SKN - FINNISH NDC - NORWEGIAN ARCHIMOB - SWISS GERMAN

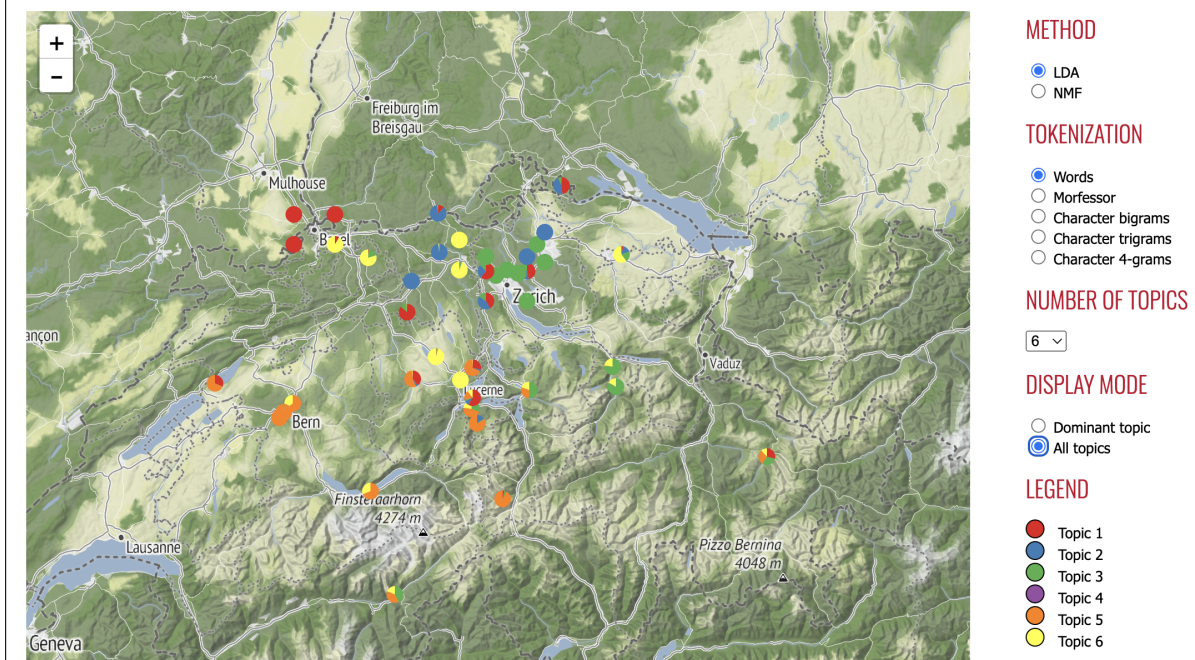


Figure 1: Interactive visualization of a topic modeling experiment for Swiss German. Each point represents one interview. The colored pie charts reflect the degree of membership in the different topics.

terrain style from Stadia Maps,⁵ which are based on OpenStreetMap data.⁶ The server-side backend is implemented in Flask.⁷ All these libraries and sources are licensed under Creative Commons or other open source licenses. The current setup does not require any database, since all the data is available in precomputed CSV or JSON files.

Figure 1 shows a screenshot of a **topic modeling** experiment. The map itself takes up most of the screen, whereas the rightmost part is reserved for user interaction (e.g. to select a different parameter) and metadata display (e.g. the legend associating colors to topics). Each point on the map corresponds to one interview and each color corresponds to one inferred topic. The main benefit of topic models is that an interview can “belong” to several topics to varying degrees. The pie charts on the maps show the degree of membership to the different topics. A simpler visualization that only shows the dominant topic for each interview, is available by selecting *Dominant topic*. Further information about the composition

of the topics (i.e., the tokens most strongly associated with each topic) can be shown in a popup window (not shown in Figure 1).

Technically, and quite similarly to geographic information systems in general, such a visualization relies on two data files: a corpus-specific GeoJSON file that describes the points (with their coordinates and IDs), and a task-specific JSON file that contains the distribution of topics for each point. Leaflet makes it easy to add the GeoJSON file as an additional layer on top of the map background, and to define the style (e.g. the colors) of each point based on the JSON file.

A particularity of corpus-based analyses is that there can be several interviewed persons from the same place, and the corresponding points on the map would be superimposed. The current implementation detects superimposed points and moves them away from their original locations to ensure their visibility. We plan to further improve this functionality.

The **representation learning** experiment is illustrated by Figures 2 and 3. Figure 2 shows a PCA of the speaker embeddings of the Norwegian NDC dataset. As is commonly the case

⁵<https://stadia.com/>

⁶<https://www.openstreetmap.org>

⁷<https://flask.palletsprojects.com/>

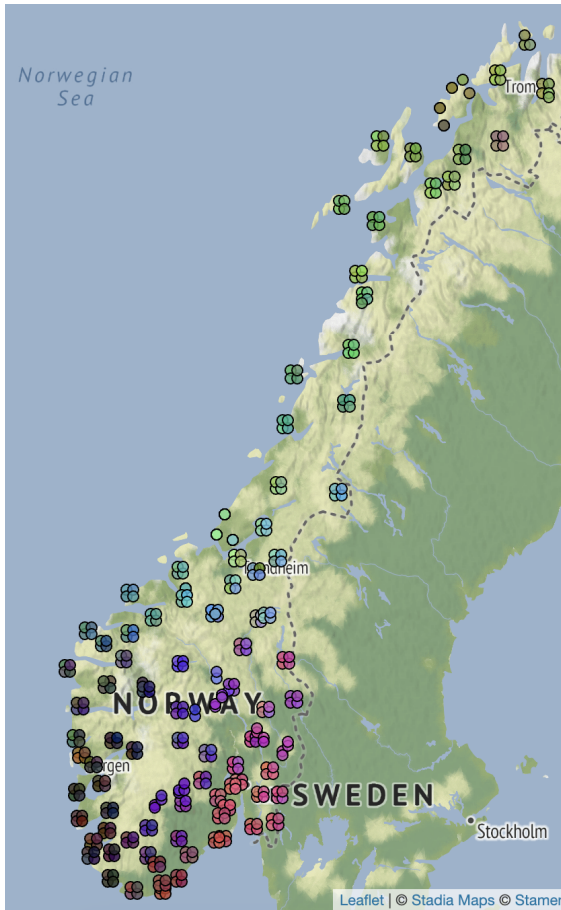


Figure 2: PCA map fragment of the learned speaker representations of the NDC dataset. The three PCA components correspond to the red–green–blue components of the colors.

with dimensionality reduction techniques, the map clearly shows the major dialect areas (Southwestern dialects in dark brown, Eastern dialects in red, central dialects in light blue and northern dialects in light green), without showing clear-cut borders between the areas.

Figure 3, on the other hand, visualizes the speaker embeddings of the Finnish SKN dataset. In this case, a hierarchical clustering algorithm has been selected. The result shows clearly identifiable dialect areas corresponding relatively well with atlas-based divisions.⁸ An exception is the cluster represented in blue on the map, which includes points in the Greater Helsinki area, in a transition area in the Southwest, as well as in Northern Finland. At the moment, the visualization website supports two clustering algorithms (Ward and K-means) and any number of clusters

⁸The dendrogram of the hierarchical clustering can be displayed on demand (not shown here).

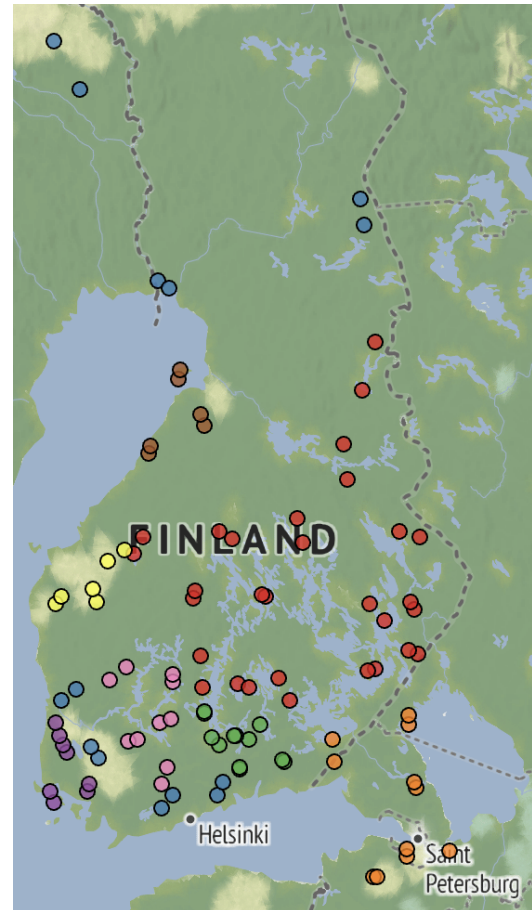


Figure 3: Cluster map of the learned speaker representations of the SKN dataset. The clustering is created with the Ward algorithm and displays 8 clusters.

between 2 and 10.

5 Conclusion

Following up on our recent research where we propose to use topic modeling and representation learning to explore the dialectal divisions arising from corpora of transcribed dialect speech, we present an interactive website where it is possible to view the experimental results in the form of maps. Different parameter settings and modes of visualization can be easily chosen.

At the moment, the website covers two tasks (topic modeling and representation learning) and three linguistic areas (Finland with the SKN corpus, Norway with the NDC corpus, and German-speaking Switzerland with the ArchiMob corpus). The design of the website is modular and permits the easy inclusion of additional tasks, language areas and corpora.

Acknowledgements

This work is supported by the Research Council of Finland through project No. 342859 “CorCoDial – Corpus-based computational dialectology” and project No. 360356 “Speech as speech – acoustic modeling in variational linguistics”.

References

- David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.
- Sheila Embleton and Eric S. Wheeler. 1997. [Finnish dialect atlas for quantitative studies](#). *Journal of Quantitative Linguistics*, 4(1-3):99–102.
- Hans Goebel. 2011. [Dialectometry and quantitative mapping](#). In Alfred Lameli, Roland Kehrein, and Stefan Rabanus, editors, *An International Handbook of Linguistic Variation*, volume 2, pages 433–464. De Gruyter Mouton, Berlin, New York.
- Harold Hotelling. 1936. [Relations between two sets of variates](#). *Biometrika*, 28(3/4):321–377.
- Institute for the Languages of Finland. 2021. [Samples of Spoken Finnish, VRT Version](#).
- Janne Bondi Johannessen, Joel James Priestley, Kristin Hagen, Tor Anders Åfarli, and Øystein Alexander Vangsnes. 2009. [The Nordic Dialect Corpus – an advanced research tool](#). In *Proceedings of the 17th Nordic Conference of Computational Linguistics (NODALIDA 2009)*, pages 73–80, Odense, Denmark. Northern European Association for Language Technology (NEALT).
- Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. [Google’s multilingual neural machine translation system: Enabling zero-shot translation](#). *Transactions of the Association for Computational Linguistics*, 5:339–351.
- Lauri Kettunen. 1940. *Suomen murteet. 3, A, Murrekartasto*. Suomalaisen Kirjallisuuden Seuran toimituksia ; 188. Suomalaisen kirjallisuuden seura, Helsinki.
- Olli Kuparinen and Yves Scherrer. 2023. [Dialect representation learning with neural dialect-to-standard normalization](#). In *Tenth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2023)*, pages 200–212, Dubrovnik, Croatia. Association for Computational Linguistics.
- Olli Kuparinen and Yves Scherrer. 2024. [Corpus-based dialectometry with topic models](#). *Journal of Linguistic Geography*, 12(1):1–12.
- J. MacQueen. 1967. Some methods for classification and analysis of multivariate observations. *Proc. 5th Berkeley Symp. Math. Stat. Probab.*, Univ. Calif. 1965/66, 1, 281–297 (1967).
- Pentti Paatero and Unto Tapper. 1994. [Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values](#). *Environmetrics*, 5(2):111–126.
- Tanja Samardžić, Yves Scherrer, and Elvira Glaser. 2016. [ArchiMob - a corpus of spoken Swiss German](#). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 4061–4066, Portorož, Slovenia. European Language Resources Association (ELRA).
- Yves Scherrer. 2023. [dialektkarten.ch – Interactive dialect maps for German-speaking Switzerland and other European dialect areas](#). In Thomas Krefeld, Stephan Lücke, and Christina Mutter, editors, *Berichte aus der digitalen Geolinguistik (II)*, volume 9 of *Korpus im Text*. Ludwig-Maximilians-Universität München, Germany.
- Yves Scherrer, Tanja Samardžić, and Elvira Glaser. 2019. Digitising Swiss German: how to process and study a polycentric spoken language. *Language Resources and Evaluation*, 53(4):735–769.
- Yves Scherrer and Philipp Stoeckle. 2016. [A quantitative approach to Swiss German – Dialectometric analyses and comparisons of linguistic levels](#). *Dialectologia et Geolinguistica*, 24(1):92–125.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. [Neural machine translation of rare words with subword units](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.
- Peter Smit, Sami Virpioja, Stig-Arne Grönroos, and Mikko Kurimo. 2014. [Morfessor 2.0: Toolkit for statistical morphological segmentation](#). In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 21–24, Gothenburg, Sweden. Association for Computational Linguistics.
- Kaj Syrjänen, Terhi Honkola, Jyri Lehtinen, Antti Leino, and Outi Vesakoski. 2016. [Applying population genetic approaches within languages: Finnish dialects as linguistic populations](#). *Language Dynamics and Change*, 6(2):235 – 283.
- 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.
- Joe H. Ward Jr. 1963. [Hierarchical grouping to optimize an objective function](#). *Journal of the American Statistical Association*, 58(301):236–244.

Profiling Bias in LLMs: Stereotype Dimensions in Contextual Word Embeddings

Carolyn M. Schuster, Maria-Alexandra Dinisor, Shashwat Ghatiwala and Georg Groh

TUM School of Computation, Information and Technology

Technical University of Munich

{carolin.schuster, alexandra.dinisor, shashwat.ghatiwala}@tum.de
grohg@in.tum.de

Abstract

Large language models (LLMs) are the foundation of the current successes of artificial intelligence (AI), however, they are unavoidably biased. To effectively communicate the risks and encourage mitigation efforts these models need adequate and intuitive descriptions of their discriminatory properties, appropriate for all audiences of AI. We suggest bias profiles with respect to stereotype dimensions based on dictionaries from social psychology research. Along these dimensions we investigate gender bias in contextual embeddings, across contexts and layers, and generate stereotype profiles for twelve different LLMs, demonstrating their intuition and use case for exposing and visualizing bias.

1 Introduction

Amongst many other semantic concepts, large language models (LLMs) pick up stereotypes from the data they are trained on. Unbiased data is hard to come by, especially in the amounts needed for the ever-larger models, which are the foundation of the current successes of AI and the respective hype. Thus bias in these models is basically unavoidable, making it necessary to understand its characteristics and extents to communicate the risks and find ways to mitigate adverse discriminatory effects on affected populations.

Past research on bias has often involved word embedding association tests (Caliskan et al., 2017), inspired by the implicit association tests (IAT) (Greenwald et al., 1998) of social psychology. By another inspiration from the social sciences, a newer direction of natural language processing (NLP) research transforms opaque embeddings into a space of meaningful dimensions

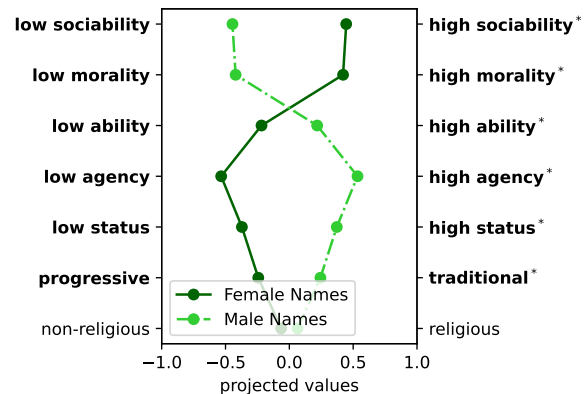


Figure 1: 7D stereotype profile for Llama-3-8B, revealing differences in embeddings of 100 female and 100 male-associated names.

*Statistically significant differences ($p < 0.05$).

(Mathew et al., 2020; Kwak et al., 2021; Şenel et al., 2022; Engler et al., 2022), enabling new ways to study concepts. Similar to semantic differentials (Osgood et al., 1957), this methodology relies on antonyms (e.g. fast vs. slow) or opposing concepts described by lexicons.

In this work we study bias in LLMs by transforming their embeddings based on the stereotype content model (SCM) by Fiske et al. (2002), enabling the study along theoretically and empirically grounded stereotype dimensions. The SCM entails two primary dimensions originating from interactions, where people seek to understand the other party’s intent (dimension of warmth) and their capabilities (dimension of competence). Stereotypically, women are thereby associated with higher warmth, and men with higher competence. Furthermore we employ the extended model (Abele et al., 2016; Ellemers, 2017; Goodwin, 2015; Koch et al., 2016), allowing us to provide detailed 7D bias profiles as shown in Figure 1.

dimension	direction	n	terms	$n_{add.}$	additional terms
sociability	high	43	nice, friendliness, warmth	199	accomodating, witty
	low	42	unfriendly, unsociability, distant	162	acid, withdrawn
morality	high	51	humane, morality, benevolent	205	allegiance, true
	low	69	untrustworthiness, evil, insincere	635	abandon, wrongful
ability	high	40	intelligence, capable, graceful	302	accomplished, ace
	low	39	ignorant, stupid, inefficient	160	awkward, unadvised
agency	high	42	motivated, autonomous, resolute	256	action, worker
	low	39	vulnerable, submission, helpless	113	bowing, unsure
status	high	21	superior, wealth, important	187	advantage, win
	low	13	poor, insignificant, unsuccessful	117	bankrupt, welfare
politics	traditional	12	conventional, conservative	34	classical, capitalist
	progressive	16	modern, liberal, democrat	45	contemporary, feminist
religion	religious	18	believer, church, god-fearing	146	spirit, testament
	non-religious	10	atheist, skeptical, secular	6	unholy, impious

Table 1: Examples of terms and their directions on stereotype dimensions from the theoretically grounded dictionary by Nicolas et al. (2021). The additional terms were collected from their extended dictionary created by a semi-automated method. High-level stereotype dimensions are constructed as follows: warmth = sociability + morality, competence = ability + agency.

Contributions. In summary, we (i) show how the stereotype content model can be employed to expose and visualize bias in contextual embeddings¹, (ii) generate bias profiles for twelve LLMs for gender-associated names and gendered terms, displaying overall stereotypical associations of warmth and competence, (iii) provide insights on stereotype dimensions and gender bias across context examples and network layers.

2 Related Work

Inspired by the human implicit association test (Greenwald et al., 1998), Caliskan et al. (2017) developed the first Word Embedding Association Test (WEAT) to assess the association between two target concepts (e.g., scientist vs. librarian) and two attributes (e.g., male vs. female) in static word embeddings by cosine similarity and a permutation test. Later Tan and Celis (2019) built a first approach to measure bias for LLMs using the contextual embeddings of the words within examples. The Contextual Embedding Association Test (CEAT) (Guo and Caliskan, 2021) employs a random effects model to quantify bias with sampled contexts from a corpus.

A newer approach to interpreting the high-dimensional embedding spaces is again inspired by a concept from the social sciences; seman-

tic differentials (Osgood et al., 1957). Mathew et al. (2020) introduced POLAR, a transformation of static word embeddings to a new polar, interpretable space. The polar opposites are antonyms such as hot-cold or soft-hard, and their word vectors are employed to define the new dimensions, which were shown to align with human judgment in an evaluation study. Similar frameworks are SemAxis (An et al., 2018), FrameAxis (Kwak et al., 2021) and BiImp (Şenel et al., 2022).

More recently, the SensePolar framework was introduced by Engler et al. (2022), extending the POLAR approach to contextual word embeddings. The poles are hereby defined not by the word alone but by their embedding within sense-specific example sentences from a dictionary. The authors showed that these more interpretable embeddings can achieve similar performance to regular ones on natural language understanding (NLU) tasks, and furthermore confirmed the approach by a human evaluation study.

Most similar to our work Fraser et al. (2021) analyzed stereotype dimensions in static embeddings, combining the POLAR framework by Mathew et al. (2020) with the warmth and competence dimensions of the stereotype content model (Fiske et al., 2002). They demonstrated that static word embeddings can recreate the stereotype dimensions from literature by predicting the cold-warm and competent-incompetent associations for

¹Code available at <https://github.com/carolinmschuster/profiling-bias-in-llms>

additional known words, and by further comparing the results to psychological surveys.

For contextual embeddings Ungless et al. (2022) measured bias with CEAT (Guo and Caliskan, 2021) based on the warmth and competence dimensions and in a generation-based approach Jeoung et al. (2023) elicited evaluation of different social groups on these dimensions, with multiple prompting strategies. The stereotype content model has also been used for de-biasing methods (Ungless et al., 2022; Omrani et al., 2023). The researchers suggest that this theory-driven approach has an advantage because it is social-group-agnostic and thus does not require iteration over discriminated groups or previous knowledge of specific bias.

In another projection approach, Omrani Sabbaghi et al. (2023) used a maximum margin support vector classifier to learn the valence subspace (pleasantness vs. unpleasantness) and projected the word ‘person’ to this dimension, placing different words in its context. The bias between words is measured by their effect on the contextualized representation of ‘person’.

This work furthermore relates with a broader range of studies trying to understand the contents of contextual representations, most notably by knowledge probing (e.g. Tenney et al. (2019b); Schuster and Hegelich (2022)). See Cao et al. (2024) for a recent survey.

3 Experimental Setup

3.1 Stereotype Dimensions & Dictionaries

Our analysis of stereotype dimensions and bias in LLMs is grounded in the stereotype content model by Fiske et al. (2002), who showed that there are two major dimensions of warmth and competence and that many stereotypes are mixed along these two. Following Nicolas et al. (2021) we also study the more fine-grained dimensions of sociability and morality for warmth (Abele et al., 2016) and ability and agency for competence (Ellemers, 2017; Goodwin, 2015). Expanding the set of concepts by the Agency-Beliefs-Communion model (Koch et al., 2016), we further include the dimensions of status, politics, and religion. This allows us to also provide a more detailed and extended stereotype profile for the LLMs.

Akin to previous work (Fraser et al., 2021; Omrani et al., 2023) we use the dictionaries in the sup-

plementary data from Nicolas et al. (2021)², which were validated by human evaluation studies, for the construction of the stereotype space.

The ‘seed dictionary’ is theory-driven, using terms from literature, while the ‘full dictionary’ contains additional terms collected by a semi-automated method, identifying synonyms using the English lexical database WordNet (Miller et al., 1990)³. Both dictionaries distinguish seven stereotype dimensions, as shown in Table 1 with examples of terms per dimension and direction. While most dimensions are coded low–high, the politics dimension is coded progressive–traditional and the dimension of religion is coded non-religious–religious.

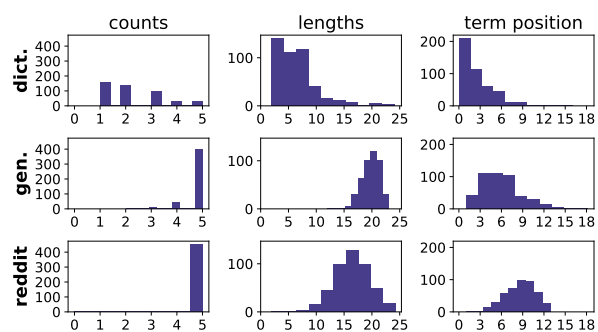


Figure 2: Properties of context examples: Histograms of example counts, numbers of words and positions of dictionary terms within the examples.

3.2 Context Examples

As we are working with contextual embeddings the context of the terms becomes a crucial design choice for the study of stereotype dimensions and bias (see also Engler et al. (2022)).

Generated Examples: For our main experiments, we generate gender non-specific contexts with Llama-3-8B-instruct (AI@Meta, 2024a; Dubey et al., 2024) by the instruction to avoid names and gender-specific pronouns. Thus, no additional gender bias is introduced. As synset information is available for the terms in the seed dictionary, the prompts for these terms additionally include the term definition from WordNet (Miller et al., 1990), allowing a more precise generation for the specific word meaning.

Dictionary Examples: As in the original SensePolar paper (Engler et al., 2022), we also

²<https://osf.io/yx45f/>

³<https://wordnet.princeton.edu/>

retrieve context examples from WordNet (Miller et al., 1990), and only for the seed stereotype dictionary do we manually add examples from other dictionaries where WordNet does not provide any.

Reddit Examples: We include one setup with natural data, where we sample term examples from a Reddit Corpus⁴, similar as done in the Contextual Word Embedding Association Test (Guo and Caliskan, 2021).

No Context: In this setting only the terms are passed through the models, preventing contextualization beyond term subwords.

Example properties are shown in Figure 2. We set the number of examples to five for comparison and to limit computational time, but there often are fewer available for the dictionary examples. Dictionary examples are also the shortest, as they are often short phrases, e.g. “friendly advice”. The generated examples are the longest with an average of 20 words. Reddit examples are truncated on both sides to include context around the term, which may explain later term positions.

3.3 Polar Projection

For the computation of stereotype dimensions, we follow the SensePolar framework by Engler et al. (2022), which is an extension of the POLAR framework (Mathew et al., 2020) for contextual embeddings. Hereby, the embeddings are transformed into an interpretable space based on polar dimensions, which, in our case, are defined by the stereotype content dictionary. We transform the embeddings at two levels: (i) Warmth + competence, and (ii) seven granular dimensions of the extended stereotype content model.

Similar to Fraser et al. (2021), we take the words for each stereotype dimension from the theory-driven ‘seed dictionary’ (Nicolas et al., 2021), and we average individually the word embeddings for the high and for the low classified words, for which the numbers are shown in Table 1. Word lists for warmth (sociability + morality) and competence (ability + agency) are compiled of the words of their subordinate dimensions.

As a first step, we calculate the sense embedding \mathbf{s} for a word with its specific sense and m sense-specific context examples, as shown in Equation 1. We hereby average the contextual embeddings \mathbf{w} across the different context examples,

also averaging across subwords when words are split due to subword tokenization.

$$\mathbf{s} = \frac{1}{m} \sum_{j=1}^m \mathbf{w}_{c_j}^s \quad (1)$$

For n words belonging to the pole of a stereotype dimension, e.g. “friendliness” and “sociability” for the pole “high sociability”, we average their sense embeddings to obtain the average pole embedding \mathbf{p} . Next, we stack the vectors and subtract the low-dimension embeddings from the high-dimension embeddings to obtain the change of basis matrix \mathbf{a} , describing the newly defined space with \mathbf{h} stereotype dimensions:

$$\mathbf{p} = \frac{1}{n} \sum_{i=1}^n \mathbf{s}_i \quad (2)$$

$$\mathbf{a}_h = \mathbf{p}_{h_{high}} - \mathbf{p}_{h_{low}} \quad (3)$$

Regarding the warmth and competence transformation, there are only two direction vectors. If, for example, the original contextual embedding has 768 dimensions, the change of basis matrix \mathbf{a} has a shape of (2, 768).

Before projecting a word of interest to the new dimensions, we compute its embedding \mathbf{x} by again averaging across its context examples as shown in Equation 4. Following prior work (Engler et al., 2022) we then project the embedding to the new interpretable space by the inverted change of basis matrix as shown in Equation 5.

$$\mathbf{x} = \frac{1}{k} \sum_{i=1}^k \mathbf{x}_{c_i} \quad (4)$$

$$\mathbf{d} = (\mathbf{a}^T)^{-1} \mathbf{x} \quad (5)$$

The new embedding \mathbf{d} in the 2D or 7D stereotype space can be interpreted as follows: Similar to the semantic differential technique, a higher value signifies a higher association with the high pole, for example, “high morality”, and a lower value signifies a more significant association with the low pole, e.g., “low morality”. By projecting multiple terms we can compare their differences on these dimensions.

⁴<https://www.kaggle.com/datasets/kaggle/reddit-comments-may-2015>

	Warmth	Competence	Sociability	Morality	Ability	Agency	Status	Politics	Religion
Llama-3-8B (AI@Meta, 2024a)	0.79	0.81	0.66	0.87	0.72	0.77	0.8	0.73	0.74
Llama-3-8B-Instruct (AI@Meta, 2024a)	0.79	0.82	0.65	0.88	0.75	0.77	0.83	0.83	0.81
Llama-3.2-3B (AI@Meta, 2024b)	0.81	0.8	0.63	0.87	0.66	0.76	0.79	0.77	0.62
Llama-3.2-3B-Instruct (AI@Meta, 2024b)	0.78	0.84	0.62	0.88	0.75	0.78	0.81	0.86	0.62
Gemma-2B (TeamGemma et al., 2024a)	0.66	0.67	0.58	0.75	0.66	0.69	0.78	0.58	0.96
Gemma-2-2B (TeamGemma et al., 2024b)	0.7	0.67	0.64	0.8	0.74	0.7	0.75	0.63	0.95
OLMo-1B-hf (Groeneveld et al., 2024)	0.83	0.84	0.76	0.86	0.83	0.77	0.78	0.87	0.6
Bloom-1B7 (Le Scao et al., 2022)	0.79	0.76	0.68	0.87	0.85	0.77	0.62	0.71	0.84
GPT-Neo-125M (Black et al., 2022)	0.66	0.33	0.55	0.75	0.34	0.69	0.41	0.58	0.05
GPT2 (Radford et al., 2019)	0.66	0.67	0.57	0.76	0.53	0.72	0.77	0.6	0.93
ALBERT-base-v2 (Lan et al., 2019)	0.7	0.68	0.62	0.77	0.7	0.69	0.71	0.69	0.65
BERT-base-uncased (Devlin et al., 2019)	0.79	0.83	0.7	0.83	0.8	0.72	0.78	0.72	0.53

Table 2: Accuracy for the direction prediction task. Additional terms in the extended stereotype dictionary (Nicolas et al., 2021) are embedded and projected to the stereotype dimensions. Projected positive/negative values predict high/low direction, e.g. a value of -0.3 for warmth is registered as low warmth. The highest accuracy for each dimension is shown in bold. Please refer to Table 1 for examples of terms with high and low labels for each dimension.

Projection of Additional Terms To evaluate the consistency of the stereotype dimensions we follow the approach by Fraser et al. (2021) and project additional terms from the extended dictionary by Nicolas et al. (2021) to the stereotype space. Hereby, we use the same types of context examples as for the polar space creation. For each term, we predict its direction on its assigned dimension by the sign of its polar value, e.g., a value of -0.5 for sociability is registered as low sociability. To calculate the accuracy, we compare these predictions against the labels in the dictionary.

Projection of Gender-Associated Names & Gendered Terms For the analysis of gender bias, we project gender-associated words to our stereotype dimensions, utilizing two larger binary ‘vocabulary populations’ and individual terms for transgender and nonbinary gender (see Figure 4).

The largest populations are 100 historically female-associated names (e.g., Mary, Patricia) and 100 male-associated names (e.g., James, Michael), taken from the most popular given names of the last century in the United States⁵.

Second, we employ binary gendered terms by definition as utilized in experiments of WEAT (Math vs. Arts and Science vs. Art) (Caliskan et al., 2017). For each gender, we project nine terms:

- Female terms: female, woman, girl, sister, she, daughter, mother, aunt, grandmother
- Male terms: male, man, boy, brother, he, son, father, uncle, grandfather

As examples for our gendered terms and names, we use neutral templates, such as “This is [NAME]” or “This is [TERM]” to provide context without unnecessarily introducing additional bias (compare May et al. (2019); Tan and Celis (2019)). We average across the different templates for a more robust contextual representation of names and terms.

For easier interpretation and comparison between models, we standardize the projected polar values separately for names and terms, as preliminary work showed that named entities and pronouns can show different average tendencies on stereotype dimensions. To assess the significance of the observed differences, we employ t-tests.

3.4 Models

For our evaluation, we project open source models of multiple generations available in the Huggingface Library⁶ onto stereotype dimensions. Model names and references are shown in Table 2. Except for the layer-wise analysis, we extract their average contextual representations across all layers, including the first embedding layer.

⁵<https://www.ssa.gov/oact/babynames/decades/century.html>

⁶<https://huggingface.co/>

4 Results

4.1 Prediction of Direction for Additional Terms

Predicting the direction on the stereotype dimensions for new terms from the extended stereotype dictionary (Nicolas et al., 2021), we find most studied models can perform this task well, with different strengths. Table 2 shows the results when embedding and projecting with the generated examples. OIMo-1B-hf (warmth and competence) and Llama3.2-3B-Instruct (only competence) achieve the performance closest to that of static embeddings by Fraser et al. (2021), where the FastText-based model scored respectively 0.85 for warmth and 0.86 for competence. OIMO and the Llama models also perform very well for the granular dimensions. Predicting morality is the overall easiest task for the models, while the other subdimension of warmth, sociability, poses the most difficult task. Accuracy varies greatly for religion, where there are 142 high, but only 6 low-labeled additional terms.

Only GPT-Neo-125M performs worse than chance for some dimensions, however, this pertains only to raw polar values. When we use a different cut-off than zero for predicting low/high directions by mean-centering the projected values, the model achieves much better results, e.g., 0.76 accuracy for warmth and 0.71 for competence. Similarly, GPT2 and the Gemma models benefit from a mean-based cut-off value, gaining up to 10 percentage points per dimension. Thus projections are spread on different ranges of values, but all models can reasonably discriminate between low and high-labeled terms on the stereotype dimensions.

4.2 Gender Stereotype Profiles

For all twelve studied LLMs, we find statistically significant bias for gender-associated names, as evident in the 2D profiles for warmth and competence in Figure 3. In line with human bias found in studies of the stereotype content model (Fiske et al., 2002), the models highly agree on the relative associations of female names with warmth and male names with competence when using the gender non-specific generated contexts. GPTNeo poses an exception, where both dimensions are associated with female names, and for OIMo, the difference in competence is insignificant.

For the much smaller ‘vocabulary populations’

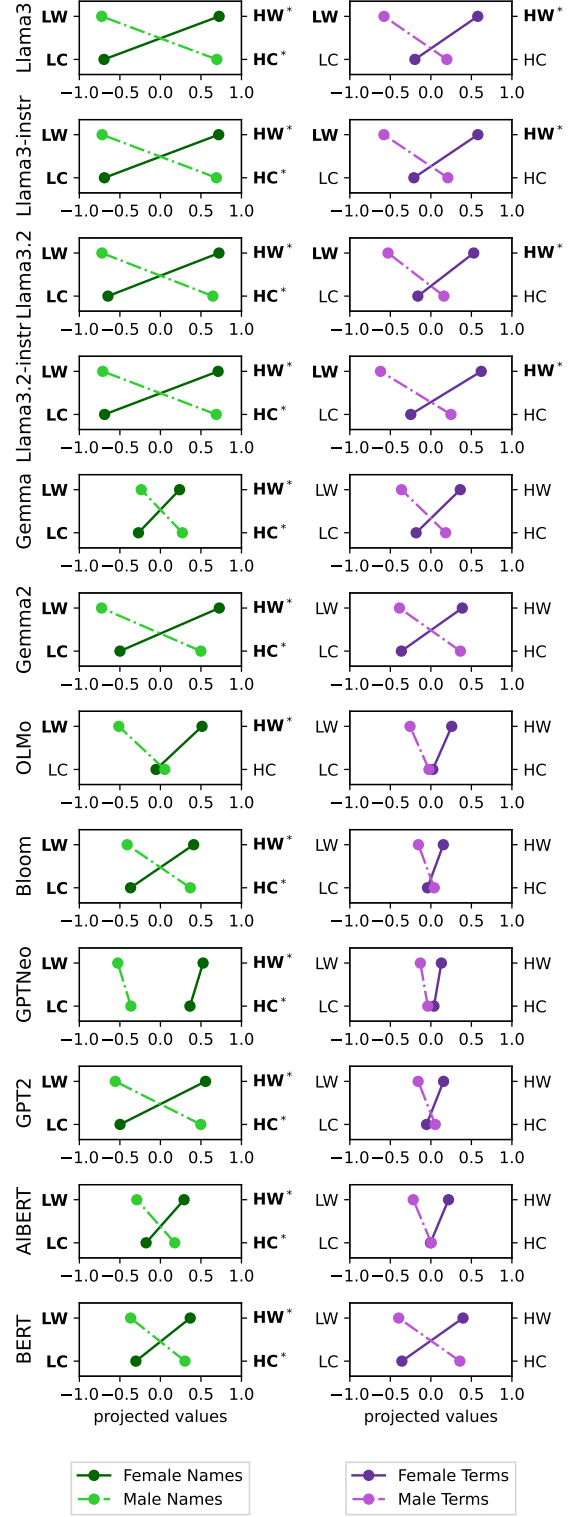


Figure 3: 2D stereotype profiles for 100 female/male-associated names (left) and 9 female/male gendered terms (right). LW/HW = Low/High Warmth. LC/HC = Low/High Competence. *Dimensions with statistically significant differences ($p < 0.05$).

of gendered terms (nine terms per gender), warmth is the more relevant dimension than competence, with only the former being significantly biased for all four Llama-3 models. For some models, e.g. GPT2, the gendered term differences are small, but the bias direction is very consistent when comparing name and term stereotype profiles. In Figure 4, we additionally see the projections of five individual terms beyond binary gender. Nonbinary and transgender-related terms are associated with lower warmth than binary term means, which was found for all newer models (studied variants of Llama3, Gemma, OLMo). For competence, there was no observable trend. For further and statistical analyses of individual terms and small vocabulary populations, future work needs to extend the context examples (as discussed in section 5).

Zooming into the 7-dimensional stereotype profiles for Llama-3-8B (see Figure 1) and Llama-3.2-3B-instruct in Figure 5, we can perceive the previously shown gender associations in more detail. The profiles between Llama-3 and the newer and instruction-tuned 3.2 version are quite similar: Sociability and morality (warmth) are linked with female names, which is true for 10 and 11 of all studied models. Ability and agency (competence) are significantly related with male names, which is true for 9 and 7 of the studied models. Also for status, six models show a significant association with male names. Furthermore six models find male names to be more traditional on the political dimension, while there is no perceivable trend for religion. For the small populations of binary gendered terms, there is only one clearly biased dimension, where seven models agree on a stereotypical association of sociability with female terms.

4.3 Context Examples and Bias across Layers

Comparing the performance for generated, dictionary, Reddit examples, and no context across layers in Figure 6a, we find that the influence of context on the new term prediction task depends on the model. For Llama-3-8B, the accuracy seems quite stable compared to the smaller Gemma-2B variant, where we see high variation across layers and context types. However, as elaborated in subsection 4.1, while the cut-off value of zero works well for most, including all Llama models, Gemma could better discriminate between low/high labeled terms by a different cut-off value.

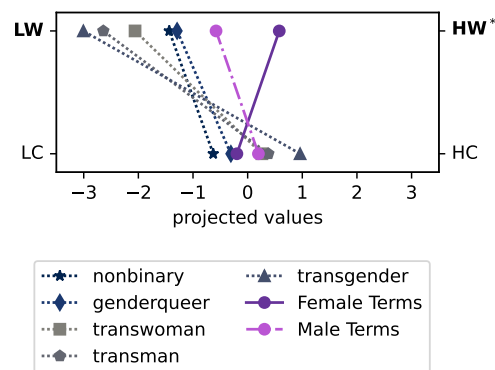


Figure 4: 2D Stereotype profile for Llama-3-8B (see Figure 3) with additional projections of individual nonbinary terms. LW/HW = Low/High Warmth. LC/HC = Low/High Competence.

For GPT2, the Reddit examples lead to much lower accuracy, while for BERT, the no context condition performs markedly worse. Overall the concepts of warmth and competence behave similarly throughout the layers.

On the right in Figure 6b, we see that bias across layers is rather consistent for all models, where higher values signify bias towards female-associated names/terms and lower values signify bias towards male-associated names/terms. Shown by the example of the generated contexts, stereotypical associations permeate throughout the networks. In some cases, the first and last layers behave differently, with a reversed bias direction compared to the overall model.

5 Discussion

Our results provide substantial evidence of stereotype dimensions in the embedding space of LLMs and a gender bias that predominantly corresponds to the human bias found in studies of the stereotype content model (Fiske et al., 2002). For all studied models, female names are relatively associated with higher warmth, and for most models, male names are associated with higher competence. There is less evidence of bias for the studied gendered terms, which in part is likely due to the small groups of only nine terms per gender. The direction of gender differences is, however, overwhelmingly consistent. We furthermore find stereotype dimensions and bias across layers, in line with prior work that semantics are spread throughout the network (Tenney et al., 2019a).

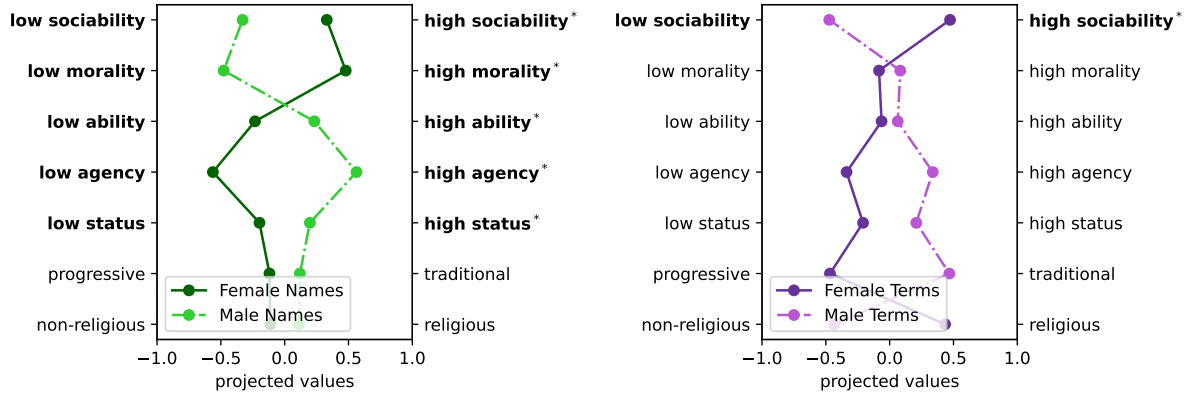


Figure 5: 7D stereotype profiles for 100 female/male-associated names (left) and 9 female/male gendered terms (right) for Llama-3.2-3B-instruct. *Statistically significant differences ($p<0.05$).

The projection of contextual embeddings based on the stereotype content model can deliver robust insights when analyzing large vocabulary groups. As the magnitude of the values depends on the properties of the original embedding space, statistical analysis is employed to assert the significance of gender differences. This is viable with the larger collection of gender-associated names, also providing context for the differences between binary gendered terms. For the analysis of small vocabulary groups or individual terms, e.g. for comparing terms of binary and nonbinary gender, increasing the number of context examples offers potential for statistical tests.

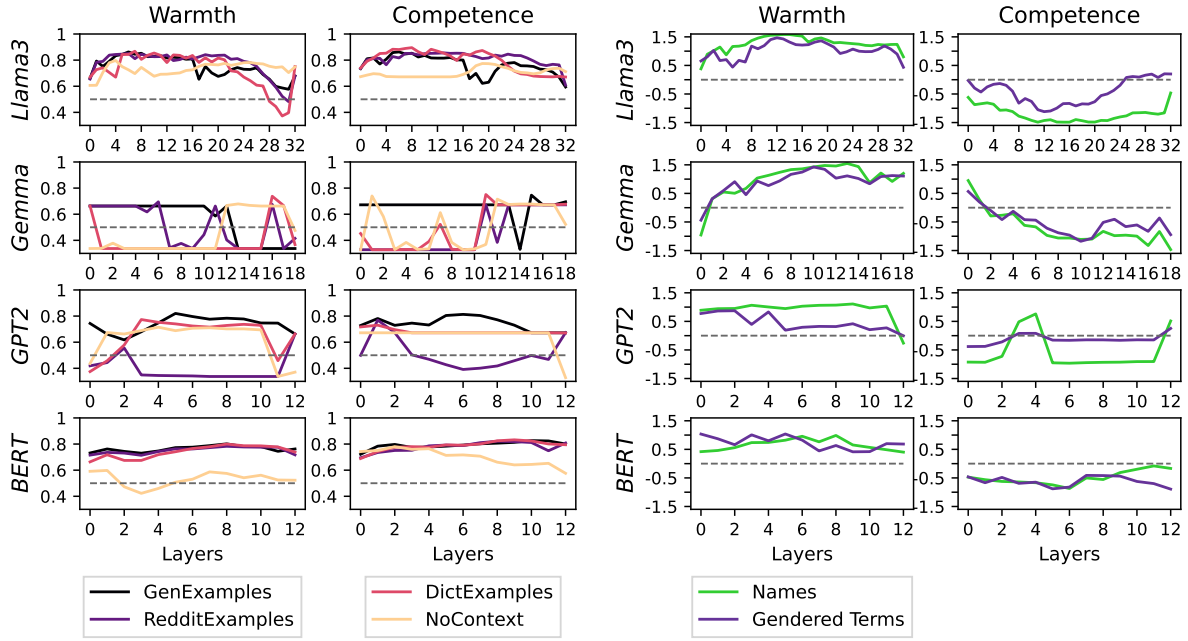
While both 2D and 7D stereotype dimensions provide interesting results, a significant gender bias is most evident in the warmth and competence dimensions. These benefit from the larger numbers of low and high-rated words in the dictionaries, increasing the robustness of the concept representations. Likely associations are also more stable when relating to broader concepts. Therefore, the high-level projection is a suitable first level of analysis and starting point for bias mitigation.

Significant gender bias, however, may also occur on a more granular level. Different dimensions can be relevant depending on domains and tasks, such as progressive-traditional in the realm of politics, and the mode of projection can be easily adapted with the presented methodology. A combined projection with other dimensions such as valence (unpleasantness vs. pleasantness) (see e.g. Omrani Sabbaghi et al. (2023)), could provide even further insights.

As we have shown, the term context can have

a considerable influence on the behavior of the stereotype dimensions. Thus, examples for pole and projected terms should be chosen deliberately. Gender non-specific context is our default choice because no additional bias is introduced through the examples and we get a clearer picture of the bias already present within the pre-trained embeddings. Even smaller open-source LLMs are now able to provide examples with this property at scale. However, measurement is certainly best conducted with domain-specific data when a specific use case exists. While we use simple templates as contexts for the gendered terms and names, these could as well be sampled from a target domain or be generated to test specific scenarios. For example, similar to May et al. (2019), this could involve introducing success in a historically male-dominated field to the term/name context, to test if a penalty exists for females, as found in psychological studies (Heilman et al., 2004).

While embedding-based methods for bias measurement have been critiqued for their remoteness from downstream applications (Gallegos et al., 2024), and are certainly no substitute for task-specific investigations, they have multiple advantages. First, the methodology does not depend on natural language datasets that can be leaked into training data and are therefore applicable to older and newer models alike. Second, the same stereotype dimensions can easily be used for bias mitigation (Ungless et al., 2022; Omrani et al., 2023), alleviating representational harm and the risk that it influences downstream behavior. Finally, our paper shows they can be exploited for intuitive visualizations exposing gender bias.



(a) Accuracy for the direction prediction task across layers, with different context examples. Additional terms are projected to the stereotype dimensions; positive/negative values predict high/low direction.

(b) Gender bias across layers with generated gender non-specific examples. Higher values signify bias towards female-associated names/terms; lower values signify bias towards male-associated names/terms.

Figure 6: Layerwise visualization of prediction accuracy and gender bias for selected models.

6 Conclusion

Large pre-trained language models reflect the biases in their training data, which in turn reflect the biases of their creators. As the foundation for AI applications, their biases are further propagated, warranting their study to uncover the risks and promote mitigation efforts.

In this work, we profile gender stereotypes in twelve LLMs by means of the stereotype content model of social psychology (Fiske et al., 2002), thereby theoretically grounding the analysis, which has in the past been described as the missing link for bias measurements (Blodgett et al., 2020). By a matrix transformation, the opaque contextual embeddings of the models reveal interpretable stereotype dimensions. Along the two major dimensions of warmth and competence, we find significant bias for gender-associated names and some evidence of bias for gendered terms, widely aligned with stereotypes found in human studies.

The shown presence of stereotype dimensions in LLMs is a comprehensible replication of semantics in human language, however, the differential associations of social groups along these dimensions constitute a representational harm.

Equal treatment starts with equal representation; stereotypes already statistically significant in embedding space come with the risk of being exploited in downstream tasks, which could lead to different and unfair treatment of social groups. While the first access point would be the training data itself, the embedding space allows a quantification of patterns that is useful for bias assessment and mitigation. The analysis of embedding spaces by interpretable dimensions provides a means to evaluate both functional semantics and harmful associations that should be ‘unlearned’ to prevent their propagation.

Awareness of bias in LLMs needs to be increased beyond the expert audience, as biased models are already deployed, and completely debiased models may hardly be attainable, as they are trained on vast amounts of biased human-created data. The here presented bias profiles based on the stereotype content model employ an intuitive scoring along meaningful scales of opposing concepts (e.g. low vs. high warmth), as proven effective by semantic differentials in human surveys (Osgood et al., 1957). The result is a highly visual solution for communicating bias to wider audiences and users of artificial intelligence.

7 Limitations

The bias profiles presented in this paper concern only gender, but there is a whole range of biases to be profiled in LLMs to evaluate and communicate representational harms. The scope of analysis was also constrained to English contextual embedding spaces and needs to be extended to a multi-lingual setting in the future.

Furthermore, the focus of this paper was binary gender, with historically gender-associated names and gendered terms. While we projected a few terms for transgender and nonbinary gender to the stereotype dimensions, future analysis needs to extend the methodology for these smaller and diverse ‘vocabulary populations’. Increasing the number of context samples for terms offers potential for greater robustness and applicability of statistical tests.

Finally, no general bias measurement benchmark or method, including the one presented in this paper, precludes the absolute necessity of task-specific bias measurements. However, they can be a piece of the puzzle by revealing learned general bias tendencies and providing a means to mitigate and communicate these effectively.

References

- Andrea E. Abele, Nicole Hauke, Kim Peters, Eva Louvet, Aleksandra Szymkow, and Yanping Duan. 2016. Facets of the fundamental content dimensions: Agency with competence and assertiveness—Communion with warmth and morality. *Frontiers in psychology*, 7:1810.
- AI@Meta. 2024a. Llama 3 model card. URL: https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.
- AI@Meta. 2024b. Llama 3.2 model card. URL: https://github.com/meta-llama/llama-models/blob/main/models/llama3_2/MODEL_CARD.md.
- Jisun An, Haewoon Kwak, and Yong-Yeol Ahn. 2018. SemAxis: A lightweight framework to characterize domain-specific word semantics beyond sentiment. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2450–2461, Melbourne, Australia. Association for Computational Linguistics.
- Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. 2022. Gpt-neo: Large scale autoregressive language modeling with mesh-tensorflow, 2021. URL: <https://doi.org/10.5281/zenodo.5297715>.
- Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. 2020. Language (technology) is power: A critical survey of “bias” in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online. Association for Computational Linguistics.
- Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186.
- Boxi Cao, Hongyu Lin, Xianpei Han, and Le Sun. 2024. The life cycle of knowledge in big language models: A survey. *Machine Intelligence Research*, 21(2):217–238.
- 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.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Naomi Ellemers. 2017. *Morality and the regulation of social behavior: Groups as moral anchors*. Psychology Press.
- Jan Engler, Sandipan Sikdar, Marlene Lutz, and Markus Strohmaier. 2022. SensePOLAR: Word sense aware interpretability for pre-trained contextual word embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4607–4619, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Susan T Fiske, Amy JC Cuddy, Peter Glick, and Jun Xu. 2002. A model of (often mixed) stereotype content: Competence and warmth respectively follow from perceived status and competition. *Journal of Personality and Social Psychology*, 82(6):878–902.
- Kathleen C. Fraser, Isar Nejadgholi, and Svetlana Kiritchenko. 2021. Understanding and countering stereotypes: A computational approach to the stereotype content model. 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 600–616, Online. Association for Computational Linguistics.
- Isabel O Gallegos, Ryan A Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K

- Ahmed. 2024. Bias and fairness in large language models: A survey. *Computational Linguistics*, pages 1–79.
- Geoffrey P. Goodwin. 2015. Moral Character in Person Perception. *Current Directions in Psychological Science*, 24(1):38–44.
- Anthony G. Greenwald, Debbie E. McGhee, and Jordan LK Schwartz. 1998. Measuring individual differences in implicit cognition: the implicit association test. *Journal of personality and social psychology*, 74(6):1464.
- 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 Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muenighoff, 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*.
- Wei Guo and Aylin Caliskan. 2021. Detecting emergent intersectional biases: Contextualized word embeddings contain a distribution of human-like biases. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 122–133.
- Madeline E. Heilman, Aaron S. Wallen, Daniella Fuchs, and Melinda M. Tamkins. 2004. Penalties for success: reactions to women who succeed at male gender-typed tasks. *Journal of applied psychology*, 89(3):416.
- Sullam Jeoung, Yubin Ge, and Jana Diesner. 2023. StereoMap: Quantifying the awareness of human-like stereotypes in large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12236–12256, Singapore. Association for Computational Linguistics.
- Alex Koch, Roland Imhoff, Ron Dotsch, Christian Unkelbach, and Hans Alves. 2016. The ABC of stereotypes about groups: Agency/socioeconomic success, conservative–progressive beliefs, and communion. *Journal of personality and social psychology*, 110(5):675.
- Haewoon Kwak, Jisun An, Elise Jing, and Yong-Yeol Ahn. 2021. FrameAxis: characterizing microframe bias and intensity with word embedding. *PeerJ Computer Science*, 7:e644.
- Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.
- Teven Le Scao, Angela Fan, Christopher Akiki, Elie 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 arXiv:2211.05100*.
- Binny Mathew, Sandipan Sikdar, Florian Lemmerich, and Markus Strohmaier. 2020. The polar framework: Polar opposites enable interpretability of pre-trained word embeddings. In *Proceedings of The Web Conference 2020*, pages 1548–1558.
- Chandler May, Alex Wang, Shikha Bordia, Samuel R. Bowman, and Rachel Rudinger. 2019. On measuring social biases in sentence encoders. 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 622–628, Minneapolis, Minnesota. Association for Computational Linguistics.
- George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. 1990. Introduction to WordNet: An on-line lexical database. *International journal of lexicography*, 3(4):235–244.
- Gandalf Nicolas, Xuechunzi Bai, and Susan T. Fiske. 2021. Comprehensive stereotype content dictionaries using a semi-automated method. *European Journal of Social Psychology*, 51(1):178–196.
- Ali Omrani, Alireza Salkhordeh Ziabari, Charles Yu, Preni Golazizian, Brendan Kennedy, Mohammad Atari, Heng Ji, and Morteza Dehghani. 2023. Social-group-agnostic bias mitigation via the stereotype content model. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4123–4139, Toronto, Canada. Association for Computational Linguistics.
- Shiva Omrani Sabbaghi, Robert Wolfe, and Aylin Caliskan. 2023. Evaluating biased attitude associations of language models in an intersectional context. In *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*, pages 542–553.
- Charles Egerton Osgood, George J. Suci, and Percy H. Tannenbaum. 1957. *The measurement of meaning*. 47. University of Illinois press.
- 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.
- Carolyn M. Schuster and Simon Hegelich. 2022. From BERT’s Point of View: Revealing the Prevailing

- Contextual Differences. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1120–1138, Dublin, Ireland. Association for Computational Linguistics.
- Yi Chern Tan and L. Elisa Celis. 2019. Assessing social and intersectional biases in contextualized word representations. *Advances in neural information processing systems*, 32.
- TeamGemma, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024a. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.
- TeamGemma, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024b. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.
- Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019a. BERT Rediscovered the Classical NLP Pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601. Association for Computational Linguistics.
- Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R Bowman, Dipanjan Das, et al. 2019b. What do you learn from context? probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations*.
- Eddie Ungless, Amy Rafferty, Hrichika Nag, and Björn Ross. 2022. A robust bias mitigation procedure based on the stereotype content model. In *Proceedings of the Fifth Workshop on Natural Language Processing and Computational Social Science (NLP+CSS)*, pages 207–217, Abu Dhabi, UAE. Association for Computational Linguistics.
- Lütfi Kerem Şenel, Furkan Şahinuç, Veysel Yücesoy, Hinrich Schütze, Tolga Çukur, and Aykut Koç. 2022. Learning interpretable word embeddings via bidirectional alignment of dimensions with semantic concepts. *Information Processing & Management*, 59(3):102925.

Entailment Progressions: A Robust Approach to Evaluating Reasoning Within Larger Discourse

Rishabh Shastry

University of Chicago
Chicago, IL, USA

rishabhshastry@uchicago.edu

Patricia Chiril

University of Chicago
Chicago, IL, USA

pchiril@uchicago.edu

Joshua Charney

Morningstar, Inc.
Chicago, IL, USA

josh.charney@morningstar.com

David Uminsky

University of Chicago
Chicago, IL, USA

uminsky@uchicago.edu

Abstract

Textual entailment, or the ability to deduce whether a proposed hypothesis is logically supported by a given premise, has historically been applied to the evaluation of language modelling efficiency in tasks like question answering and text summarization. However, we hypothesize that these zero-shot entailment evaluations can be extended to the task of evaluating discourse within larger textual narratives. In this paper, we propose a simple but effective method that sequentially evaluates changes in textual entailment between sentences within a larger text, in an approach we denote as “*Entailment Progressions*”. These entailment progressions aim to capture the inference relations between sentences as an underlying component capable of distinguishing texts generated from various models and procedures. Our results suggest that entailment progressions can be used to effectively distinguish between machine-generated and human-authored texts across multiple established benchmark corpora and our own EP4MGT dataset. Additionally, our method displays robustness in performance when evaluated on paraphrased texts, a technique that has historically affected the performance of well-established metrics when distinguishing between machine generated and human authored texts.

1 Introduction

As Large Language Models (LLMs) expand and evolve to accommodate more complex language

generation tasks (e.g., significant advances in machine translation (Lai et al., 2023), logical reasoning (Liu et al., 2023), summarization (Zhang et al., 2023), complex question answering (Tan et al., 2023)), we are witnessing a growing number of machine-generated text (MGT) in both online and offline environments.¹ This, in turn, has raised concerns regarding authenticity and regulations,^{2,3} drawing attention to MGT detection as both a safeguard and indicator for authentic human authorship, which has become quite a hot topic in Natural Language Processing (NLP).⁴

Intuitively, machine-generated texts can display lexical, syntactic, and semantic properties that are distinguishable from human authored texts, potentially guiding MGT detection implicitly, as a latent property, or explicitly as a directly encoded feature (Georgiou, 2024). For example, MGT detection methods like entropy and log-likelihood, which assess the probability of a text being machine generated based upon individual token probabilities encoded by a given LLM, take into account how LLMs functionally operate as next word predictors (He et al., 2023). Thus, evaluating where LLMs situationally differ from human authorship in relation to both their observed behaviour and functionality can expand the scope of feature selection within MGT detection to capture these differences more effectively and in a more interpretable manner.

Textual entailment, or the relationship between a given premise and its potentially inferred hy-

¹For a comprehensive overview of LLM capabilities see Guo et al. (2023) and Chang et al. (2024).

²<https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A52021PC0206>

³<https://www.whitehouse.gov/ostp/ai-bill-of-rights/>

⁴For an in-depth analysis of the task, existing corpora and detection methods, see Wu et al. (2023).

pothesis, has been previously used to evaluate how LLM text generation differs from human authorship in regard to an LLM’s ability to generate text in accordance with prior informational constraints (Dagan et al., 2022). In areas like question answering and dialogue systems, calculating the textual entailment between a prior conversation and a machine-generated response can examine whether a model produces relevant and accurate text, a behaviour assumed to be exhibited in human authorship and communication (Ben Abacha and Demner-Fushman, 2019; Dziri et al., 2019). Based on observations of differences in textual entailment between MGTs and human-authored texts in relation to prior conversations, an interesting question arises: *can textual entailment be directly encoded and utilized as a feature for MGT detection?*

In this paper, we:

(1) Introduce *entailment progressions*, a framework in which a given piece of text can be represented as a series of values, with each value representing the level of textual entailment between sentences in a text. These entailment progressions aim to measure the extent to which a model generates each individual utterance in logical reference to its previously generated utterances (i.e., identifying how new information is introduced in relation to the preceding content: *in support*, *in contradiction*, or with *no relation* (neutral)). We believe that entailment progressions provide a unique perspective and should be considered in qualifying LLM behaviour to achieve a more in-depth analysis.

(2) Propose a novel dataset, EP4MGT (Entailment Progressions for Machine Generated Text), comprising 70,158 machine-generated responses across eight state-of-the-art LLMs.⁵

2 Related Work

The definition of recognizing textual entailment (RTE) as outlined by Dagan et al. (2005) and later expanded upon by Korman et al. (2018) is as follows: “a text T textually entails a hypothesis H relative to a group of end users G just in case,

typically, a member of G reading T would be justified in inferring the proposition expressed by H from the proposition expressed by T ”. This definition incorporates three key aspects of RTE. First, it does not require any knowledge beyond the justifiable inference that can be made between a given text and its hypothesis (Feldman, 2003). Second, this justifiable inference is subject to the characteristics exhibited by a group of end users G , in which users outside this group may differ in their inferences due to personal factors that may influence how they interpret logical relationships (Bos and Markert, 2005). Third, the logical component of entailment is textually constrained, rendering it dependent on linguistic factors such as grammar, semantic, and syntactical choices (Braun, 2001).

Current RTE modelling approaches require two main steps. First, the features of premise T and hypothesis H are extracted in order to represent the statements in accordance with relevant linguistic mechanisms associated with textual entailment. Second, the statements are fed into a supervised multi-class classification model which predicts whether a premise-hypothesis pair possesses *positive* (the hypothesis can be inferred to be true if the premise is true), *negative* (the hypothesis can be inferred to be false if the premise is true), or *neutral* (the hypothesis’ truth is not sufficiently conditional upon the premise being true) entailment. For an in-depth overview of RTE resources, approaches, and applications, see Putra et al. (2024).

3 Methodology

3.1 Hypothesis

We incorporate Korman’s RTE approach into the task of detecting MGT under the premise that determining inference relations between sentences in a text is a component of identifying authentic human authorship.

Take, for example, a short story written by ChatGPT. While the story may contain relevant content pertaining to the subject matter and utilize vocabulary similar to its human counterpart, ChatGPT may employ a more simplistic narrative structure without the stylistic nuance or variability typical of human authors. While these LLMs are autoregressive models that generate the next token based

⁵The code and dataset, along with the prompts used for constructing the corpus, are freely available at: <https://github.com/patriChiril/Entailment-Progressions>.

on the previous sequence (without explicitly modelling entailment in the process), our interest lies in exploring whether certain logical patterns are internally captured to some degree.

Regardless of the manner in which these evaluations are conducted, the structure of a textual narrative (like a short story) is an identifiable linguistic feature that can be used to distinguish between texts. We posit that in settings where texts must be logically structured to advance a given claim or narrative purpose, sentence-level evaluation can identify and distinguish structural differences between different generative processes. This process involves examining the inference relations between a new sentence and its overarching premise, as well as between sentences within the text. RTE models can determine the probabilities of entailment, contradiction, and neutrality between a sentence and its preceding text (to identify how the sentence logically corresponds to prior context). These probabilities can be then assembled into “*entailment progressions*”, which are vectors composed of sequentially calculated probabilities of inference relations between a given sentence and the sentences preceding it.

The formal definition of the entailment progressions of a given text can be expressed as follows:

$$EP_{3 \times n} = \begin{bmatrix} c_0 & c_1 & \cdots & c_{n-1} \\ p_0 & p_1 & \cdots & p_{n-1} \\ n_0 & n_1 & \cdots & n_{n-1} \end{bmatrix}$$

where EP is a matrix composed of c, p, n row vectors representing the contradiction, positive, and neutral entailment probabilities between a sentence at a chosen index and its prior sentences in a given text. To compute these values at a given point in a text, we introduce the following equations:

$$EP_{0,i} = C(s_{i+1-w:i+1}, s_{i+1})$$

$$EP_{1,i} = P(s_{i+1-w:i+1}, s_{i+1})$$

$$EP_{2,i} = N(s_{i+1-w:i+1}, s_{i+1})$$

where E represents the model used for calculating entailment between a sentence s at a given point in the text i and the sentences preceding it within a context window of size w .

Motivated by observed discourse phenomena, such as the referential connection between (sum-

marizing) titles and the sentences in their corresponding texts, as well as between sentences in close proximity (Mirkin et al., 2010), entailment progressions use entailment as a heuristic for identifying logical relationships between key components of a text. Given this emphasis on the logical relation between a chosen sentence and its overarching premise (i.e., a title), we also include the following equations:

$$EP_{0,i} = C(p, s_i)$$

$$EP_{1,i} = P(p, s_i)$$

$$EP_{2,i} = N(p, s_i)$$

where E represents the model used for calculating entailment between the general premise defining the full text p and a sentence or collection of sentences s .

Based on our analysis of existing RTE literature, we hypothesize that if the logical relationships between components of a text are distinguishable linguistic features that underlie a set of texts produced by either models or humans, and if entailment progressions effectively represent this set of relationships, then entailment progressions can be used to identify the source of a set of texts. Our hypothesis hinges upon two interconnected inquiries: *Are entailment progressions a meaningful feature of a text?* And, if so, *is the governing structure of these logical relationships reproducible across texts produced by the same author?* We suggest that our hypothesis can be validated by evaluating whether entailment progressions can serve as a feature for identifying and interpreting human authorship. If we can identify MGTs using only their entailment progressions, this would experimentally confirm that they are both meaningful and reproducible features across texts generated through the same procedure.

3.2 Datasets

We conduct our experiments on two freely available English corpora from previous studies and one newly created dataset.

MULTITuDE. This dataset includes 74,081 texts (comprising 7,992 human-written and 66,089 machine-generated texts), distributed across 11 languages (Macko et al., 2023).⁶ The human-

⁶For the purpose of our analysis, we selected only the English subset of the dataset.

written portion of the corpus consists of news articles from the MassiveSumm dataset (Varab and Schluter, 2021). The authors used the titles of the human-written articles for prompting eight different LLMs to generate the corresponding MGTs.

Ghostbuster. This corpus includes both human-authored and ChatGPT-generated text across three domains: creative writing, news, and student essays (Verma et al., 2023). The creative writing collection is sourced from the /r/WritingPrompts subreddit and contains both the original prompts and the corresponding MGT/human-authored texts. The human written collection for the news dataset is based on the Reuters 50-50 authorship identification dataset (Houvardas and Stamatatos, 2006), while the student essay dataset contains high school and university-level essays collected from IvyPanda.⁷

In order to bypass the fixed structure of some of these texts (e.g., news articles), while also covering a diverse set of topics, we build a new dataset, EP4MGT, through which we aim to assess the differences in structure between human-authored and MGTs, specifically within the context of online debates and discussions.

EP4MGT. We draw the human-authored texts from the CMV dataset (Tan et al., 2016), which consists of user interactions from the /r/ChangeMyView subreddit. This Reddit community features posts in which a user presents their original beliefs and rationales, challenging others to contest these viewpoints.⁸ Given a title from the CMV dataset, we task the following LLMs: ChatGPT, GPT4 (Achiam et al., 2023), Gemini (Team et al., 2023), and Mistral (Jiang et al., 2023) (mixtral-8x7b, mistral-7b, mistral-small, mistral-medium, mistral-large) with writing an argument (that could provide compelling reasoning either in favour or against the topic) consisting of at least seven sentences.

It is important to note the varying sentence lengths (and by extension varying word counts) of

the texts included in these corpora. In order to prevent sentence length being a confounding factor in our analysis, we removed both human-authored and machine-generated texts that were outliers in their respective sentence length distributions (e.g., texts containing only one or two sentences, groups of texts that contained fewer than 50 instances of a specific length). The distribution of the sentence counts across the various models in the corpora used in this study is presented in Figure 1, while Table 1 presents an overview of the filtered and unfiltered corpora.

DATASET	MODEL	TOTAL	USED
EP4MGT	GPT4	3,658	3,658
	ChatGPT	10,000	9,928
	gemini-1.0-pro	6,500	5,868
	mistral-7b	10,000	10,000
	mistral-small	10,000	8,663
	mistral-medium	10,000	10,000
	mistral-large	10,000	10,000
	mixtral-8x7b	10,000	10,000
	human-written	10,000	3,864
MULTITuDE	vicuna-13b	3,298	982
	llama-65b	3,288	764
	GPT4	3,300	1,828
	GPT3.5-turbo	3,300	1,262
	text-davinci-003	3,300	1,056
	alpaca-lora-30b	3,297	749
	opt-66b	3,293	755
	opt-1ml-max-30b	3,287	707
	human-written	3,097	1,006
Ghostbuster	claude	1,000	958
	GPT	1,000	920
	GPT-prompt 1	1,000	884
	GPT-prompt 2	1,000	899
	GPT-writing	1,000	910
	GPT-semantic	1,000	955
	human-written	1,000	730

Table 1: Number of machine-generated and human-written texts in the corpora.

3.3 Experimental Design

To ensure that our hypothesis is satisfied, we design an experimental setup that effectively accounts for potential confounding limitations that may arise during analysis.

First, in order to establish a fair comparison between a set of human-authored and machine-generated texts, both sets must “*further the same logical premise*” and pertain to the same language generation task. This effectively controls for style (e.g., news articles, social media discourse, persuasive essays) that could otherwise overemphasize the differences in entailment progressions between human-authored and model-generated texts.

⁷As the authors did not have access to the original news headlines or essay prompts, they used ChatGPT to generate headlines and prompts before creating the corresponding articles and essays.

⁸The dataset can be found at: <https://convokit.cornell.edu/documentation/winning.html>.

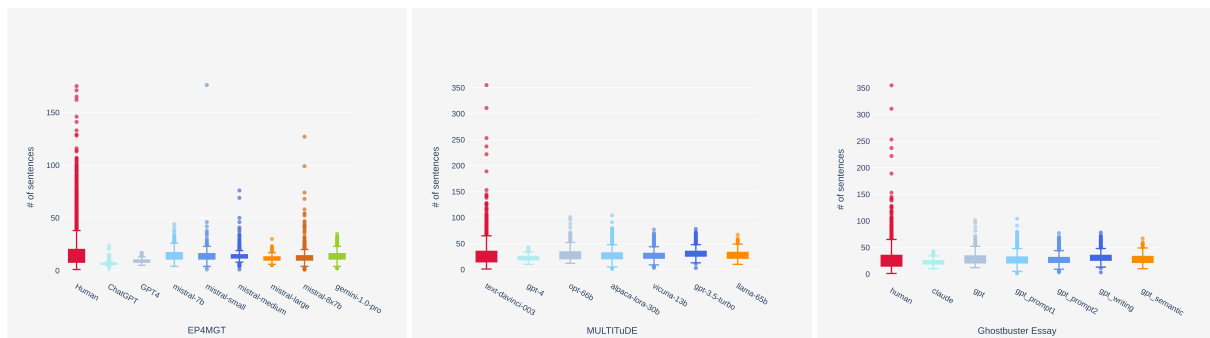


Figure 1: Distribution of number of sentences across the various models in the corpora used in this study.

Second, the texts under examination must be preprocessed in a way that removes any textually confounding identifiers that can further accentuate comparative differences in entailment progressions. This process involves removing any elements within the text that are not relevant to the narrative at hand. These elements include, but are not limited to, the language in which the texts are written, identifiable markers from the media sources (e.g., platforms like Reddit include identifiable tags), and anomalies in sentence length. This helps ensure that the analysis focuses solely on the content of the text.

When controlling for these conditions, we design an experimental setting that is suitable for determining whether entailment progressions can be effectively used as a feature for assessing human and model authorship. This setting involves calculating the entailment progressions for texts from both human-authored and model-generated sets, and then training a classification algorithm to distinguish between the two sources. If the algorithm performs well on the classification task, then we can assume that entailment progressions are a viable feature for differentiating between machine-generated and human-authored texts.

Based on our hypothesis (cf. Section 3.1), we propose two key approaches for constructing the entailment progressions. The first approach (denoted “*Title-Sentence*”) involves calculating the entailment between the general premise of the text and the sentences within the text. This approach assesses the logical relationship between each sentence and the premise it (is attempting to) support. The second approach (denoted “*Sentence-Sentence*”) involves calculating the entailment be-

tween a given sentence and its preceding context. This method uses a sliding context window, examining a given number of sentences (based on the selected window size) directly prior to the evaluated sentence.

In line with the experimental design previously outlined, we generated the *Sentence-Sentence* entailment progressions using context window sizes of 1, 2, and 3 sentences for all datasets. Regarding *Title-Sentence* entailment progressions, as we do not have the general premise for the MULTITUDE and Ghostbuster datasets, we only generate it for the EP4MGT dataset. In this case, the general premise is the title of the original human-authored CMV post, which we used to generate the LLM responses addressing the argument conveyed by the title.

While most of the existing datasets (e.g., SNLI (Bowman et al., 2015), MNLI (Williams et al., 2017)) address the RTE task at sentence-level, logical connections can go beyond consecutive sentences. As such, we rely on DeBERTa pretrained on eight RTE datasets, including DocNLI (Yin et al., 2021), a dataset spanning various lengths for both premises and hypotheses. For performing the experiments, we relied on the HuggingFace transformers library (Wolf et al., 2020).⁹ To test our hypothesis, we trained multi-layer perceptrons (MLPs) with a single hidden layer on these entailment progressions to classify texts within a dataset as either model-generated or human-authored. It is important to note that when assembling the training and testing datasets for the MLP models,

⁹<https://huggingface.co/MoritzLaurer/DeBERTa-v3-base-mnli-fever-docnli-lin-g-2c>



Figure 2: Examples from the EP4MGT dataset displaying low semantic similarity and high entailment progression similarity.

EP4MGT	ENTAILMENT + MLP					MULTITuDE	ENTAILMENT + MLP			GHOSTBUSTER	ENTAILMENT + MLP		
	TITLE-SENTENCE	CONTEXT-1	CONTEXT-2	CONTEXT-3	Δ		CONTEXT-1	CONTEXT-2	CONTEXT-3		CONTEXT-1	CONTEXT-2	CONTEXT-3
GPT4	0.681	0.832	0.896	0.903	-0.046	vicuna-13b	0.786	0.839	0.827	Claude	0.827	0.804	0.776
ChatGPT	0.743	0.892	0.979	0.979	-0.008	llama-65b	0.570	0.659	0.663	GPT3.5-turbo	0.922	0.922	0.911
gemini-1.0-pro	0.681	0.818	0.897	0.902	-0.031	GPT4	0.784	0.857	0.841	GPT3.5-turbo - prompt 1	0.834	0.825	0.837
mistral-7b	0.735	0.825	0.911	0.915	-0.001	GPT3.5-turbo	0.768	0.810	0.811	GPT3.5-turbo - prompt 2	0.909	0.917	0.871
mistral-small	0.695	0.834	0.939	0.940	-0.042	text-davinci-003	0.720	0.704	0.750	GPT3.5-turbo - writing	0.926	0.920	0.921
mistral-medium	0.718	0.869	0.935	0.939	-0.054	alpaca-lora-30b	0.696	0.657	0.669	GPT3.5-turbo - semantic	0.956	0.906	0.902
mistral-large	0.710	0.869	0.932	0.945	-0.015	opt-66b	0.524	0.661	0.690				
mistral-8x7b	0.723	0.845	0.935	0.936	-0.011	opt-1ml-max-30b	0.588	0.768	0.767				

Table 2: Macro F_1 scores for *Title-Sentence* and *Sentence-Sentence* (using context window sizes of 1, 2, and 3 sentences) entailment progressions across the EP4MGT, MULTITuDE, and Ghostbuster corpora.

we only selected entailment progressions that met the same conditions (e.g., *Sentence-Sentence* entailment progressions with a context window size of 2 sentences).¹⁰ Since the entailment progressions vary in length and are sequential, we leveraged a Time Series MLP implementation available through tslearn,¹¹ a Python package dedicated to time series modelling and machine learning.

¹⁰We perform a binary classification task between human-authored texts and texts generated by a specific LLM (e.g., GPT4).

¹¹<https://tinyurl.com/TimeSeriesMLPClassifier>

4 Results and Discussion

In Figure 2 we showcase two MGTs from the EP4MGT dataset. Although these two MGTs are generated by different models (i.e., GPT4 and mistral-large), pertain to different subject matters, and display low textual similarity (0.0718 as calculated using SentenceBERT (Reimers, 2019), a modified BERT that derives semantically sentence embeddings that can be compared using cosine similarity), they exhibit high entailment progression similarity (5.9948 using Dynamic Time Warping distance, that measures the similarity between time series (Müller, 2007))

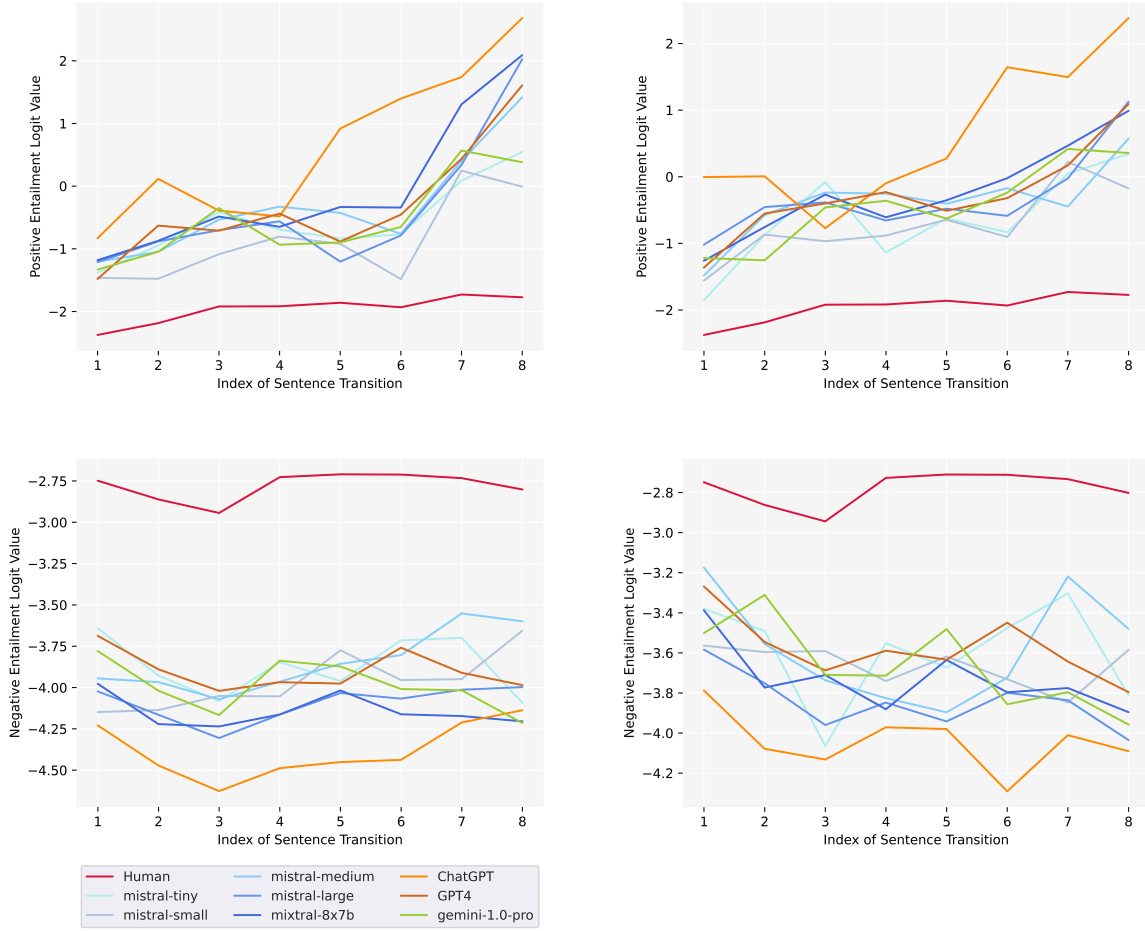


Figure 3: Mean positive (top) and negative (bottom) entailment progressions of texts from EP4MGT dataset before (left) and after (right) paraphrasing.

between each other.

Table 2 highlights the performance of our MLP model when trained solely on various types of entailment progressions across the EP4MGT, MULTITuDE, and Ghostbuster corpora. In our analysis of the two approaches for constructing entailment progressions, we observe that the *Title-Sentence* approach generally underperforms in the EP4MGT dataset. For the EP4MGT dataset, in terms of F_1 score, the performance drop ranges from 13% to 21% when comparing the *Title-Sentence* approach to the *Sentence-Sentence* approach with a one-sentence context window (CONTEXT-1), to two (CONTEXT-2) and three-sentence (CONTEXT-3) context windows, respectively. While the three-sentence context window approach consistently outperforms other entail-

ment progression methods in the EP4MGT dataset, this trend does not hold for the MULTITuDE and Ghostbuster datasets, where the best performing method depends on both the model and the narrative style. Overall, the results show that entailment progressions capture aspects of the evaluated text that can help models (like MLP) to identify human authorship, highlighting the potential insights entailment progressions could provide through further exploration.

Similar to recent work leveraging paraphrasing as a means of evaluating the robustness of different MGT detection approaches (Verma et al., 2023), we also examine the change in performance exhibited by our MLP model when trained on the entailment progressions of the paraphrased texts (where $\Delta = \text{best model } F_1 - \text{best model}$

paraphrased F_1). For this, we leveraged the same methodology as Verma et al. (2023) and Chakraborty et al. (2023), in which each sentence is individually paraphrased using the Pegasus transformer model (Zhang et al., 2020). When trained on the entailment progressions of the paraphrased texts from the EP4MGT dataset, the model exhibits a performance degradation of up to 5% in terms of F_1 score. In addition to these scores, Figure 3 illustrates the changes in between the mean positive and negative entailment progressions for the EP4MGT dataset and their paraphrased counterpart.

5 Conclusion

In this paper, we introduce entailment progressions, a novel representation of the underlying logical structures of textual narratives for identifying human and model authorship. We also present EP4MGT, a dataset specifically designed to evaluate the logical approaches of humans and those produced by a suite of state-of-the-art LLMs, highlighting new avenues for exploring the properties and scope of entailment progressions as a latent descriptor of authorship.

Given that entailment progressions can be generated from any multi-sentence text, their potential applications could extend to the broader area of text attribution, thus providing insights in their utility as identifiers of authorship (be it human or model-based). This would also position our framework alongside more traditional lexical, syntactic, and semantic descriptors of style.

In future work, we plan on examining the effectiveness of entailment progressions in other experimental settings, across different languages, tasks, and genres. Although through our framework we have successfully detected MGTs in several English corpora with fixed narrative structure (i.e., personal claims, news articles), testing entailment progressions on datasets in languages with different underlying logical conventions or within conversational settings (dialogue) with variable logical constraints could reveal broader applicability.

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*.

Asma Ben Abacha and Dina Demner-Fushman. 2019. A question-entailment approach to question answering. *BMC bioinformatics*, 20:1–23.

Johan Bos and Katja Markert. 2005. Recognising textual entailment with logical inference. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 628–635.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

David Braun. 2001. Indexicals.

Megha Chakraborty, SM Tonmoy, SM Zaman, Krish Sharma, Niyar R Barman, Chandan Gupta, Shreya Gautam, Tanay Kumar, Vinija Jain, Aman Chadha, et al. 2023. Counter turing test ct²: Ai-generated text detection is not as easy as you may think—introducing ai detectability index. *arXiv preprint arXiv:2310.05030*.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. 2024. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine learning challenges workshop*, pages 177–190. Springer.

Ido Dagan, Dan Roth, Fabio Zanzotto, and Mark Sammons. 2022. *Recognizing textual entailment: Models and applications*. Springer Nature.

Nouha Dziri, Ehsan Kamalloo, Kory W Mathewson, and Osmar Zaiane. 2019. Evaluating coherence in dialogue systems using entailment. *arXiv preprint arXiv:1904.03371*.

Richard Feldman. 2003. Epistemology. *Tijdschrift Voor Filosofie*, 68(2).

Georgios P Georgiou. 2024. Differentiating between human-written and ai-generated texts using linguistic features automatically extracted from an online computational tool. *arXiv preprint arXiv:2407.03646*.

Zishan Guo, Renren Jin, Chuang Liu, Yufei Huang, Dan Shi, Linhao Yu, Yan Liu, Jiaxuan Li, Bojian Xiong, Deyi Xiong, et al. 2023. Evaluating large language models: A comprehensive survey. *arXiv preprint arXiv:2310.19736*.

- Xinlei He, Xinyue Shen, Zeyuan Chen, Michael Backes, and Yang Zhang. 2023. Mgtbench: Benchmarking machine-generated text detection. *arXiv preprint arXiv:2303.14822*.
- John Houvardas and Efstathios Stamatatos. 2006. N-gram feature selection for authorship identification. In *International conference on artificial intelligence: Methodology, systems, and applications*, pages 77–86. Springer.
- 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*.
- Daniel Z Korman, Eric Mack, Jacob Jett, and Allen H Renear. 2018. Defining textual entailment. *Journal of the Association for Information Science and Technology*, 69(6):763–772.
- Viet Dac Lai, Nghia Trung Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Huu Nguyen. 2023. Chatgpt beyond english: Towards a comprehensive evaluation of large language models in multilingual learning. *arXiv preprint arXiv:2304.05613*.
- Hanmeng Liu, Ruoxi Ning, Zhiyang Teng, Jian Liu, Qiji Zhou, and Yue Zhang. 2023. Evaluating the logical reasoning ability of chatgpt and gpt-4. *arXiv preprint arXiv:2304.03439*.
- Dominik Macko, Robert Moro, Adaku Uchendu, Jason Samuel Lucas, Michiharu Yamashita, Matúš Pikuliak, Ivan Srba, Thai Le, Dongwon Lee, Jakub Simko, et al. 2023. Multitude: Large-scale multilingual machine-generated text detection benchmark. *arXiv preprint arXiv:2310.13606*.
- Shachar Mirkin, Jonathan Berant, Ido Dagan, and Eyal Shnarch. 2010. Recognising entailment within discourse. In *Proceedings of the 23rd international conference on computational linguistics (COLING 2010)*, pages 770–778.
- Meinard Müller. 2007. Dynamic time warping. *Information retrieval for music and motion*, pages 69–84.
- I Made Suwija Putra, Daniel Siahaan, and Ahmad Saikhu. 2024. Recognizing textual entailment: A review of resources, approaches, applications, and challenges. *ICT Express*, 10(1):132–155.
- N Reimers. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.
- Chenhao Tan, Vlad Niculae, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2016. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. In *Proceedings of the 25th International Conference on World Wide Web, WWW '16*. International World Wide Web Conferences Steering Committee.
- Yiming Tan, Dehai Min, Yu Li, Wenbo Li, Nan Hu, Yongrui Chen, and Guilin Qi. 2023. Evaluation of chatgpt as a question answering system for answering complex questions. *arXiv preprint arXiv:2303.07992*.
- 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*.
- Daniel Varab and Natalie Schluter. 2021. Massivesumm: a very large-scale, very multilingual, news summarisation dataset. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10150–10161.
- Vivek Verma, Eve Fleisig, Nicholas Tomlin, and Dan Klein. 2023. Ghostbuster: Detecting text ghost-written by large language models. *arXiv preprint arXiv:2305.15047*.
- Adina Williams, Nikita Nangia, and Samuel R Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*.
- 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.
- Junchao Wu, Shu Yang, Runzhe Zhan, Yulin Yuan, Derek F Wong, and Lidia S Chao. 2023. A survey on llm-generated text detection: Necessity, methods, and future directions. *arXiv preprint arXiv:2310.14724*.
- Wenpeng Yin, Dragomir Radev, and Caiming Xiong. 2021. Docnli: A large-scale dataset for document-level natural language inference. *arXiv preprint arXiv:2106.09449*.
- Haopeng Zhang, Xiao Liu, and Jiawei Zhang. 2023. Extractive summarization via chatgpt for faithful summary generation. *arXiv preprint arXiv:2304.04193*.

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.

Generative AI for Technical Writing: Comparing Human and LLM Assessments of Generated Content

Karen de Souza

Alexandre Nikolaev

Maarit Koponen

University of Eastern Finland University of Eastern Finland University of Eastern Finland
{kpatricki, alexandre.nikolaev, maarit.koponen}@uef.fi

Abstract

Large language models (LLMs) have recently gained significant attention for their capabilities in natural language processing (NLP), particularly generative artificial intelligence (AI). LLMs can also be useful tools for software documentation technical writers. We present an assessment of technical documentation content generated by three different LLMs using retrieval-augmented technology (RAG) with product documentation as a knowledge base. The LLM-generated responses were analyzed in three ways: 1) manual error analysis by a technical writer, 2) automatic assessment using deterministic metrics (BLEU, ROUGE, token overlap), and 3) evaluation of correctness by LLM as a judge. The results of these assessments were compared using a Network Analysis and linear regression models to investigate statistical relationships, model preferences, and the distribution of human and LLM scores. The analyses concluded that human quality evaluation is more related to the LLM correctness judgment than deterministic metrics, even when using different analysis frameworks.

1 Introduction

Technical communication means creating content based on factual data, as consistently and clearly as possible, so that users can easily understand complex technical concepts. Various professionals are involved in it, such as technical translators, developers, information architects, and technical writers (Society for Technical Communication, n.d.). Technical documentation provides specialized and task-oriented information for the user on how to use and interact with a given product. It

is not feasible to cover all possible use cases; instead, the focus should be on the main functionalities or use cases to maintain objectivity. (Swarts, 2018).

LLMs can be useful not only for code generation but also for technical writing because they can simplify the documentation process by generating drafts when prompted with code snippets. This can facilitate the work of technical writers and reduce the effort needed for research. A good model could lower the technical barriers, automate lengthy tasks, and act as an extra solution for their problems (Evtikhiev et al., 2023). However, due to several facts, one of them possibly being outdated training data, LLM-based chatbots can also hallucinate information that does not accurately reflect reality. An alternative to tackle this issue is RAG. It allows external data to be incorporated into the model, which improves its ability to provide more relevant or up-to-date responses, based on the data used to implement the RAG method (Gao et al., 2023).

This study leverages the content produced by a chatbot using multiple LLMs (GPT-3.5 Turbo, GPT-4.0, and Mistral AI 7B) and RAG technology on a specific topic: Network as Code (NaC) technical product documentation. Briefly, NaC simplifies the programming of networks, such as automatically adjusting streaming capabilities and improving bandwidth, for example, for online games or concert streaming (Nokia, 2024).

The LLM responses were evaluated by a technical writer using an error-typology framework and an LLM as a judge based on answer correctness. The aim of this paper is to understand how different (automatic and human) evaluations are distributed according to the attributed scores and how, or if, they relate. Additionally, the ultimate purpose is not to show which evaluation is the best but to offer insights on how performing different analyses, for instance, automatic (quantitative)

and qualitative can complement one another.

We first discuss relevant work on evaluating LLM output quality (Section 2), and then present the dataset and evaluation approaches used in this study (Section 3). We report the comparison of human, deterministic, and LLM evaluations using linear regression models, score distributions (Section 4.1), and Network Analysis (Section 4.2). Finally, we discuss the implications of the comparison (Section 5) and present the conclusions and directions for future work (Section 6).

2 Related work

2.1 Leveraging Translation Quality Evaluation for LLM Analysis

For evaluating the output of LLMs, frameworks and methods have been applied from other contexts, such as (machine) translation quality evaluation. Deterministic metrics, such as Bilingual Evaluation Understudy (BLEU) (Papineni et al., 2002), which are based on the comparison of a “gold-standard” human translation and an automatically generated “hypothesis”, have traditionally been used in the machine translation (MT) field. However, recent advancements in the quality of generative systems have led to increasing skepticism about their reliability, often showing little or no correlation with human assessment (Freitag et al., 2022). BLEU, as well as other metrics such as Recall-Oriented Understudy for Gisting Evaluation (ROUGE), chrF, and token overlap, are also used for evaluating LLM generated output (Zhang and Antonante, 2023).

However, deterministic metrics are not standalone solutions and should be combined with human qualitative evaluation. High-quality and granular standards can again be provided by translation quality evaluation frameworks where professionals manually check and annotate errors according to their severity following the specified error criteria (Fernandes et al., 2023). Error typologies, such as the Dynamic Quality Framework (DQF) and the Multidimensional Quality Metrics (MQM), provide detailed quality criteria to evaluate translations based on categories for accuracy, fluency, style, and design, and attribute point deductions or penalties according to the severity of the error identified (Castilho et al., 2018). The combined DQF-MQM framework defines error categories and sub-categories, penalties, and thresholds that can be adjusted by professional evaluators depend-

ing on their current work needs (Castilho et al., 2018). This framework is commonly used in the field and is considered a reliable methodology for translation quality evaluation. Due to its focus on specific textual features and adjustable categories, it can also be applied to texts independently of their type, such as technical content creation.

2.2 LLMs as quality evaluators

Recent work has also investigated how LLMs can be used as quality evaluators and be prompted (using instructions) to analyze the output quality of MT and generative AI. Kocmi and Federmann (2023a) used a GPT-based evaluation metric called GEMBA-DA for translation quality assessment. Their encoder-only and encoder-decoder language models used supervised data consisting of human gold-standard evaluations in the form of 0 to 100 direct assessments from the WMT22 Metrics shared task. Their approach used zero-shot prompts requesting different LLMs to score each source-target pair on a scale from 0 to 100. The study concluded that this evaluation method was only successful in GPT-3.5 and larger models (Kocmi and Federmann, 2023a). Later work by Kocmi and Federmann (2023b) created an improved model called GEMBA-MQM, which uses GPT-4 to assess translation quality error spans following the MQM framework criteria as a reference. The researchers used a more detailed few-shot prompting technique providing the LLM with the same instructions a human evaluator would receive. The work concludes that the GEMBA-MQM model assessment has higher correlations with human judgment because of a common translation quality evaluation framework. It also compares the LLM scores against deterministic metrics in the scientifically related literature, such as BLEU and chrF, to reach this conclusion (Kocmi and Federmann, 2023b).

These studies indicate that LLMs can provide state-of-the-art quality evaluations that are more correlated with human judgment when they learn or are fine-tuned with human evaluation data, such as the ones produced with the MQM framework (Fernandes et al., 2023). Other approaches, such as those within Continuous-eval packages evaluate LLM-generated text and code with granular or holistic approaches, including quality and deterministic assessment of generative AI content (Zhang and Antonante, 2023). How to evaluate the

quality of generative AI content remains, however, an open question. The present study further explores this question by evaluating texts generated for technical documentation purposes and comparing the evaluations provided by deterministic metrics and an LLM to a manual assessment by a professional technical writer.

3 Material and Methods

3.1 Dataset

The outputs analyzed in this study were generated with Nokia's chatbot tool which is internally used in the company. Nokia authorized its use for this research. This application allows the ingestion of documents, such as Markdown files as input so the chatbot can provide better responses based on the knowledge base provided. RAG can be implemented through technologies such as LangChain, which enables semantic search on relevant content (LangChain, 2023). Further details of this implementation cannot be revealed as the tool is proprietary. The prompts and responses relate strictly to the public documentation of NaC. The dataset includes 12 prompts, each generating 6 different types of responses from three distinct LLMs: GPT-3.5-turbo-16k, GPT-4-1106-preview (OpenAI, 2023) and Mistral AI 7B (Mistral AI, 2023) with two different temperature sets to either 0.4, a more deterministic tone, or 0.7, a more creative one. The chatbot was set to a maximum of 2,048 tokens to avoid overly extensive generated content. The data were collected and analyzed as part of a Master's thesis project in the spring of 2024 (de Souza, 2024).

Zero and few-shot prompt-engineering are the chosen techniques, in which zero shots are simple prompts with no further instructions on how the response should be, and the few-shot ones provide a simple and limited number of examples or instructions for the response (DAIR.AI, n.d.). For instance, in few-shot prompts that requested an LLM to generate a documentation page for a given code snippet about a NaC functionality, instructions were given on how to organize the documentation page in Markdown language, and the main technical concepts to be clarified by the LLM were specified in the prompt.

3.2 Manual evaluation

The responses generated by the different LLMs were analyzed by a professional technical writer

according to the DQF-MQM framework (TAUS, n.d.) using error categories adapted for the purpose of prompt analysis. A discussion of the error analysis is outside of the scope of this paper, but the list of categories and error evaluations can be found in a GitHub repository.¹ A more detailed description is given in de Souza (2024). Based on the errors identified, point deduction penalties were applied according to the severity of the error as follows:

- 0 - no points deducted, in which case the response is correct.
- 0.25 - deducted when errors do not lead to loss of meaning or major confusion.
- 0.5 - deducted when errors are significantly misleading or confusing.
- 0.75 - deducted if errors could affect the company image, e.g. responses that do not include or disregard privacy reminders.

More than one penalty could be applied to the same response if multiple errors were identified in the same response. After the penalties, each response received a total score ranging from 0 (totally irrelevant responses) to 1 (totally correct and relevant responses).

3.3 LLM quality analysis

A qualitative analysis was also performed with the Continuous-eval LLM-based correctness package, which can implement different LLM models to evaluate answer correctness (Relari, 2023). The chosen judge model for this study is GPT-4-1106-preview. The code package allows importing an LLM, which runs through a JSONL file with multiple lines, each containing a set of prompt, response, and related ground truth contexts. The generated responses are evaluated according to their relevance to the prompts, and a total score is given to each response ranging from 0 to 1 as follows:

- 0 indicates the response is totally irrelevant to the prompt.
- 0.25 for responses that are relevant to the prompt but contain major errors.

¹<https://github.com/kjp-souza/tech-writing-LLM-human-evaluation>

- 0.5 for responses that are relevant to the prompt but are partially correct.
- 0.75 is attributed to responses that are relevant to the prompt and correct.
- 1.0 is for responses that are relevant to the prompt and complete.

3.4 Deterministic metrics

The human and LLM quality analyses were compared against deterministic metrics, which evaluate the tokens in both prompt and response sets to leverage their token similarity. This deterministic analysis was done using the deterministic metrics also within the Continuous-eval package and are described as follows (Relari, 2023):

- Token overlap refers to words shared by both sets of texts (ground truth reference and generated response).
- ROUGE-L calculates the longest shared subsequence between the generated response and the ground truth text used as reference.
- BLEU measures how well a generated text matches a reference text using n-gram precision, where each n-gram has a specific weight and applies a brevity penalty to overly short translations.

3.5 Network Analysis

The different analysis results, including the manual human quality evaluation scores and automatically generated deterministic and LLM-based correctness scores, were compared in the R statistical software (R Core Team, 2024). Four different types of analyses were created: Network Analysis, linear regression models, score distribution, and average plots.

The Network Analysis used the `bootnet` package (Epskamp et al., 2018), which allows for estimating the network structure based on the observed data. To obtain a conservative network model, it was necessary to apply the least absolute shrinkage and selection operator (LASSO) method. This helped identify only the most important edges (relationships) in the network, formed by nodes, which contain a descriptive label for the deterministic metrics, human quality evaluation,

and LLM correctness evaluation scores. By doing so, over-fitting is avoided so the model can remain interpretable. In other words, Network Analysis edges can capture how changes in one variable relate to changes in another without putting a single one in evidence. Furthermore, a tuning parameter value of 0.5 was chosen for the Extended Bayesian Information Criterion (EBIC), which helps balance model complexity and goodness of fit. A smaller EBIC value indicates a better-fitting model, in this case, the tuning offers a moderate level of regularization, which penalizes very weak connections (edges) in a sparse network (Nikolaev and Bermel, 2022). Once the network was estimated, a threshold was applied (setting the option to true) to remove weak associations based on correlation strength, leaving only meaningful connections in a network that becomes easier to interpret (Nikolaev and Bermel, 2022).

However, to observe if there is any LLM preference in common between human and LLM analysis, linear regression models were created in R programming language and these use dependent variables: Human quality and LLM correctness judgments against multiple independent variables which indicate different LLMs (GPT-3.5, GPT-4 and Mistral). The `sjPlot` library (Lüdecke, 2024) was used to plot and better visualize the relationship between human and LLM evaluation scores and model types (Nikolaev and Bermel, 2022). Additionally, a figure visually representing the distribution of human and LLM scores was created using the `ggplot2` R package (Wickham, 2016a), which is part of the `tidyverse` ecosystem (Wickham et al., 2019) and used to design graphics according to the grammar of graphics approach (Wickham, 2016b).

4 Results

4.1 Comparison of human and LLM scores

Figure 1 shows the distribution of scores, detailed in sections 3.2 and 3.3, according to human quality evaluation and LLM correctness judgment. The Y axes contain the response counts, visually representing the distribution, while the X axes represent the score distributions according to both human and LLM judgment columns. The human-evaluation scores plot has several peaks and is more evenly distributed, while the LLM one has 2 major peaks with very few outliers, in which responses got 0 or 1 scores and no 0.5 scores.

This shows that the human evaluation scores varied more on a scale from 0 to 1 while the LLM almost exclusively assigned the scores 0.25 or 0.75 and did not use the score 0.5 at all. Additionally, figure 8 in the appendix A shows the average scores for automatic metrics and human quality evaluation (Human QE), which were also generated using the same R programming language packages.

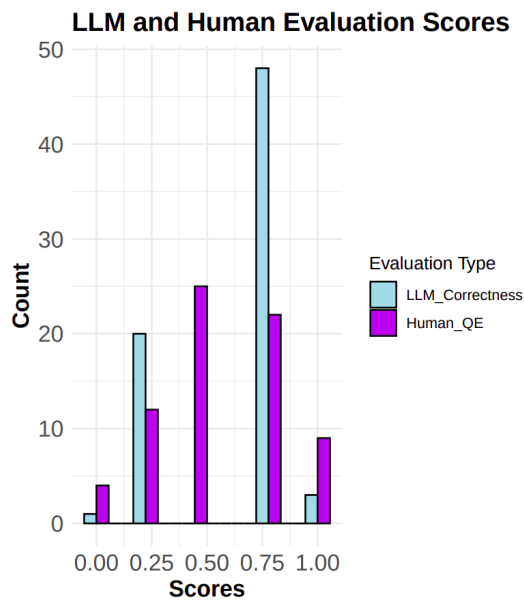


Figure 1: LLM and human scores distribution

Linear regression models were used to analyze the relationships between the dependent variables, human and LLM correctness judgments, and the independent variables, which included multiple language models such as GPT-3.5, GPT-4, and Mistral. We found this approach to be particularly well-suited for handling categorical independent variables, offering significant advantages in terms of flexibility and interpretability. Specifically, it allows the quantification of effect sizes of each language model relative to a designated baseline, either human or LLM judgment, offering deeper insights beyond simple comparisons of group means. Furthermore, the framework accommodates extensions such as continuous predictors and interaction terms, ensuring versatility in our analysis. These features align seamlessly with our research objectives, enabling a nuanced and comprehensive interpretation of the relationships between predictors and outcomes.

Figures 2 and 3 illustrate how well an LLM can predict or explain the quality evaluations by human and LLM (GPT-4-1106-preview) evalua-

tors. The dependent variables are human quality evaluation scores and LLM correctness judgments, while the independent variables represent different LLMs (GPT-3.5, GPT-4, and Mistral), showcasing their influence on human and LLM assessments and enabling a more precise comparison of alignment and similarities in judgment.

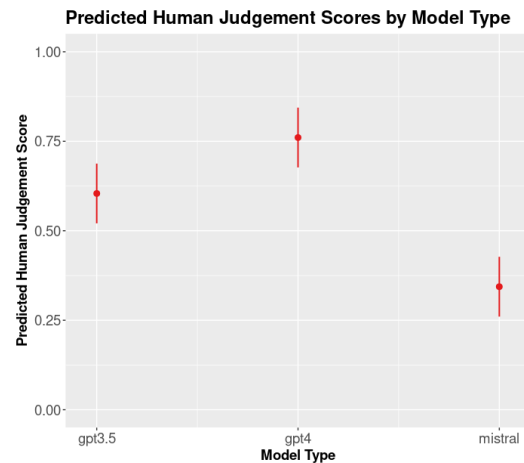


Figure 2: Human analysis linear regression model

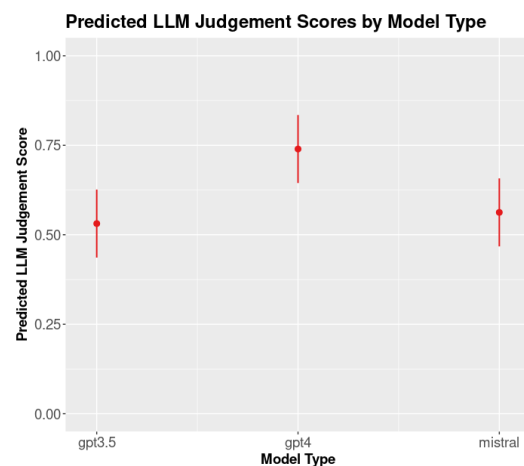


Figure 3: LLM analysis linear regression model

Both figures show that human and LLM evaluations attributed higher or better scores, which surpass 0.75 on a scale from 0 to 1, to content generated by the GPT-4 model. However, GPT-3.5 was better evaluated in the human analysis than in the LLM one. The scores did not drop below the average of 0.50 for GPT-3.5 in human analysis, while they did in the LLM evaluation scores. Mistral models received scores below average in both analyses. However, in LLM analysis, Mistral's scores exceeded 0.60 points while in the human one, it did not even reach 0.50 on average. In conclusion, GPT-4 model responses were better evaluated by both human and LLM analyses, the

GPT-3.5 model was better evaluated only in human analysis and Mistral AI responses were better evaluated only by the LLM one.

4.2 Network Analysis results

As there is no dependent variable in a Network Analysis, it is possible to observe different positive or negative relationships, which indicate behaviors among multiple variables, represented by the evaluation scores, without emphasizing a single one. Figure 4 shows a Network Analysis of all observed variables: qualitative, including human and LLM, and multiple deterministic metrics. It consists of nodes (independent variables) that are connected by edges (statistical relationships). These variables and edges are connected in a spiral format representing statistical relationships between them. The goal is to understand how these variables interact with each other. The code and logic used here follow the same ones used in Nikolaev and Bermel (2022).

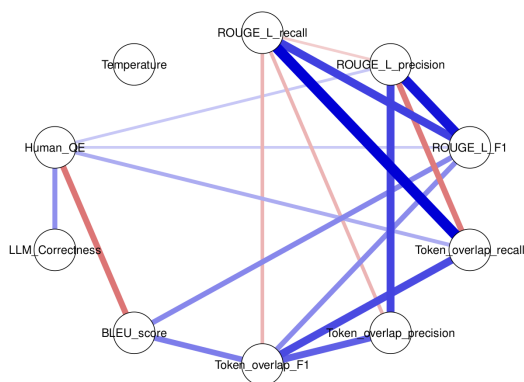


Figure 4: Correlation Network Analysis

The edges in the network (shown as lines) explain the co-variation structure among the observed variables. The blue edges indicate a positive correlation, while the red edges signify a negative one. The color intensity reflects relationship strength. Additionally, this method allows observing the relations between human versus LLM evaluation on one side, and the deterministic correctness metrics relation on the other.

Figure 4 shows that human and LLM evaluations are positively correlated, though the strength of the association is weak to moderate, which can be observed not only in the network but also in the calculated strength of association (0.27). A weight near 0.30 suggests a weak to moderate relationship, while values closer to 1 indicate a stronger association. On the deterministic side, human evaluation is in a weak positive association with

the metrics token-overlap recall (0.20), ROUGE-L precision (0.14), and F1 (0.13). This can be connected to content accuracy since the more token overlap there is between the ground truth and the LLM-generated response, the more it is possible to trust it was based on precise and verifiable data. On the other hand, human evaluation has a moderate negative association with BLEU (-0.33). This may be due to the brevity penalty assigned by BLEU to short responses. The LLM temperature node does not influence other evaluations.

Centrality indices were also employed to visualize the relationships between automatic and human evaluation metrics. Figure 5 was generated using the `qgraph` package with the `centralityPlot` function (Epskamp et al., 2018). Nodes represent variables, and edges show statistical relationships in a network. Centrality indices (strength, betweenness, closeness) quantify node importance as standardized z-scores, with higher scores indicating greater centrality or influence relative to the network average. Betweenness represents the number of times a node lies on the shortest path between other nodes, indicating its control over communication in the network; Closeness is the inverse of the sum of distances from a node to all other nodes, measuring how close a node is to all other nodes; and Strength is the sum of the absolute weights of all edges connected to a node, representing how strongly connected a node is to the network.

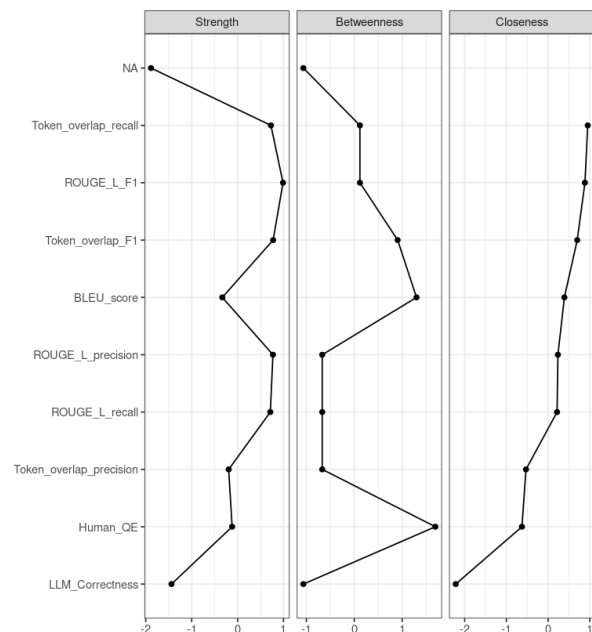
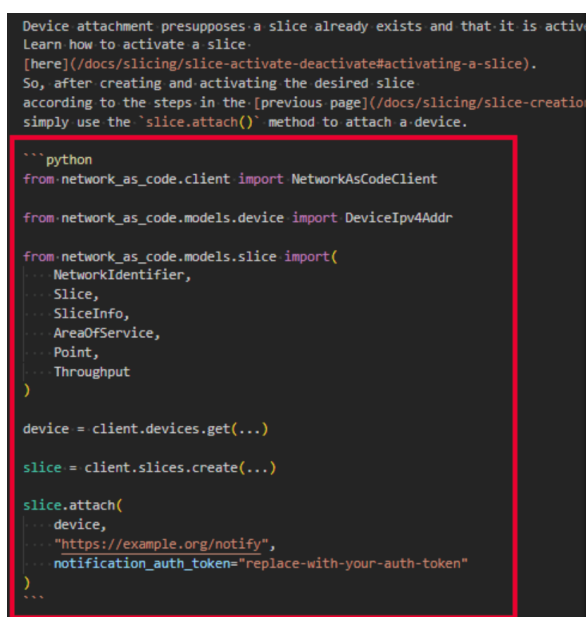


Figure 5: Closeness centrality index

This centrality analysis ordered by closeness reveals distinct patterns. The automatic metrics display more strength and influence due to their quantitative nature while interacting with each other, with token overlap and ROUGE being the strong ones while BLEU is more peripheral. Human QE demonstrates a neutral strength compared to the deterministic metrics. Additionally, the betweenness index shows that it serves as a bridge between LLM evaluation and deterministic metrics. This can be observed through its positive z-score. LLM evaluation remains relatively isolated, suggesting a unique perspective that aligns more closely with human judgment, possibly due to their qualitative nature, than with deterministic metrics. This highlights the complementary nature of qualitative and quantitative evaluations, while human or LLM judgments can provide more holistic assessments, the quantitative metrics offer a granular analysis of quantitative aspects of text quality, such as token overlap. This closeness plot and other centrality ones also ordered by betweenness and strength are included in the appendix A.

4.3 Qualitative analysis example

Figures 6 and 7 show an example of the ground truth reference and a response generated by GPT-4 for the prompt “Can you provide an example of attaching a device to a slice using an SDK?”. The human and LLM quality evaluations for this response are given below the figures.



```

Device attachment presupposes a slice already exists and that it is active.
Learn how to activate a slice.
[here]([docs/slicing/slice-activate-deactivate#activating-a-slice]).
So, after creating and activating the desired slice
according to the steps in the [previous page]([docs/slicing/slice-creation]),
simply use the 'slice.attach()' method to attach a device.

'''python
from network_as_code.client import NetworkAsCodeClient

from network_as_code.models.device import DeviceIpv4Addr

from network_as_code.models.slice import(
    NetworkIdentifier,
    Slice,
    SliceInfo,
    AreaOfService,
    Point,
    Throughput
)

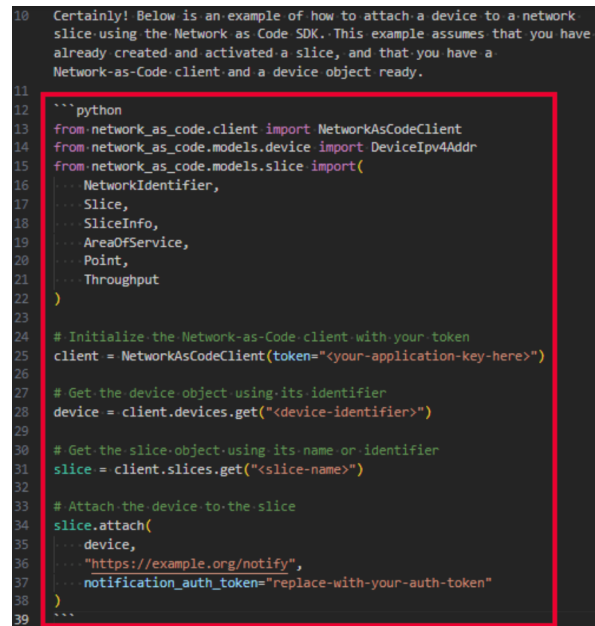
device = client.devices.get(...)

slice = client.slices.create(...)

slice.attach(
    device,
    "https://example.org/notify",
    notification_auth_token="replace-with-your-auth-token"
)
'''

```

Figure 6: Ground truth context reference



```

10 Certainly! Below is an example of how to attach a device to a network
11 slice using the Network as Code SDK. This example assumes that you have
12 already created and activated a slice, and that you have a
13 Network-as-Code client and a device object ready.
14
15 '''python
16 from network_as_code.client import NetworkAsCodeClient
17 from network_as_code.models.device import DeviceIpv4Addr
18 from network_as_code.models.slice import(
19     NetworkIdentifier,
20     Slice,
21     SliceInfo,
22     AreaOfService,
23     Point,
24     Throughput
25 )
26
27 # Initialize the Network-as-Code client with your token
28 client = NetworkAsCodeClient(token="<your-application-key-here>")
29
30 # Get the device object using its identifier
31 device = client.devices.get("<device-identifier>")
32
33 # Get the slice object using its name or identifier
34 slice = client.slices.get("<slice-name>")
35
36 # Attach the device to the slice
37 slice.attach(
38     device,
39     "https://example.org/notify",
40     notification_auth_token="replace-with-your-auth-token"
41 )
42 '''

```

Figure 7: GPT-4 response

- Human evaluation of GPT-4’s response:
 “GPT-4 does not use the same style of shortening the code by using ellipsis between parentheses. Instead, it provides example values that work similarly. This LLM did not miss nor omit any imports necessary for the code to work and focused on the main objective of the question. The additional note on the prerequisite of having an active slice is only relevant to the whole documentation page. Since this is an answer to a direct question, points were not deducted for that”. Score: 1.0.
- LLM evaluation of GPT-4’s response:
 “The answer is relevant to the question and correct. It provides a code example of how to attach a device to a network slice using an SDK, which aligns with the ground truth reference. However, it does not mention the prerequisite of having an active slice or provide the link to learn about activating a slice, which is included in the ground truth. Therefore, it is not entirely complete”. Score: 0.25.

This example illustrates how human and LLM quality analyses may differ. According to human evaluation, when prompted with “Can you provide an example of attaching a device to a slice using an SDK?”, GPT-4’s response in Figure 7 did not contain code or text errors and was more similar to the ground truth content in Figure 6, so no points

were deducted. However, according to the LLM evaluation, mentioning that it is necessary to have an active slice before attaching a device is considered important, and points were deducted from its evaluation. Perhaps, a human might consider that it is not necessary to repeat information since the active slice requirement for device attachment is mentioned upon slice creation and previous pages in the documentation.

5 Discussion

As seen in the (4.2), human evaluation is still somehow correlated with deterministic metrics, even though not strongly, while LLM evaluation is peripheral, only associating with the human one. Overall, the deterministic metrics had higher correlations amongst themselves, mostly likely due to the token overlap between ground truth and generated content. It is important to remember that “hallucinated” responses can also have high token overlap without necessarily generating more accurate responses relevant to a real use-case scenario.

On the other hand, human and LLM quality analyses seem more closely related to each other, even when using different frameworks to evaluate quality. This shows that an increase in the human evaluation variable relates to an increase in the LLM one. Additionally, the linear regression model plots seemed to reflect similarities in model preference by both human and LLM judgment. However, as the distribution of human quality evaluation scores had several peaks, it could indicate that the human evaluation had a more varied score distribution, due to the detailed error typology followed by the quality analysis type, while the LLM analysis mostly considered correctness and relevance as the main reference. Furthermore, as illustrated by the qualitative example (4.3), the assessments of a human evaluator and the LLM do not always correspond to each other.

6 Conclusion and future work

This paper analyzed technical documentation content generated by three different LLMs using RAG technology and compared the responses to product documentation. Different types of analyses were done: a qualitative analysis using DQF-MQM error typology by a technical writer, an automatic LLM correctness assessment, and a deterministic evaluation using BLEU, ROUGE, and token overlap. The evaluation scores were contrasted and

compared using Network Analysis, linear regression models, and a histogram with score distribution. Deterministic metrics have strong relationships with each other, while human analysis correlates moderately with LLM analysis and weakly with deterministic metrics.

The DQF-MQM translation quality evaluation framework was found to be a useful model also for the evaluation of technical writing content and shows potential for improving evaluation methods in the generative AI field. The current study is limited by involving only one technical writer as an evaluator, but the approach can provide a basis for further studies involving multiple evaluators. Future work could also include integrating such qualitative evaluation frameworks to evaluation packages like Continuous-eval for a more granular approach to evaluating LLM output.

Overall, the evaluation results indicate that LLMs are not yet ready as producers of novel content or standalone solutions for technical writing content. Future research is likely to continue exploring methods for enhancing chatbot responses for technical documentation purposes through RAG or different prompting techniques such as chain-of-thought (Wei et al., 2024). Developing novel and reliable quality evaluation methods is therefore also an essential challenge for positive advancements in this area.

7 Limitations

The dataset is relatively small with a total of 72 responses analyzed. However, the responses were relatively long with several of them comprising whole Markdown pages. The evaluations, though subjective with only one evaluator, are informed by their expertise as a professional technical writer familiar with the product. Additionally, using a proprietary RAG chatbot technology limited the possibility of providing a more detailed technical description here.

Acknowledgments

We thank Nokia, particularly the Network as Code Team and chatbot developers, for authorizing data usage for this research.

References

Sheila Castilho, Stephen Doherty, Federico Gaspari, and Joss Moorkens. 2018. Approaches to Human

- and Machine Translation Quality Assessment. In Joss Moorkens, Sheila Castilho, Federico Gaspari, and Stephen Doherty, editors, *Translation Quality Assessment: From Principles to Practice*, pages 9–38. Springer International Publishing, Cham.
- DAIR.AI. Basics of prompting [online]. n.d. Prompt Engineering Guide.
- Sacha Epskamp, Denny Borsboom, and Eiko I. Fried. 2018. Estimating psychological networks and their accuracy: A tutorial paper. *Behavior Research Methods*, 50:195–212.
- Mikhail Evtikhiev, Egor Bogomolov, Yaroslav Sokolov, and Timofey Bryksin. 2023. Out of the BLEU: How should we assess quality of the Code Generation models? *Journal of Systems and Software*, 203:111741.
- Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André Martins, Graham Neubig, Ankush Garg, Jonathan Clark, Markus Freitag, and Orhan Firat. 2023. The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1066–1083. Association for Computational Linguistics.
- Markus Freitag, Ricardo Rei, Nitika Mathur, Chiklu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022. Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. In *Proceedings of the Seventh Conference on Machine Translation*, pages 46–68. Association for Computational Linguistics.
- Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv*.
- Tom Kocmi and Christian Federmann. 2023a. GEMBA-MQM: Detecting translation quality error spans with GPT-4. In *Proceedings of the Eighth Conference on Machine Translation*, pages 768–775. Association for Computational Linguistics.
- Tom Kocmi and Christian Federmann. 2023b. Large language models are state-of-the-art evaluators of translation quality. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 193–203. European Association for Machine Translation.
- LangChain. Langchain documentation [online]. 2023. GitHub Repository.
- Daniel Lüdecke. 2024. *sjPlot: Data Visualization for Statistics in Social Science*. R package version 2.8.16.
- Mistral AI. Announcing Mistral 7B: The best 7B model to date, Apache 2.0 [online]. 2023.
- Alexandre Nikolaev and Neil Bermel. 2022. Explaining uncertainty and defectivity of inflectional paradigms. *Cognitive Linguistics*, 33(3):585–621.
- Nokia. Network as Code portal documentation [online]. 2024. Nokia Network as Code Portal.
- OpenAI. Models [online]. 2023. OpenAI Documentation.
- 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. Association for Computational Linguistics.
- R Core Team. 2024. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.
- Relari. Open-source evaluation framework and synthetic data generation pipeline [online]. 2023. Continuous Eval.
- Karen de Souza. 2024. How much can AI assist in the generation of technical documentation research on AI as a support for technical writers. Master’s thesis, University of Eastern Finland.
- James Swarts. 2018. *Wicked, Incomplete, and Uncertain: User Support in the Wild and the Role of Technical Communication*. Utah State University Press, Logan.
- TAUS. Error annotation based on DQF-MQM [online]. n.d. TAUS, The Language Data Network.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2024. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, page 14. Curran Associates Inc.
- Hadley Wickham. 2016a. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.
- Hadley Wickham. 2016b. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.
- Hadley Wickham, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D’Agostino McGowan, Romain François, Garrett Grolemund, Alex Hayes, Lionel Henry, Jim Hester, Max Kuhn, Thomas Lin Pedersen, Evan Miller, Stephan Milton Bache, Kirill Müller, Jeroen Ooms, David Robinson, Dana Paige Seidel, Vitalie Spinu, Kohske Takahashi, Davis Vaughan, Claus Wilke, Kara Woo, and Hiroaki Yutani. 2019. Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686.
- Yi Zhang and Pasquale Antonante. A practical guide to RAG pipeline evaluation (Part 1: Retrieval) [online]. 2023.

A Appendices

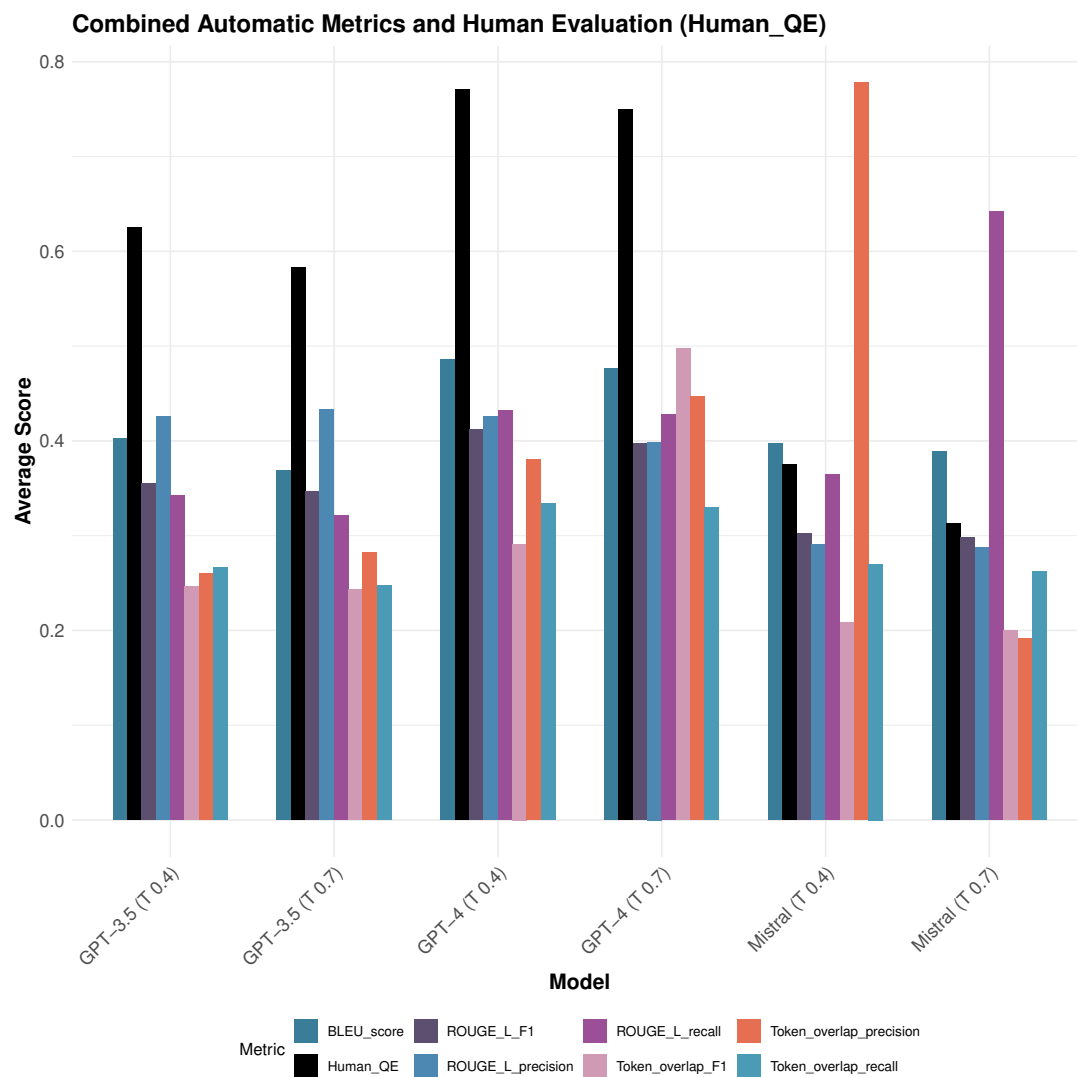


Figure 8: Human QE and deterministic averages

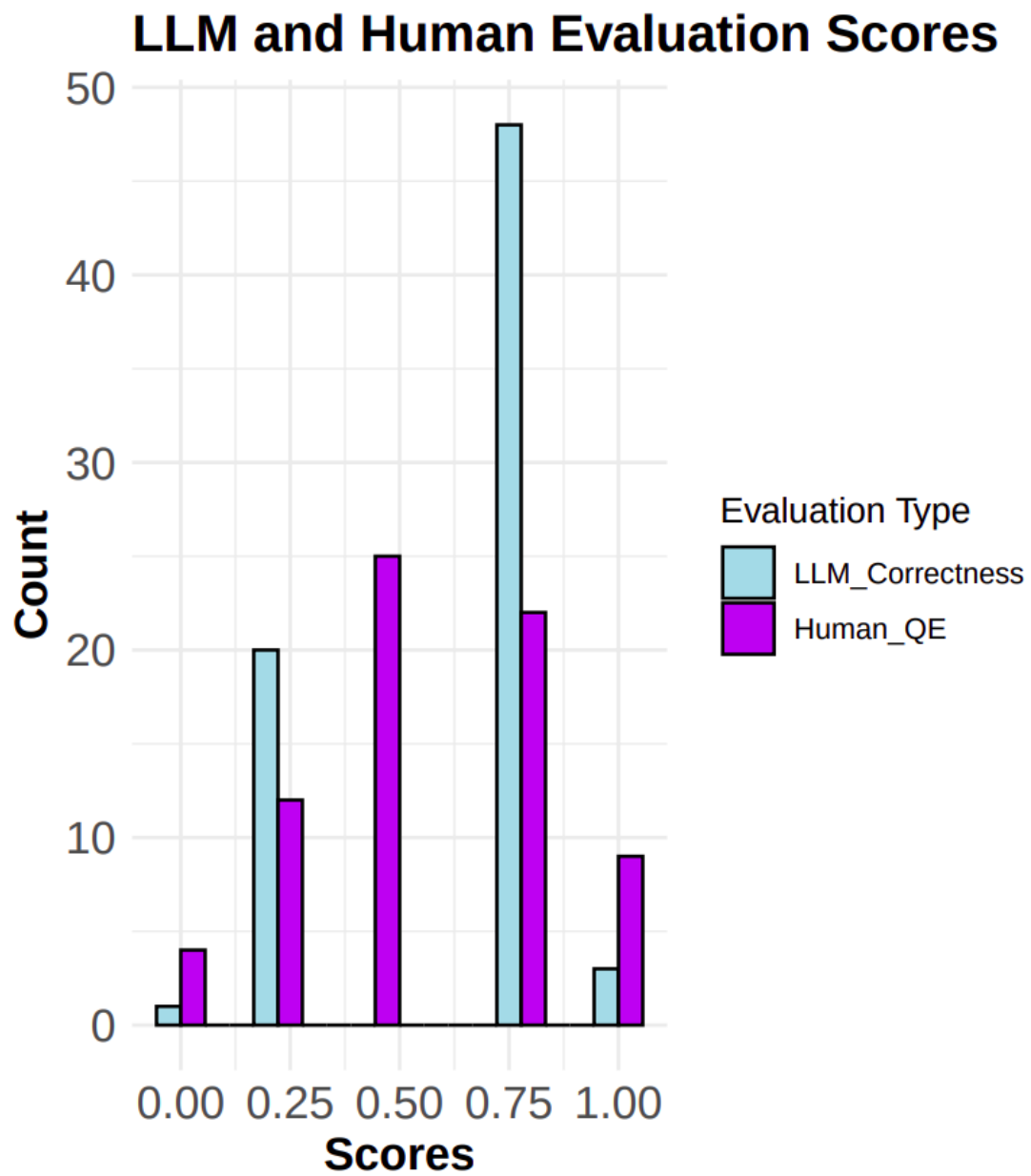


Figure 9: LLM and human scores distribution

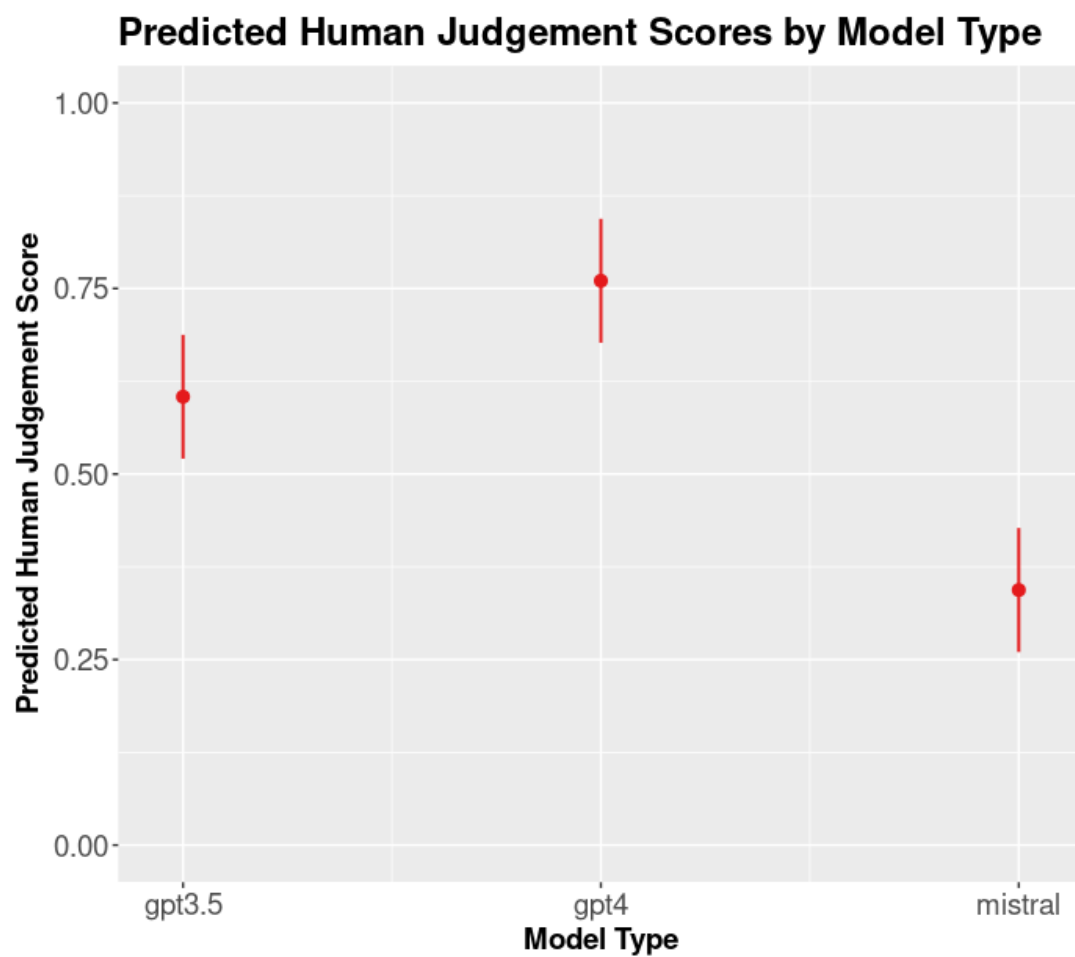


Figure 10: Human analysis linear regression model

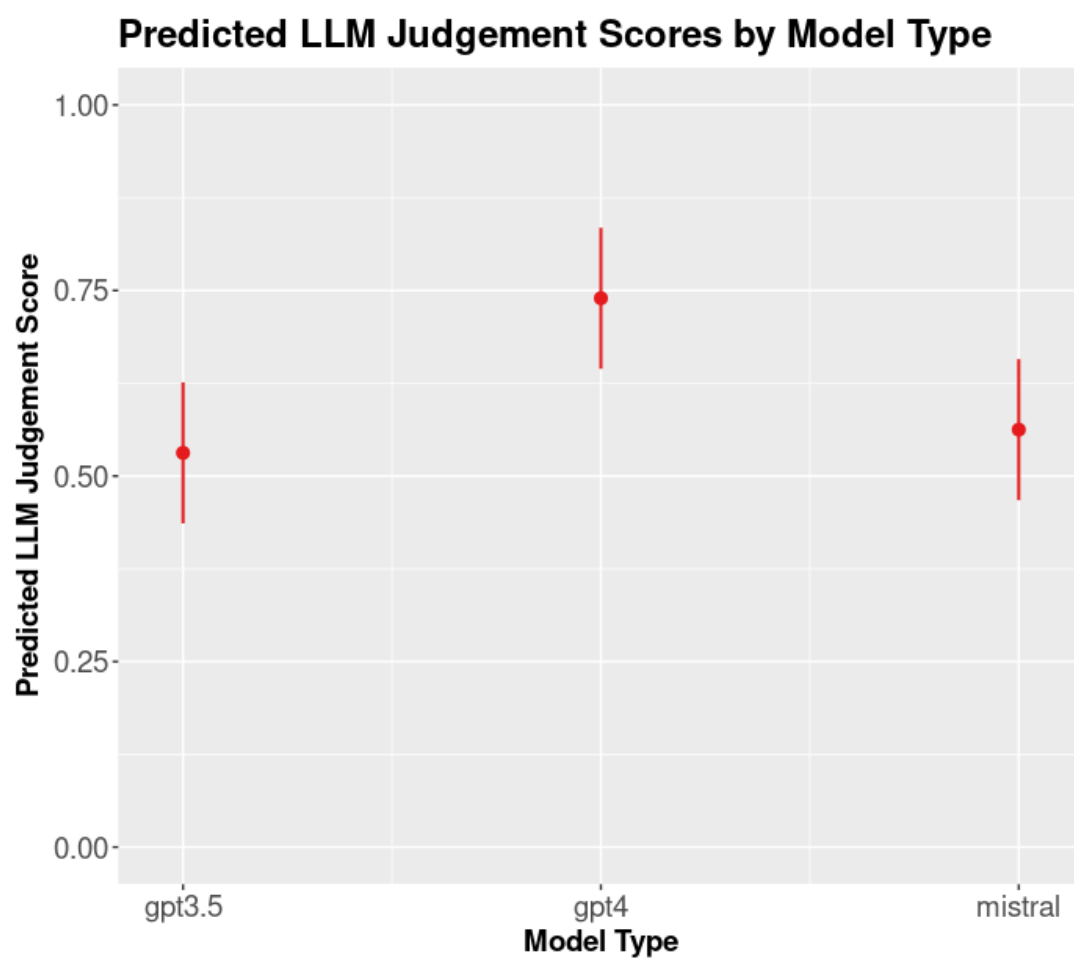


Figure 11: LLM analysis linear regression model

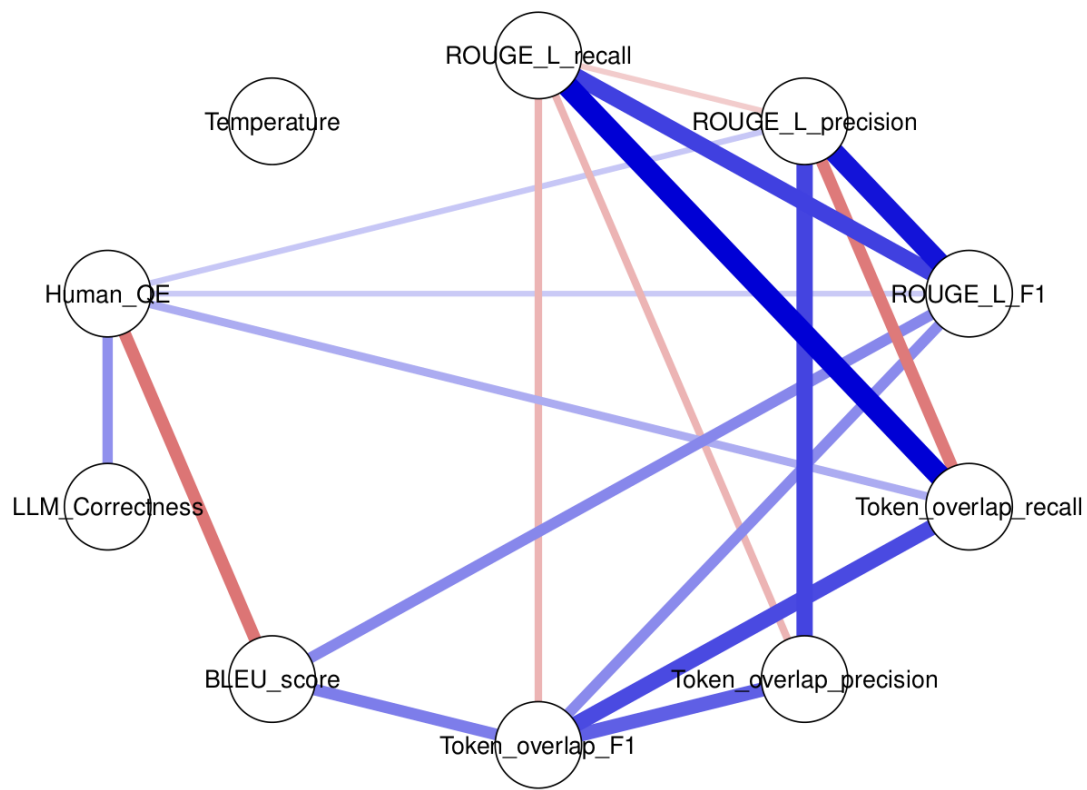


Figure 12: Correlation Network Analysis

Device attachment presupposes a slice already exists and that it is active. Learn how to activate a slice. [\[here\]\(/docs/slicing/slice-activate-deactivate#activating-a-slice\)](#). So, after creating and activating the desired slice according to the steps in the [\[previous page\]\(/docs/slicing/slice-creation\)](#), simply use the `slice.attach()` method to attach a device.

```
```python
from network_as_code.client import NetworkAsCodeClient

from network_as_code.models.device import DeviceIpv4Addr

from network_as_code.models.slice import (
 NetworkIdentifier,
 Slice,
 SliceInfo,
 AreaOfService,
 Point,
 Throughput
)

device = client.devices.get(...)

slice = client.slices.create(...)

slice.attach(
 device,
 "https://example.org/notify",
 notification_auth_token="replace-with-your-auth-token"
)
```
```

Figure 13: Ground truth context reference

```

10 Certainly! Below is an example of how to attach a device to a network
    slice using the Network as Code SDK. This example assumes that you have
    already created and activated a slice, and that you have a
    Network-as-Code client and a device object ready.
11
12 ```python
13 from network_as_code.client import NetworkAsCodeClient
14 from network_as_code.models.device import DeviceIpv4Addr
15 from network_as_code.models.slice import(
16     ... NetworkIdentifier,
17     ... Slice,
18     ... SliceInfo,
19     ... AreaOfService,
20     ... Point,
21     ... Throughput
22 )
23
24 # Initialize the Network-as-Code client with your token
25 client = NetworkAsCodeClient(token="<your-application-key-here>")
26
27 # Get the device object using its identifier
28 device = client.devices.get("<device-identifier>")
29
30 # Get the slice object using its name or identifier
31 slice = client.slices.get("<slice-name>")
32
33 # Attach the device to the slice
34 slice.attach(
35     ... device,
36     ... "https://example.org/notify",
37     ... notification_auth_token="replace-with-your-auth-token"
38 )
39 ```

```

Figure 14: GPT-4 response

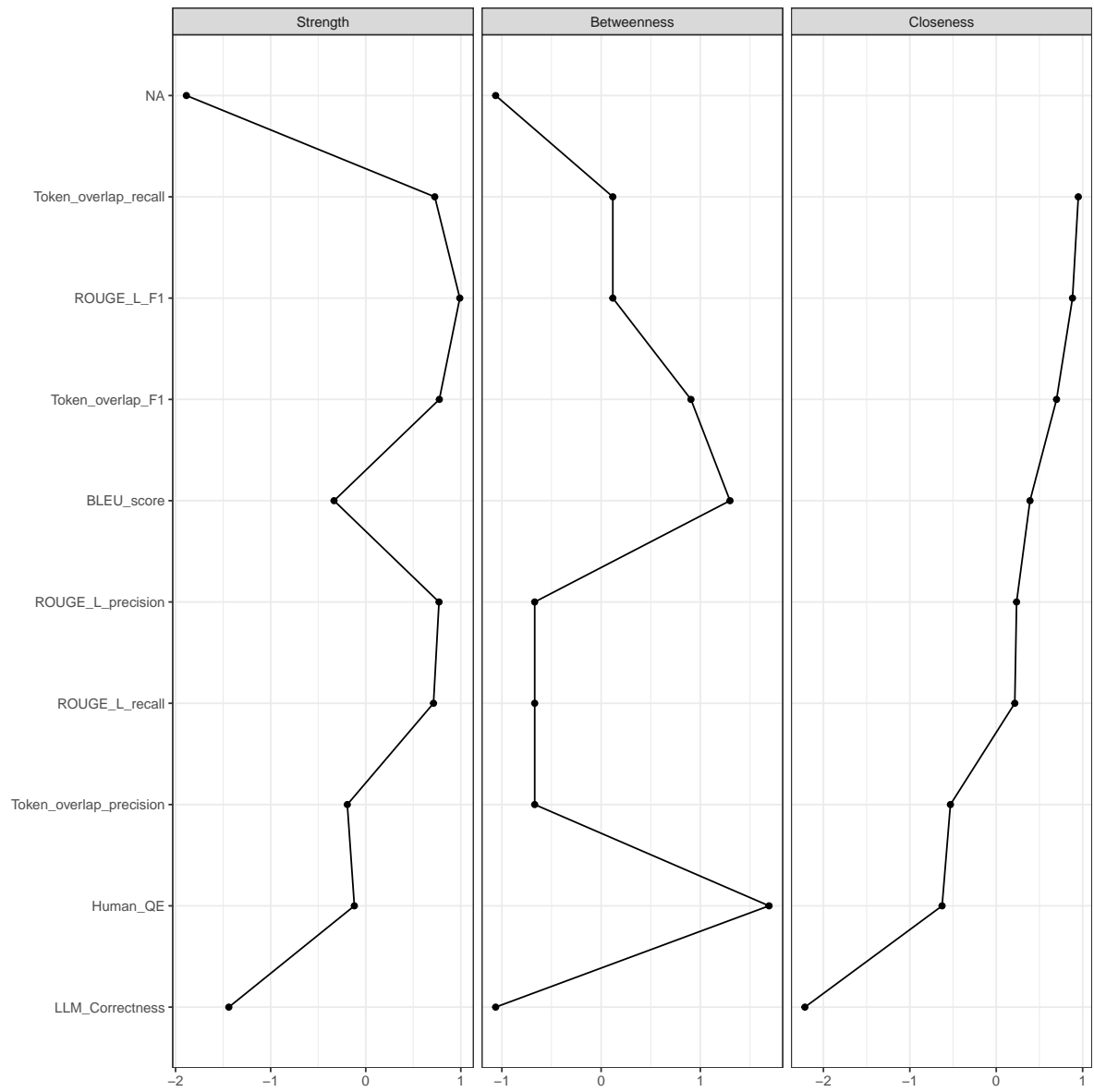


Figure 15: Closeness centrality index

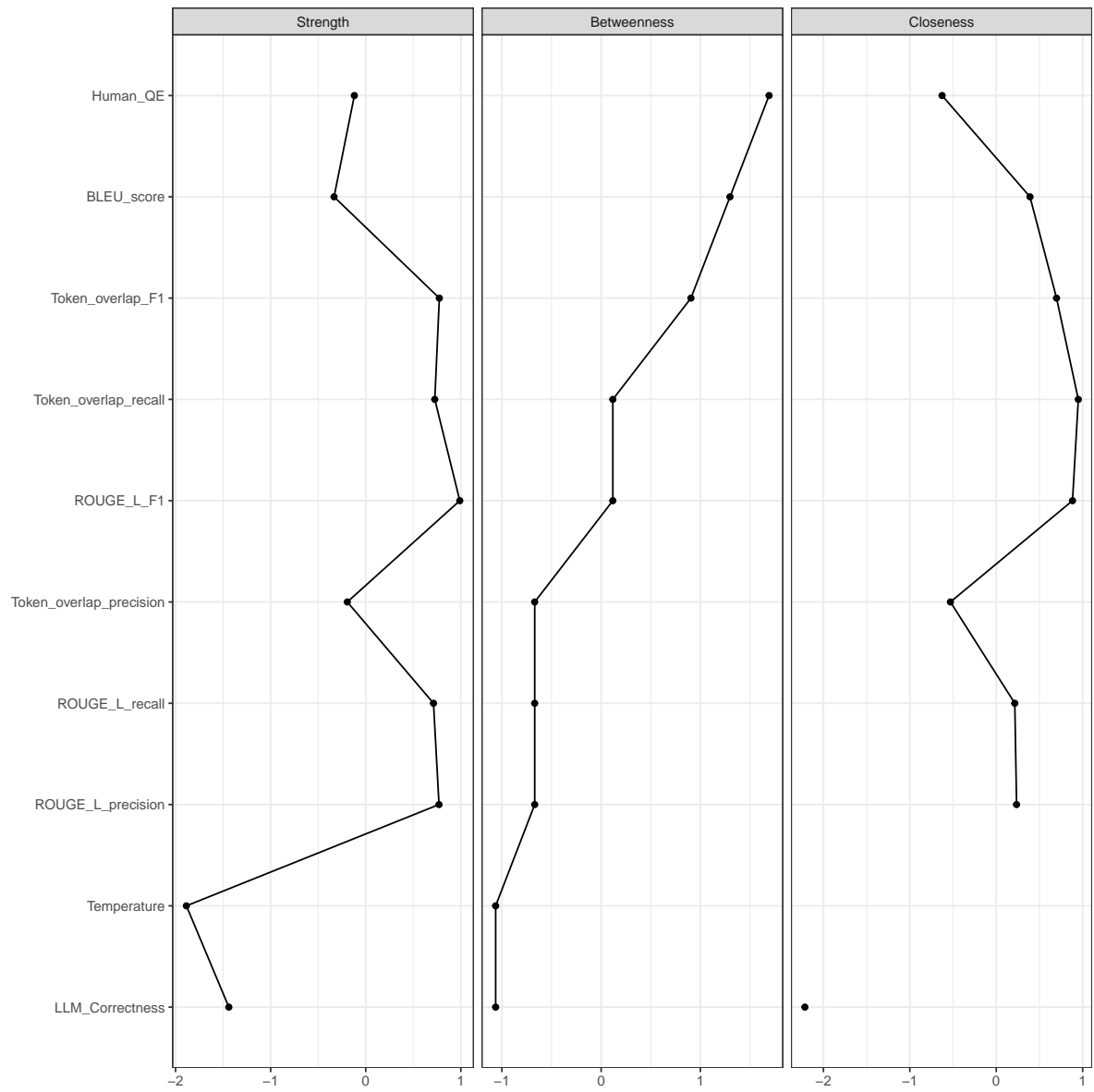


Figure 16: Betweenness centrality index

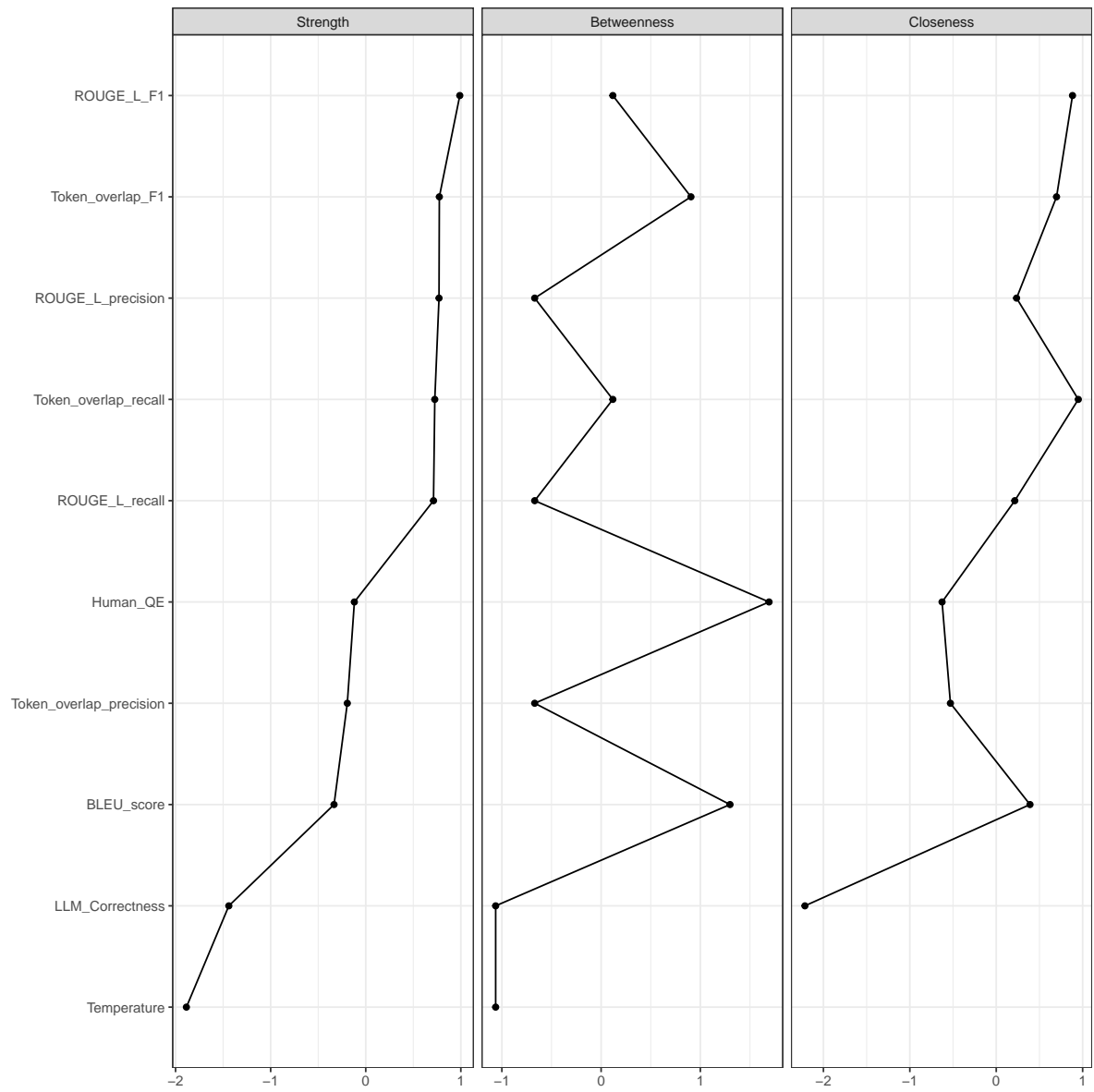


Figure 17: Strength centrality index

MC-19: A Corpus of 19th Century Icelandic Texts

Steinþór Steingrímsson Einar Freyr Sigurðsson Atli Jasonarson

Árni Magnússon Institute for Icelandic Studies

{steinthor.steingrimsson, einar.freyr.sigurdsson,
atli.jasonarson}@arnastofnun.is

Abstract

We present MC-19, a new Icelandic historical corpus containing texts from the period 1800–1920. We describe approaches for enhancing a corpus of historical texts, by preparing the texts so that they can be processed using state-of-the-art NLP tools. We train encoder-decoder models to reduce the number of OCR errors while leaving other orthographical variation be. We generate a separate modern spelling layer by normalizing the spelling to comply with modern spelling rules, using a statistical modernization ruleset as well as a dictionary of the most common words. This allows for the texts to be PoS-tagged and lemmatized using available tools, facilitating usage of the corpus for researchers and language technologists. The published version of the corpus contains over 270 million tokens.

1 Introduction

For most areas of language technology, large text corpora and other textual resources have become increasingly important in recent years, not least due to large language models (LLMs) becoming ever more pervasive. Textual resources are not only necessary to train such models to use and decipher language, but also for question answering, information extraction and other generative tasks. With better access to data and tools to work with linguistic data, data-oriented approaches to linguistic research and lexicography have become more common and more useful, allowing more researchers to use such approaches in their work. Most commonly, large text corpora comprise recent texts. Texts from the digital era, written to be published online, can be a good tool to study recent changes and variation in language, as well as

recent events and how they are perceived as they are happening. When we want to study older language, the new methods fall short if the data is lacking. In order to facilitate linguistic research for older texts, we have compiled a new corpus, the 19th Century Megacorpora (MC-19). Such research might include diachronic linguistic studies and syntactic analysis.

The aim of the MC-19 project is to compile as large a corpus as possible, comprising texts written from 1800 to 1920. The first edition of the corpus contains texts from journals and newspapers published in this period and scanned by the National and University Library of Iceland (LBS), but we intend to extend the corpus in a later edition to also include published books. We use the OCRred texts published by LBS and develop post-processing models to find and fix OCR errors in the texts, while aiming to not change anything else. Finally, we normalize the texts using modern spelling.

The contributions of the project, presented in this paper, include:

- The corpus itself, published in TEI-format¹ and in a keyword-in-context (KWIC) search engine.² The published corpus contains post-processed OCRred texts and a version transcribed to modern spelling, PoS-tagged and lemmatized.
- A list of common OCR-errors when processing Icelandic texts. We manually checked a wide range of random texts on *Tímarit.is* from this period and analyzed the OCR errors. The error list, available on GitHub,³ was used for generating synthetic training data for post-processing (see Section 4.2).

¹<http://hdl.handle.net/20.500.12537/360>

²<https://malheildir.arnastofnun.is/>

³<https://github.com/stofnun-arna-magnussonar/MC19/OCRErrors>



Figure 1: Token count by year (1801–1920).

- Approaches to post-processing OCR texts and transcribing to modern spelling. Models and scripts are available on GitHub.⁴

2 Why Do We Need a 19th Century Corpus?

Syntacticians studying Icelandic syntax, linguists studying word formation, inflectional morphology or semantics and lexicographers compiling dictionaries have been the most active users of the Icelandic Gigaword Corpus (IGC, Steingrímsson et al. 2018; Barkarson et al. 2022). Amongst these users there has been a call for corpora covering larger periods and going as far back in time as possible, in order to further the study of, for example, semantic or syntactic change. Language technologists, working on LLMs, are interested in studying how different LLMs comprehend older language in comparison with current language and to add older texts into the training process to see if it enhances the models’ abilities to generate informative texts covering previous time periods. With MC-19, we aim to facilitate work in all these different fields of research.

As a demonstration of research that could be furthered with a corpus like ours, we could look at

an empirical study on the reflexive passive in Icelandic conducted by Árnadóttir et al. (2011). This construction can be dated back to the 19th century as Árnadóttir et al. show. To find as old examples as they could at *Tímarit.is*, the authors had to look for word strings. To find different examples of *flýta sér* ‘hurry (oneself)’ in the reflexive passive, they had to search for, e.g., “var flýtt sér” (‘was hurried oneself’), “var flýtt sjer”, “er flýtt sér”, “er flýtt sjer”, “verið flýtt sér”, “verið flýtt sjer”, etc.; they also searched for, e.g., adverbs like *oft* ‘often’ intervening between the auxiliary *vera* ‘be’ and the participle (cf. Árnadóttir et al. 2011, 64).

This is rather time consuming, especially when one wants to look for as many different verbs as possible. This is, however, made easier in MC-19 as the corpus is PoS-tagged and lemmatized and we can therefore look for both certain word forms and tags. A search query that looks for the lemma *vera* ‘be’ followed by past participle (and between *vera* and the participle can be at most one word) which in turn is immediately followed by the reflexive pronoun forms *sig/sér/sín* seems to return most of the 19th-century examples from Árnadóttir et al.’s study (but of course not the ones that differ in structure from the setup in the query). This search query also returns at least two exam-

⁴<https://github.com/stofnun-arna-magnussonar/MC19>

ples from the 19th century that are not reported on in Árnadóttir et al. (2011).

3 Related Work

A wide range of historical corpora has been compiled and made available for different languages. Many of these are small, less than a million words, but there are notable exceptions. The Corpus of Late Modern English Texts (De Smet, 2005) contains over 34 million words in texts from the period 1710–1920, and the Royal Society Corpus (Kermes et al., 2016) includes all publications of the Philosophical Transactions of the Royal Society of London from 1665 to 1869, approximately 32 million tokens. ChroniclItaly (Viola, 2021) is a corpus of Italian language newspapers published in the United States between 1898 and 1920, 16.6 million words in total, and the Diorisis Ancient Greek Corpus contains 10.2 million words in texts spanning from Homer to the fifth century AD (Vatri and McGillivray, 2018). Turning to Icelandic, the Icelandic Parsed Historical Corpus (IcePaHC, Rögnvaldsson et al. 2012; Wallenberg et al. 2024) contains approximately 1 million words written between the 12th and 21st centuries. The Saga Corpus (Rögnvaldsson and Helgadóttir, 2011) contains the texts of the Icelandic sagas as well as a few other historical texts in modernized editions, and the IGC, which is 2.6 billion words in total and mostly has texts from the 21st century and the end of the 20th century, contains a few thousand words in texts written before the year 1900, all from the IGC-Law (Barkarson and Steingrímsson, 2022) subcorpus, containing law texts.

A number of studies have been carried out on how best to correct historical OCR data. Bjerring-Hansen et al. (2022) present a pipeline for correcting 19th century Danish fraktur. Their approach is rather different from ours, starting by changing “obvious and unambiguous OCR errors”, then aligning multiple OCR output candidates and perform selective correction with reference to these and finally employing a spell checker.

Different approaches have been taken when doing historical spelling normalization. Schneider et al. (2017) use machine translation (MT) systems, translating original spelling into normalized texts. While they compare rule-based and SMT-based MT systems, Tang et al. (2018) evaluate the effectiveness of using neural-based MT for the task. Bollmann (2019) highlights that there is no

consensus on the state-of-the-art approach to historical text normalization and compares a number of approaches. He finds that lookups based on naive memorization are most often effective for seen tokens, while MT-based methods perform best in unseen cases.

4 Data Processing

Our data is collected from *Tímarit.is*, a digital library platform for newspapers and periodicals that goes back to the early 19th century. The platform allows users to search texts, with OCR-generated text files for each page in the library. Rather than running our own OCR-models on the pages, which would have been resource intensive and not necessarily very beneficial, we decided to use the texts OCRred by the providers of *Tímarit.is*, LBS. In order to facilitate our work, LBS provided us with all text files for our project, covering the period in question, 1800–1920.

So that we could exclude too noisy texts, we manually checked the OCR quality of newspapers and periodicals that were candidates for our corpus. The process is described in Section 4.1.

During the selection process we compiled a list of common OCR errors. We then enlarged it by extracting a list of OCR errors from manually corrected texts from this period that we had access to. The information was used to automatically introduce OCR-like errors to correct texts, thus creating a parallel data set for training models to post-process OCRred data. We also took random samples from the texts that we decided to use and manually fixed the OCR errors to create an evaluation set. We describe this in more detail in Section 4.2.

All the selected texts were run through the post-processing models we trained, before normalizing them to modern spelling, using the approaches described in Section 4.3. Having the modern spelling variants we could PoS-tag and lemmatize the texts using the best available tools for Icelandic, which are trained on modern texts.

4.1 Data Selection

When selecting the publications to include, we checked all newspapers and periodicals available on *Tímarit.is* from the period 1800–1920, in total approximately 400 titles. Individual titles were evaluated by randomly selecting three volumes (years) and from each of the volumes three pages were inspected. In total, nine pages were thus

| Error | Correct | Error Word | Correct Word |
|-------|---------|------------|--------------|
| 3 | ð | hú3 | húð |
| l> | þ | l>jer | þjer |
| ce | æ | lceknis | læknis |
| cl | d | breidcl | breidd |
| h | li | háskóh | háskóli |
| rn | m | heirnila | heimila |

Table 1: Examples of character level OCR-errors.

checked for each title. We used three categories in our evaluation:

- Green – OCR seems to be accurate and does not contain a lot of errors;
- Yellow – Most of the text looks good, but errors are common in some parts. These texts need more rigorous fixing.
- Red – Probably unusable, mostly due to OCR not giving good results. All periodicals printed in fraktur are in this category as well as texts that the OCR model fails to reproduce, commonly due to bad print or unusual layout.

In the final corpus we decided to include everything from the first two categories, green and yellow, but leave out all material in the red category, leaving us with 317 sources deemed usable.

As we performed the checks, common OCR-errors were recorded. This way, a list of 330 errors were collected, which could later be used to help with fixing the errors. Examples of this can be seen in Table 1.

4.2 OCR Post-processing

We carried out post-processing on all texts delivered to us by LBS, using the approaches described in Jasonarson et al. (2023). This involved using an encoder-decoder Transformer model (Vaswani et al., 2017) trained from scratch using parallel data containing OCRred texts and manual corrections of these, as well as texts populated with artificial errors in conjunction with the unspoiled data.

We had access to manually corrected texts from 19th century periodicals and journals, which we matched to the uncorrected texts.⁵ This dataset

⁵These texts are a product of the project *Language Change and Linguistic Variation in 19th-Century Icelandic and the Emergence of a National Standard*, led by Ásta

| Original | Corrected | Frequency |
|-----------|-----------|-----------|
| <i>p</i> | <i>þ</i> | 2,779 |
| <i>i</i> | <i>í</i> | 1,141 |
| <i>li</i> | <i>h</i> | 247 |
| <i>rn</i> | <i>m</i> | 166 |
| <i>m</i> | <i>rn</i> | 77 |

Table 2: Examples of automatically extracted errors and statistics on them.

contains in total over 2 million tokens. We also used this data to gather more examples of OCR errors and to create statistics on which errors are the most common, examples of which are shown in Table 2. In turn, this information was used to generate a new dataset containing artificial errors.

The data into which the artificial errors were inserted were texts published between 1830 and 1920, taken from the Icelandic Text Archive.⁶ By doing this we have parallel data, with correct texts on the one hand and the same texts with errors like the ones commonly found in OCR output on the other. This data can then be used to train a system that effectively translates erroneous texts to correct texts, fixing many errors like the ones found in Table 1. In total, the artificial corpus contained almost 3 million tokens. We combined our two parallel datasets and split it into training and validation data, with the validation data being 15% of the total, approximately 750 thousand tokens, and the training set approximately 4.2 million tokens.

To evaluate the post-processing accuracy, we created an evaluation set by selecting random pages from the corpus and manually correct them. The evaluation set contains in total 18k tokens.

We trained three models, as described in Jasonarson et al. (2023), the best being a fine-tuned version of ByT5-base (Xue et al., 2022) which achieved a word error rate reduction of 55.07% – cutting the number of erroneous words in half.

4.3 Modernizing the Spelling

We manually modernized the 10,000 most common words in our training data and created a lookup dictionary. We also built a statistical spelling modernization ruleset by iterating over a small, manually modernized, parallel corpus, one token at a time, extracting the necessary ed-

Svavarsdóttir at the Árni Magnússon Institute for Icelandic Studies (e.g. Svavarsdóttir et al. 2014).

⁶<https://clarin.is/en/resources/textarchive/>

its needed to convert an old token into a modern one. This resulted in 101 rules, such as $je \rightarrow \acute{e}$ and $p \rightarrow f$, both of which are a frequent change from old tokens to modern ones.

To modernize our corpus, our system iterates over every sentence in a given original text and generates a modern counterpart. It looks at every token in the original sentence and checks whether it exists in the Database of Icelandic Morphology (DIM, Bjarnadóttir et al. 2019). If it does, the token gets added unchanged to the new modern sentence. If the token is not found in DIM, the system checks whether the word exists in the manually corrected lookup dictionary, and if so, the modernized spelling variant gets added to the new sentence. If a token is shorter than 3 characters, we do not try to modernize it and simply add it to the new sentence.

If an original token’s modern counterpart has not been found at this point, we create an empty list, which we populate with plausible candidates that we produce with several methods.

1. Using Kvistur (Daðason et al., 2020), we check whether the token is a compound word. If all of its parts exist in DIM, we add it to the candidate list.
2. We check whether there is a word in DIM that has a Levenshtein-distance (Levenshtein, 1966) of 1 (or 2, if the token is 12 characters or longer) from the original token. If it does and its edit from the original token is found in our statistical spelling modernization ruleset, we add it to the list, e.g. if the original token is *eptirlegukind* and *eftirlegukind* is found in DIM, as $p \rightarrow f$ is a known spelling modernization rule.
3. We apply all of the possible modernization rules to the token and if any of them produces a token which exists in DIM, we add it to the list.
4. We edit the token with two rules. If it ends with ‘r’, we try adding ‘u’ in front of it, e.g. *hestr* \rightarrow *hestur*, and check whether the resulting token is found in DIM. (Older forms of nouns often do not have ‘u’ in the ending before ‘r’.) We also check if doubling a consonant in the token, e.g. *byggð* \rightarrow *byggðð*, results in a known modern token. If either of these

returns a known modern token, we add it to the list of plausible candidates.

5. We use two models, a modern GEC model⁷ and IceBERT.⁸ We use the former as a spellchecker to edit the current token, and the latter, by masking the current token, to guess which token should be in its place. If either of these returns a token, which, when compared to the original token, can be inferred from the rules in our statistical spelling modernization dataset, we add it to the candidate list.

When all of these checks are completed, we simply add the most suggested token to the new sentence. If all of these methods fail, however, in producing a plausible candidate, the original token stays in the modern sentence. In such a case the token could be an uncommon one, but free of errors, or it could be the case that the applied methods fail to suggest the correct form.

4.4 Tagging and Lemmatization

The most accurate PoS-tagger and lemmatizer for Icelandic are trained to work with modern spelling varieties. We thus only tag and lemmatize the normalized version of the texts. We start by tokenizing the texts using Tokenizer,⁹ a Python program developed for tokenizing Icelandic texts. We use ABLTagger 3.0.0 (Steingrímsson et al., 2019; Jónsson et al., 2021) for PoS-tagging the texts. The tagger is reported to have an accuracy of 96.7% when using cross-validation on MIM-GOLD (Helgadóttir et al., 2014; Barkarson et al., 2021), the standard dataset for training and evaluation of PoS-tagging for Icelandic. Nefnir (Ingólfssdóttir et al., 2019) is the most suitable lemmatizer for Icelandic texts, reported to produce only a fraction of the errors other lemmatizers for Icelandic produce. It uses the tags output by the PoS-tagger to help with finding correct lemmas, using suffix substitution rules derived from DIM.

4.5 Data Statistics

MC-19 contains a total of 272,516,487 tokens from 317 sources. As shown in Figure 1, most of the tokens are from material published late in the

⁷ByT5-model: <https://huggingface.co/mideind/yfirlestur-icelandic-correction-byt5>

⁸<https://huggingface.co/mideind/IceBERT>

⁹<https://pypi.org/project/tokenizer/>

| Title | Period | Token Count |
|--------------------------------|----------------------|-------------|
| Lögberg | 1888–1920 | 41,002,958 |
| Heimskringla | 1886–1920 | 32,486,522 |
| Þjóðólfr | 1848–1920 | 15,364,734 |
| Morgunblaðið | 1913–1920 | 13,175,574 |
| Lögrétta | 1906–1920 | 8,485,516 |
| Austri | 1883–1888; 1891–1917 | 7,183,953 |
| Fjallkonan | 1884–1911 | 6,993,309 |
| Þjóðviljinn + Þjóðviljinn ungi | 1886–1915 | 6,971,801 |
| Skírnir | 1827–1920 | 6,460,688 |
| Norðurland | 1901–1920 | 5,105,768 |

Table 3: The ten publications in MC-19 that contain the largest number of tokens. The table shows the period as represented in the corpus. Some of these publications continued to be published after 1920.

period, with more than 50% being from the last 14 years (1907–1920). The first 50 years only contain approximately 3.5 million tokens (there is no data in the corpus for the years from 1803 to 1817).

Furthermore, a few publications tower over the rest, with ten publications containing more than 5 million tokens each, as shown in Table 3. These ten publications represent more than half the corpus data.

5 Use and Availability

The corpus is published under an open CC BY 4.0 license. It is available online in two different forms for different uses and users. It is made available for search online in a KWIC-portal, powered by Korp (Borin et al., 2012). Users can search for word forms in both the original version (OCRred text) and in the modern spelling transcription, with the modern spelling transcription being PoS-tagged and lemmatized, allowing for more complicated search in that data. The results are shown in parallel, so while the user can search using modern spelling varieties, the original ones are also shown. This format is expected to mostly be useful to linguists, lexicographers and students of Icelandic.

The TEI-version is available for download. It contains whole sentences in the original version as well as the normalized version using modern spelling. The normalized version is furthermore PoS-tagged and lemmatized. We expect this format to be most useful for language technologists for analyzing and building tools and language models. Linguists competent in programming may also find that working with these annotated documents allows for more complicated

analysis and research than when limited to KWIC-analysis.

6 Conclusion and Future Work

We have presented a new text corpus, MC-19, containing Icelandic texts from the 19th century and the first decades of the 20th century. The first version of this corpus has been published and is made available in a TEI-format as well as in an online KWIC-platform, powered by Korp.

While care has been taken to make the texts as readable and close to the printed material as possible, using a post-processing step and a spelling-modernization step, there is always room for improvement. The post-processing process reduces the number of OCR errors by 55.07%. Improving the performance in this step would make the corpus more accurate and useful. This could possibly be achieved by improving the post-processing models, for example by generating more artificial training data or more diverse training data. Some error reduction may be achieved simply by replacing possible errors with possible corrections, using our error list. For such an approach, which tends to be greedy, some measures would need to be taken to limit the possibility of generating new errors. This could possibly be achieved by mapping only from unknown words (containing possible errors) to known words, calculating the likelihood of the change using n-grams or perplexity calculations or other approaches that may prove useful.

While most of the sentences in the corpus are as printed in the original publications, some are garbled due to problems with OCR that our methods could not solve. Training a classifier to select bad sentences for removal could make the corpus an

even better tool.

The spelling-modernization step helps the user find common words for which the spelling has changed, allowing for easier search and usage of the corpus, but the user will still find that some words are not modernized. A more thorough examination of this and improvements in the process will help with using the corpus for research. We intend to revisit these steps for a future version of the corpus, integrating additional normalization techniques and manually evaluate the merits of different approaches to this problem. We also intend to add texts from books published in the period, and are working on OCR-reading fraktur texts. While these texts may not add very much to this corpus in terms of word count, as the bulk of published texts in the period is in newspapers and periodicals, it may show a greater variety, both in terms of language and content. Available texts from previous periods, printed and hand-written, are also being considered for a sister corpus to this one.

Acknowledgments

We would like to thank three anonymous reviewers for helpful comments. We would like to thank the following people for their collaboration and contributions to the project: Finnur Ágúst Ingimundarson and Árni Davíð Magnússon for analyzing the OCR errors and Starkaður Barkarson for his work on publishing the corpus in TEI-format and setting it up on Korp. This project was supported by Rannís Infrastructure Fund, grant number 200336-6101.

References

- Hlíf Árnadóttir, Thórhallur Eythórsson, and Einar Freyr Sigurðsson. 2011. The passive of reflexive verbs in Icelandic. *Nordlyd*, 37:39–97.
- Starkaður Barkarson, Þórdís Dröfn Andrésdóttir, Hildur Hafsteinsdóttir, Árni Davíð Magnússon, Kristján Rúnarsson, Steinþór Steingrímsson, Haukur Páll Jónsson, Hrafn Loftsson, Einar Freyr Sigurðsson, Eiríkur Rögnvaldsson, and Sigrún Helgadóttir. 2021. MIM-GOLD 21.05. CLARIN-IS.
- Starkaður Barkarson and Steinþór Steingrímsson. 2022. IGC-Law 22.10 (annotated version). CLARIN-IS.
- Starkaður Barkarson, Steinþór Steingrímsson, and Hildur Hafsteinsdóttir. 2022. Evolving large text corpora: Four versions of the Icelandic Gigaword Corpus. In *Proceedings of the Language Resources and Evaluation Conference*, pages 2371–2381, Marseille, France. European Language Resources Association.
- Kristín Bjarnadóttir, Kristín Ingibjörg Hlynisdóttir, and Steinþór Steingrímsson. 2019. DIM: The Database of Icelandic Morphology. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 146–154, Turku, Finland. Linköping University Electronic Press.
- Jens Bjerring-Hansen, Ross Deans Kristensen-McLachlan, Philip Diderichsen, and Dorte Haltrup Hansen. 2022. Mending fractured texts. a heuristic procedure for correcting OCR data. *Digital Humanities in the Nordic and Baltic Countries Publications*, 4(1):177–186.
- Marcel Bollmann. 2019. A large-scale comparison of historical text normalization systems. 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 3885–3898, Minneapolis, Minnesota. Association for Computational Linguistics.
- Lars Borin, Markus Forsberg, and Johan Roxendal. 2012. Korp – the corpus infrastructure of Språkbanken. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012, Istanbul, Turkey*. European Language Resources Association.
- Jón Daðason, David Mollberg, Hrafn Loftsson, and Kristín Bjarnadóttir. 2020. Kvistur 2.0: a BiLSTM compound splitter for Icelandic. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3991–3995, Marseille, France. European Language Resources Association.
- Hendrik De Smet. 2005. A corpus of Late Modern English texts. *ICAME Journal*, 29(2005):69–82.
- Sigrún Helgadóttir, Hrafn Loftsson, and Eiríkur Rögnvaldsson. 2014. Correcting errors in a new gold standard for tagging Icelandic text. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14)*, pages 2944–2948, Reykjavik, Iceland. European Language Resources Association.
- Svanhvít Lilja Ingólfssdóttir, Hrafn Loftsson, Jón Friðrik Daðason, and Kristín Bjarnadóttir. 2019. Nefnir: A high accuracy lemmatizer for Icelandic. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 310–315, Turku, Finland. Linköping University Electronic Press.
- Atli Jasonarson, Steinþór Steingrímsson, Einar Freyr Sigurðsson, Árni Davíð Magnússon, and Finnur Ágúst Ingimundarson. 2023. Generating errors: OCR post-processing for Icelandic.

- In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 286–291, Tórshavn, Faroe Islands. University of Tartu Library.
- Haukur Páll Jónsson, Hrafn Loftson, and Steinþór Steingrímsson. 2021. ABLTagger (PoS) - 3.0.0. CLARIN-IS.
- Hannah Kermes, Stefania Degaetano-Ortlieb, Ashraf Khamis, Jörg Knappen, and Elke Teich. 2016. The Royal Society Corpus: From uncharted data to corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association.
- Vladimir Iosifovich Levenshtein. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8):707–710.
- Eiríkur Rögnvaldsson, Anton Karl Ingason, Einar Freyr Sigurðsson, and Joel Wallenberg. 2012. The Icelandic Parsed Historical Corpus (IcePaHC). In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, pages 1977–1984, Istanbul, Turkey. European Language Resources Association.
- Eiríkur Rögnvaldsson and Sigrún Helgadóttir. 2011. Morphological tagging of Old Icelandic texts and its use in studying syntactic variation and change. In *Language Technology for Cultural Heritage: Selected Papers from the LaTeCH Workshop Series*, pages 63–76, Berlin/Heidelberg. Springer.
- Gerold Schneider, Eva Pettersson, and Michael Percil-lier. 2017. Comparing rule-based and SMT-based spelling normalisation for English historical texts. In *Proceedings of the NoDaLiDa 2017 Workshop on Processing Historical Language*, pages 40–46, Gothenburg. Linköping University Electronic Press.
- Steinþór Steingrímsson, Sigrún Helgadóttir, Eiríkur Rögnvaldsson, Starkaður Barkarson, and Jón Guðnason. 2018. Risamálheild: A very large Icelandic text corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association.
- Steinþór Steingrímsson, Örvar Kárasen, and Hrafn Loftsson. 2019. Augmenting a BiLSTM tagger with a morphological lexicon and a lexical category identification step. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1161–1168, Varna, Bulgaria. INCOMA Ltd.
- Ásta Svavarsdóttir, Sigrún Helgadóttir, and Guðrún Kvaran. 2014. Language resources for early Modern Icelandic. In *Proceedings of Language Resources and Technologies for Processing and Linking Historical Documents and Archives – Deploying Linked Open Data in Cultural Heritage*, pages 19–25, Reykjavik, Iceland.
- Gongbo Tang, Fabienne Cap, Eva Pettersson, and Joakim Nivre. 2018. An evaluation of neural machine translation models on historical spelling normalization. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1320–1331, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
- 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 30 (NIPS 2017)*, pages 5999–6009, Long Beach, California.
- A. Vatri and B. McGillivray. 2018. The Diorisis Ancient Greek Corpus: Linguistics and literature. *Research Data Journal for the Humanities and Social Sciences*, 3(1):55 – 65.
- Loirella Viola. 2021. ChronicItaly and ChronicItaly 2.0: Digital heritage to access narratives of migration. *International Journal of Humanities and Arts Computing*, 15(1–2).
- Joel C. Wallenberg, Anton Karl Ingason, Einar Freyr Sigurðsson, and Eiríkur Rögnvaldsson. 2024. IcePaHC 2024.03 – A significant treebank upgrade. In *CLARIN Annual Conference Proceedings*, pages 168–171, Barcelona, Spain. ISSN 2773-2177 (online).
- Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306.

fine-tuned models on a variation of human annotated and automatically annotated test sets. Contrarily to the majority of previous studies, which employ the romanized version of the language, our setup focuses on segmenting Inuktitut written in Inuktitut syllabics. By sharing our best performing model, we hope to inspire others to also conduct their research on Inuktitut written in syllabics without romanizing the language first. Our main contributions are:

1. We show the potential of deploying pre-trained LLMs for surface-level morphological segmentation of Inuktitut compared to previous approaches.
2. We encourage more research to be done on down-stream NLP tasks for Inuktitut written in syllabics by making our model available¹.

2 Background and related work

There are plenty of methods dealing with morphological segmentation. Here we mention a few related to our work. Creutz and Lagus (2002) introduced an unsupervised probabilistic morpheme identifying method that has seen widespread use, with many related projects following their lead (Kohonen et al., 2010; Smit et al., 2014). More recently, Eskander et al. (2020) introduced MorphAGram, another unsupervised approach based on adaptor grammars (Johnson et al., 2006). Semi-supervised methods incorporating conditional random fields have also been proposed (Ruokolainen et al., 2014), as well as fully supervised ones (Cotterell et al., 2015). Additionally, there have been numerous neural approaches (Wang et al., 2016; Micher, 2017; Kann et al., 2018) using various model architectures. Recently, Pranjic et al. (2024) leveraged off pre-trained LLMs to segment words by turning morphological segmentation into a binary classification task. They displayed the effectiveness of their approach for a number of languages in a low-resource setting. Additionally, surface-level segmentation as a community task has also been highlighted during the 2005 to 2010 Morpho Challenges (Kurimo et al., 2010) and for a few low-resource languages in the shared task LowResourceEval-2019 (Klyachko et al., 2020).

¹Available here: <https://huggingface.co/matsten/Glot500-m-iuseg>

2.1 Previous approaches for segmenting Inuktitut

The UQAILAUT Inuktitut Morphological Analyzer (Farley, 2009) is an openly available morphological analyzer for the language, developed at the National Research Council of Canada (NRC). The analyzer is a finite state transducer that makes use of hand-crafted rules to return both a surface-level morphological segmentation of an input word, and the lemma of each individual morpheme. The segmentations returned are not always unambiguous since Inuktitut words can often be correctly segmented in many ways and, consequently, for many words, more than one segmentation is returned. Unfortunately, the analyzer suffers from a flaw in that for many words, it does not return any decompositions at all, making it rather unreliable to use as a pre-processing tool for downstream tasks. In an effort to cover for words that UQAILAUT cannot process, Micher (2017) annotated more training data from the Nunavut Hansard Inuktitut-English Parallel Corpus 3.0 (Joanis et al., 2020) using the same analyzer to train a Segmental Recurrent Neural Network (SRNN) (Kong et al., 2016) for both segmentation and tagging of morpheme specific information. Le and Sadat (2020) took a different approach and deployed a bidirectional Long-Short Term Memory (LSTM) incorporating pre-trained embeddings for Inuktitut. Roest et al. (2020) trained a transformer (Vaswani et al., 2017) based model and combined it with *UQAILAUT* and BPE to form a 3-step method to segment the language. More recently, Khandagale et al. (2022) extended their adaptor grammar based tool MorphAGram with expert-based linguistic priors for morphological segmentation of Inuktitut.

3 Methodology and experimental setup

3.1 Model

For all of our experiments, we utilize Glot500-m (Imani et al., 2023), a multilingual LLM covering more than 500 languages, many of which can be considered low in resources. It builds upon the XLM-R-base multilingual model (Conneau et al., 2020) by extensively extending its vocabulary from 250K tokens to 401K, and through continued training with a masked language modelling objective. It was trained on Glot500-c², a

²Available here as a Huggingface dataset: <https://huggingface.co/datasets/cis-lmu/Glot500>

subset of the larger Glot2000-c corpus, containing roughly 126GB of text covering more than 400 languages, including Inuktitut. The model was evaluated on a diverse set of tasks and displayed great improvements on the newly introduced languages and also performs equal to, or better, than XLM-R-base on already seen languages.

We make use of the Inuktitut side of the Nunavut Hansard Inuktitut-English Parallel Corpus 3.0³ (Joanis et al., 2020), which contains around 8M Inuktitut words worth of debate proceedings from the Legislative Assembly of Nunavut. After running the recommended accompanying spelling normalization scripts, we extract each unique word and end up with a vocabulary of approximately 1,1M unique words, which we automatically annotate using the UQAILAUT analyzer (Farley, 2009). For each successfully analyzed word, it returns either a single or many possible surface-level morphological decompositions. Similarly to the reasoning by Micher (2017); Roest et al. (2020), we assume that words with single decompositions are the least ambiguous and therefore the most correctly labeled words. Roest et al. (2020) even show that training their transformer based segmenter on fewer amounts of unambiguous word segmentations is preferred compared to training on many ambiguous ones. We therefore follow their steps. Since Glot500-m has seen large parts of the Nunavut Hansard corpus during pre-training, we make sure that the train, validation and test/evaluation splits are divided in such a way that there are no unique words in the test/evaluation and validation split that also occur in the Glot500-c dataset. Of a total of 54,138 unambiguously segmented words, 45,231 are used for training, 3102 for validation and 3102 for test/evaluation. We refer to this test/evaluation set as the *test* set. In order to compare our approach to UQAILAUT’s performance, we also evaluate our approach on another dataset separate from our initial test set. This dataset⁴, referred to as the *gold* set, consists of around 1K hand-annotated Inuktitut words based on the most frequently occurring

⁴<https://github.com/LowResourceLanguages/InuktitutComputing/blob/master/Inuktitut-Java/ressources/goldstandardHansard.txt>

3.3 Turning segmentation into a binary classification task

We fine-tune our models on binary classification tasks derived from the annotated Inuktitut words described in Section 3.2 using LLMSegm (Pranjic et al., 2024) using the original code⁵. LLMSegm derives binary classification tasks from a word by introducing a custom morpheme boundary token, represented here as “@”, that is inserted into a word between two characters. This is repeated for each unique position between two characters in the word forming $n - 1$ tasks where n is the number of characters in the word and the task is to predict whether “@” is positioned at a true morpheme boundary (see Figure 2).

Figure 2: Visualization of the tasks derived from an Inuktitut word, where the morpheme separator token denoted by “@” is inserted between each unique position between two characters. The task then becomes to predict whether this is a True (1) or False (0) morpheme boundary.

Additionally, prepended to each individual task is the same untouched word in its entirety, immediately followed by another custom token called the word boundary token, represented in text as “ \ddagger ”, effectively separating the prepended word and

⁵Original code is available here: <https://github.com/sharpsy/llm-morphological-segmenter>

the given task (see Figure 3).

| Task | Input | Label |
|---|--|-------|
| $\langle @^{\text{c}} \text{a}^{\text{c}} \text{J} \rangle$ | $\langle^{\text{c}} \text{a}^{\text{c}} \text{J} \rangle \ddagger \langle @^{\text{c}} \text{a}^{\text{c}} \text{J} \rangle$ | 0 |

Figure 3: Visualization of the full model input for a single example prediction. Here “ \ddagger ” represents the word boundary token and “@” the morpheme boundary token.

By doing this, Pranjić et al. (2024) hope to prevent the loss of information from the tokens that the pre-trained model’s tokenizer normally would split the word into. By additionally including the untouched word, all original tokens are guaranteed to be retained in the input since the tokenizer will be forced to split any tokens in its vocabulary that bridges across “@”. We experiment both with and without this addition by performing minimal alterations to the original code. This extra prepended word will henceforth be referred to as the *supporting word*.

3.4 Working with syllabics

We work with Inuktitut written in syllabics for two main reasons. Firstly, it is necessary since Glot500-m was fine-tuned on Inuktitut text written in syllabics. Secondly, we hypothesize that working with Inuktitut written in syllabics, as opposed to romanized Inuktitut, might be more beneficial when utilizing LLMSegm given how each input word is turned into $n - 1$ classification tasks. Since many of the syllabic characters often equate to two or sometimes even three roman characters when transcribing, the average romanized Inuktitut word often contains many more characters than the same word written in syllabic characters. Consequently, more tasks would be derived from the romanized word, which on the one hand would mean more total training samples, but among these, some might be less relevant. We say this on the basis of observations from transcription experiments⁶ we do to and from syllabics. We take notice that the vast majority of morpheme boundaries in the romanized version of the language occur between characters, or clusters of characters, that would normally be transcribed into separate syllabic characters in the equivalent transcription

⁶We transcribe using Yudit: <https://yudit.org/>

of the same word. By working with syllabics, we thus eliminate segmentation tasks that would otherwise be derived from between roman characters that are normally represented by the same single syllabic character (see Figure 4). We deem these tasks less relevant since, according to our observations, morpheme boundaries are less likely to occur between these characters.

| Word | Truth | Potential segmentations |
|---|---|---|
| $\text{L}^{\text{b}} \text{d}^{\text{b}} \text{J}^{\text{c}}$ | $\text{L}^{\text{b}} \text{d}^{\text{b}} @ \text{J}^{\text{c}}$ | $\text{L} @^{\text{b}} @ \text{d} @^{\text{b}} @ \text{J} @^{\text{c}}$ |
| makkuktut | makkuk@tut | ma@k@ku@k@tu@t |

Figure 4: The syllabic version of the language allows us to avoid deriving tasks such as classifying whether a morpheme boundary, denoted by “@”, is present between “m” and “a”, “k” and “u”, and “t” and “u”. This is because the character clusters “ma”, “ku” and “tu” are represented as one syllabic character each, and therefore an internal boundary between them is unlikely.

This way, not only do we clear our total task pool of these hypothetically less relevant tasks, but we also create a more balanced dataset with a more evenly distributed true-to-false label ratio, as opposed to if we stick with the romanized version of the language. We calculate that out of all the tasks derived from our syllabic train set, roughly 41% are labeled as true while the rest are false. We estimate that the same train set in roman characters would have a much lower ratio of roughly 23% true labels. How effective our reasoning is will however have to be left for future efforts.

3.5 Model fine-tuning

Using the training data described in Section 3.2, we fine-tune Glot500-m for classification using LLMSegm by following the original paper (Pranjić et al., 2024). We utilize the same hyperparameters of device batch size of 256, learning rate of $2e-5$, weight decay of 0.01, 20 warm up steps and AdamW optimizer (Loshchilov and Hutter, 2019). Unlike the original paper, we also fine-tune a second model without the supporting word to investigate how this affects training and later performance. For each fine-tuning set up, we train 10 separate models on randomly sampled variations of the original training data (with replacement) and pick the best performing one for evaluation.

We call the model trained without the supporting word *Glott500-m-iuseg-n* and the one with the supporting word *Glott500-m-iuseg-s*. All model training is done using 4x Nvidia A100 GPUs.

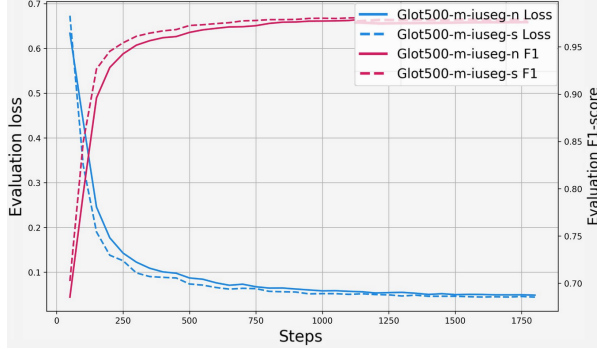


Figure 5: The evaluation loss and F1-score for (*Glott500-m-iuseg-n*) and (*Glott500-m-iuseg-s*).

During training, we take notice that the model training with the supporting word improves slightly faster than the model training without, both in terms of evaluation loss and evaluation F1-score (see Figure 5). This suggests that the tokens of the original unsegmented word produced by the tokenizer might indeed help retain valuable linguistic information from the pre-training that aids the fine-tuning process.

3.6 Evaluation

We evaluate our models on the two evaluation sets described in Section 3.2 (test and gold) and report back F1-score based on the difference between predicted morpheme boundaries and the actual boundaries. Much like (Kann et al., 2018; Roest et al., 2020; Pranjic et al., 2024), we additionally complement our F1-score by reporting the accuracy score calculated as the proportion of all words where every morpheme boundary was correctly predicted. We then end up with two complementary metrics, one calculated at morpheme-level and one at word-level. For comparison, we treat the Glott500-m (Imani et al., 2023) tokenizer as our baseline and also compare our results to previous studies where it is applicable. Due to the UQAILAUT analyzer’s tendency to fail when presented with certain words, we also evaluate a combined custom setup where our best performing model processes these failed words. We call this setup UQAILAUT+.

4 Results & discussion

We present our results in Table 1 and compare where possible to the following: *AG-SS* (Khandagale et al., 2022), *Trf. (45K single)* and *3-step* (Roest et al., 2020), *LSTM* with pre-trained embeddings (Le and Sadat, 2020), *SRNN CG* (Micher, 2017) and *UQAILAUT* (Farley, 2009). Our fine-tuned models *Glott500-m-iuseg-n* and *Glott500-m-iuseg-s* show the potential of our chosen methods compared to previous neural approaches in terms of F1-score and accuracy. Both of our models achieve a worse accuracy on the gold set, albeit higher F1, compared to the *3-step* setup.

| Model/setup | Test | | Gold | |
|---------------------------|-------------|-------------|-------------|-------------|
| | F1 | Acc. | F1 | Acc. |
| <i>Glott500-m tok.</i> | 0.59 | 0.04 | 0.42 | 0.18 |
| <i>AG-SS</i> | - | - | 0.60* | - |
| <i>Trf. (45K single)</i> | - | - | 0.68 | 0.54 |
| <i>3-Step</i> | - | - | 0.74 | 0.70 |
| <i>LSTM</i> | 0.75* | - | - | - |
| <i>SRNN CG</i> | 0.95* | - | - | - |
| <i>Glott500-m-iuseg-n</i> | 0.98 | 0.89 | 0.85 | 0.61 |
| <i>Glott500-m-iuseg-s</i> | 0.98 | 0.90 | 0.87 | 0.66 |
| <i>UQAILAUT</i> | - | - | 0.92 | 0.74 |
| <i>UQAILAUT+</i> | - | - | 0.95 | 0.81 |

Table 1: F1-score and accuracy scores from our models compared to previous studies. “-” indicates that evaluation metrics for the particular dataset were never reported or that they can not be reported. “*” next to a score indicated that the score was reported on a variation of the same dataset compared to what was used for evaluation in this study.

Worth noting is that where Micher (2017) choose 1K unambiguous samples annotated by UQAILAUT as their test set and Le and Sadat (2020) use 250 sentences as their test, we select as many unambiguous samples as possible who’s exact word form does not also appear in the training data of the Glott500-m model for a total of 3102. Hence, they are all evaluated on different amounts of words, and most likely also different words, from the Nunavut Hansard corpus (Joanis et al., 2020). Our model *Glott500-m-iuseg-n* slightly underperforms *Glott500-m-iuseg-s* trained using the supporting word. This would suggest that there is some benefit to including the supporting word not

only during training, but also during evaluation, possibly due to a retention of information from pre-training. This is also implied to be the case, since the Glot500-m tokenizer’s decent F1-score hints to the existence of some underlying knowledge of how to segment Inuktitut words, despite it returning very few fully correctly segmented words. None of the neural approaches alone outperform the UQAILAUT analyzer in terms of F1-score and accuracy, even though *Glot500-m-iuseg-s* is close. The combined setup UQAILAUT+ however, achieves the highest score on the gold set. Even though this setup does not improve F1-score too much, it improves accuracy by a not insignificant amount.

4.1 Oversegmentation

When examining the predictions on the two evaluation sets by our best performing model, we take notice of its tendency to oversegment words containing fewer than 4 true segmentations, peaking at words with 0 (see Figure 6). Going from 4 to 8 true segmentations per word, our best model achieves a more stable predicted-segmentations-per-word to true-segmentations-ratio on the test set, but seemingly underpredicts on the gold set for words in the same range.

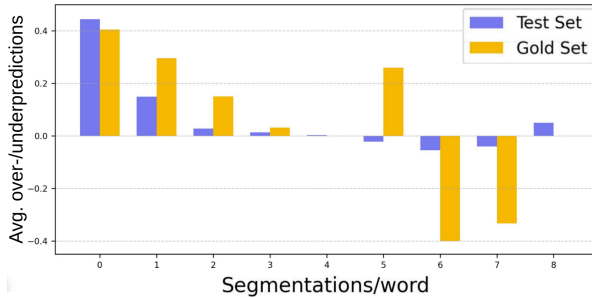


Figure 6: The average amount of morpheme boundaries over-/under predicted by *Glot500-m-iuseg-s* (y-axis) for words with n true segmentations from the test and gold set (x-axis).

Additionally, by calculating isolated F1-scores on predictions for words with 0-1 true segmentations, we see that our model performs much worse in this range compared to F1-scores in all the other ranges (see Figure 7). This underperformance is also reflected in the drop in F1-score between evaluations on the test set and the gold set, going from 0.98 to 0.87, since the gold set is made up of around 60% words in the range of 0-1 segmenta-

tions per word. The fact that our model saw many more words with segmentations in the range of 2-5 compared to the range 0-1 during fine-tuning might help explain why our model performs worse for these words. In fact, the average number of segmentations per word in our train set is much higher than in the gold set, as displayed in Table 2.

| average | train | test | gold |
|-------------------|-------|------|------|
| <i>seg./word</i> | 3.3 | 3.5 | 1.6 |
| <i>char./word</i> | 9.2 | 9.7 | 6.3 |

Table 2: Average true segmentations and syllabic characters per word in the train, test and gold set.

This suspicion is also supported by the higher F1-scores for words with true segmentations ranging from 2-5. Further building on this argument, the way the LLMSegm tool turns each annotated word into $n - 1$ segmentation tasks amplifies this training imbalance, as words with fewer segmentations typically contain fewer characters. This means that our model will see longer words many more times compared to shorter words. For this reason, we try to mitigate this imbalance by fine-tuning additional models where we upsample words in the segmentation-per-word range of 0-1 by 2x and 3x in the training data but with no positive effect on performance.

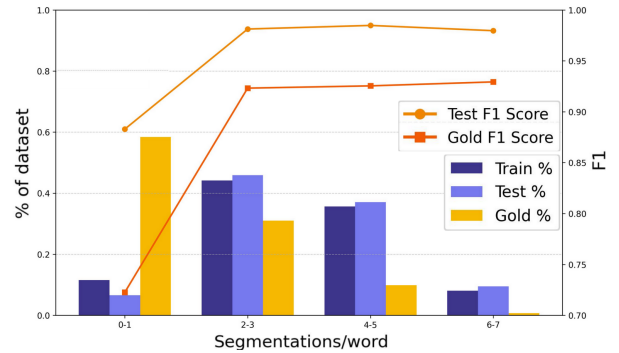


Figure 7: The percentage of words in the dataset that contain a certain amount of segmentations per word, as well as F1-score performance of *Glot500-m-iuseg-s* on words in each individual bracket for both the test and gold set.

Ignoring our model’s struggle with shorter words, we have two possible explanations for why our models perform worse overall on the hand an-

notated gold set than on the test set. Since both the training data and our test set were automatically annotated by UQAILAUT that itself does not score perfectly on the gold set, we can only assume that some of the training and test data were also incorrectly annotated. This might have inflated the scores on the test set compared to the gold set, and might also mean that our model will make the same mistakes when deployed as a pre-processing tool. We also know that the gold set contains 1K of the most frequent words in the Nunavut Hansard corpus, while our model was fine-tuned on unique words where word frequency was not taken into consideration.

4.2 UQAILAUT issues

As mentioned previously, the UQAILAUT analyzer is unable to produce decompositions for many Inuktitut words. This is despite it outperforming all other setups. We are unsure of the exact cause of the UQAILAUT analyzer’s inability to process certain words, but a quick look at these failure cases suggest that it might have to do with spelling inconsistencies and or not enough coverage in its hand-crafted rules to account for these. This might in turn explain why in the UQAILAUT+ setup, our model was able to correctly process a few words where UQAILAUT fails since spelling inconsistencies do not automatically result in a failed attempt thanks to the more dynamic nature of our neural model. However, due to the small evaluation dataset, it is not possible to draw any definitive conclusions.

When evaluating only the UQAILAUT analyzer on the gold set, we take notice that it fails to return any decompositions at all for approximately 11% of the words. However, when annotating the unique words from the Nunavut Hansard corpus to create our dataset, we note that, much like the observations made by Micher (2017), this percentage increases to approximately 30%. This suggests that, despite its high scores on the gold set, UQAILAUT is unfit to pre-process real world texts for downstream NLP tasks on its own since some very long words would be left unsegmented. Further, this suggests a performance decrease in a scenario where we have access to more human annotated gold data for evaluation that contains rarer words and not just the 1K most common ones. In fact, we calculate that only 20% of all word forms in the Nunavut Hansard corpus occur more than

once and only 11% more than twice. This abundance of unique words in Inuktitut further highlights the importance of continued research in the field to ultimately benefit downstream NLP tasks.

5 Conclusion

We contribute to ongoing research focusing on the polysynthetic language Inuktitut by fine-tuning and sharing a Glot500-m LLM for binary classification of morpheme boundaries. Our best model shows promising results when comparing to previous efforts, despite struggling to segment words with fewer true segmentation boundaries. We also show the potential of deploying existing pre-trained LLMs using LLMSegm even for under-resources polysynthetic languages without the need to train anything from scratch. Additionally, we further encourage future studies on downstream NLP tasks for Inuktitut written in syllabics. In future efforts, we intend to improve the performance of our model, as well as investigate its potential as a pre-processing tool for downstream NLP tasks such as machine translation.

6 Limitations

The main limitation with LLMSegm is the fact that it completely relies on the existence of a pre-trained model that has seen the target language during pre-training, which, ironically, excludes many of the world’s lowest resource languages. Additionally, being a low-resource language, Inuktitut suffers from a lack of well-balanced human segmented gold data for both training and evaluation. Thus, it is not possible to draw solid conclusions based on evaluation on the only available gold set, and only further highlights the need for more such data. Our method also does not take alternative segmentations into consideration, but we still believe that our model can be used as a pre-processing tool to benefit downstream performance. Further, the accuracy, as reported by Roest et al. (2020), Pranjic et al. (2024), and now also by us, is not an ideal metric for evaluating a segmenter for polysynthetic languages. Since this definition of accuracy gives the same weight to words containing different amounts of segmentations, a correctly predicted decomposition of a word containing 1 true segmentation is valued higher than a word containing 8 true segmentations, where the setup only successfully predicts 7.

Acknowledgments

This work was supported by the European Union’s Horizon 2020 research and innovation program under grant agreement No 101135671 (TrustLLM). It is co-financed by the EU-ROCC2 project, funded by the European High-Performance Computing Joint Undertaking (JU) and EU/EEA states under grant agreement No 101101903. Parts of this were also supported by the European Digital Innovation Hub (EDIH) of Iceland (EDIH-IS), partially funded by the Digital Europe Programme. The authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding this project by providing computing time on the GCS Supercomputer JUWELS at the Jülich Supercomputing Centre (JSC).

References

- 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.
- Ryan Cotterell, Thomas Müller, Alexander Fraser, and Hinrich Schütze. 2015. Labeled morphological segmentation with semi-Markov models. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 164–174, Beijing, China. Association for Computational Linguistics.
- Mathias Creutz, Teemu Hirsimäki, Mikko Kurimo, Antti Puurula, Janne Pykkönen, Vesa Siivola, Matti Varjokallio, Ebru Arisoy, Murat Saraçlar, and Andreas Stolcke. 2007. Morph-based speech recognition and modeling of out-of-vocabulary words across languages. *ACM Trans. Speech Lang. Process.*, 5(1).
- Mathias Creutz and Krista Lagus. 2002. Unsupervised discovery of morphemes. In *Proceedings of the ACL-02 Workshop on Morphological and Phonological Learning - Volume 6*, MPL ’02, page 21–30, USA. Association for Computational Linguistics.
- Christopher Dyer, Smaranda Muresan, and Philip Resnik. 2008. Generalizing word lattice translation. In *Proceedings of ACL-08: HLT*, pages 1012–1020, Columbus, Ohio. Association for Computational Linguistics.
- Ramy Eskander, Francesca Callejas, Elizabeth Nichols, Judith Klavans, and Smaranda Muresan. 2020. MorphAGram, evaluation and framework for unsupervised morphological segmentation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 7112–7122, Marseille, France. European Language Resources Association.
- Benoît Farley. 2009. The uqailaut project. <https://www.inuktitutcomputing.ca/index.php>. Accessed: 2024-07-15.
- Ayyoob Imani, Peiqin Lin, Amir Hossein Kargaran, Silvia Severini, Masoud Jalili Sabet, Nora Kassner, Chunlan Ma, Helmut Schmid, André Martins, François Yvon, and Hinrich Schütze. 2023. Glot500: Scaling multilingual corpora and language models to 500 languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1082–1117, Toronto, Canada. Association for Computational Linguistics.
- Eric Joanis, Rebecca Knowles, Roland Kuhn, Samuel Larkin, Patrick Littell, Chi-kiu Lo, Darlene Stewart, and Jeffrey Micher. 2020. The Nunavut Hansard Inuktitut–English parallel corpus 3.0 with preliminary machine translation results. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2562–2572, Marseille, France. European Language Resources Association.
- Mark Johnson, Thomas Griffiths, and Sharon Goldwater. 2006. Adaptor grammars: A framework for specifying compositional nonparametric bayesian models. In *Advances in Neural Information Processing Systems*, volume 19. MIT Press.
- Katharina Kann, Jesus Manuel Mager Hois, Ivan Vladimir Meza-Ruiz, and Hinrich Schütze. 2018. Fortification of neural morphological segmentation models for polysynthetic minimal-resource languages. 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 47–57, New Orleans, Louisiana. Association for Computational Linguistics.
- Sujay Khandagale, Yoann Léveillé, Samuel Miller, Derek Pham, Ramy Eskander, Cass Lowry, Richard Compton, Judith Klavans, Maria Polinsky, and Smaranda Muresan. 2022. Towards unsupervised morphological analysis of polysynthetic languages. 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 2: Short Papers)*, pages 334–340, Online only. Association for Computational Linguistics.
- Elena Klyachko, Alexey Sorokin, Natalia Krizhanovskaya, Andrew Krizhanovsky, and Galina Ryazanskaya. 2020. Lowresourceeval-2019: a shared task on morphological analysis for low-resource languages.

- Oskar Kohonen, Sami Virpioja, and Krista Lagus. 2010. Semi-supervised learning of concatenative morphology. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 78–86, Uppsala, Sweden. Association for Computational Linguistics.
- Lingpeng Kong, Chris Dyer, and Noah A. Smith. 2016. Segmental recurrent neural networks.
- Taku Kudo and John Richardson. 2018. Sentence-Piece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.
- Mikko Kurimo, Sami Virpioja, Ville Turunen, and Krista Lagus. 2010. Morpho challenge 2005-2010: Evaluations and results. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 87–95, Uppsala, Sweden. Association for Computational Linguistics.
- Tan Ngoc Le and Fatiha Sadat. 2020. Low-resource NMT: an empirical study on the effect of rich morphological word segmentation on Inuktitut. In *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*, pages 165–172, Virtual. Association for Machine Translation in the Americas.
- Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.
- Mick Mallon. 2000. Inuktitut linguistics for technocrats. <https://www.inuktitutcomputing.ca/Technocrats/ILFT.php>. Accessed: 2024-07-15.
- Jeffrey Micher. 2017. Improving coverage of an Inuktitut morphological analyzer using a segmental recurrent neural network. In *Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 101–106, Honolulu. Association for Computational Linguistics.
- Marko Pranić, Marko Robnik-Šikonja, and Senja Polak. 2024. LLMSegm: Surface-level morphological segmentation using large language model. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10665–10674, Torino, Italia. ELRA and ICCL.
- Christian Roest, Lukas Edman, Gosse Minnema, Kevin Kelly, Jennifer Spenader, and Antonio Toral. 2020. Machine translation for English–Inuktitut with segmentation, data acquisition and pre-training. In *Proceedings of the Fifth Conference on Machine Translation*, pages 274–281, Online. Association for Computational Linguistics.
- Teemu Ruokolainen, Oskar Kohonen, Kairit Sirts, Stig-Arne Grönroos, Mikko Kurimo, and Sami Virpioja. 2016. A comparative study of minimally supervised morphological segmentation. *Computational Linguistics*, 42(1):91–120.
- Teemu Ruokolainen, Oskar Kohonen, Sami Virpioja, and Mikko Kurimo. 2014. Painless semi-supervised morphological segmentation using conditional random fields. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers*, pages 84–89, Gothenburg, Sweden. Association for Computational Linguistics.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.
- Peter Smit, Sami Virpioja, Stig-Arne Grönroos, and Mikko Kurimo. 2014. Morfessor 2.0: Toolkit for statistical morphological segmentation. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 21–24, Gothenburg, Sweden. Association for Computational Linguistics.
- 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.
- Linlin Wang, Zhu Cao, Yu Xia, and Gerard de Melo. 2016. Morphological segmentation with window lstm neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30.
- Adam Wiemerslage, Miikka Silfverberg, Changbing Yang, Arya McCarthy, Garrett Nicolai, Eliana Colunga, and Katharina Kann. 2022. Morphological processing of low-resource languages: Where we are and what’s next. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 988–1007, Dublin, Ireland. Association for Computational Linguistics.

The Devil’s in the Details: the Detailedness of Classes Influences Personal Information Detection and Labeling

Maria Irena Szawerna[†], Simon Dobnik[‡], Ricardo Muñoz Sánchez[‡], Elena Volodina[†]

[†]Språkbanken Text, SFS, University of Gothenburg, Sweden

[‡]CLASP, FLoV, University of Gothenburg, Sweden

mormor.karl@svenska.gu.se

[†]{maria.szawerna, ricardo.munoz.sanchez, elena.volodina}@gu.se

[‡]simon.dobnik@gu.se

Abstract

In this paper, we experiment with the effect of different levels of detailedness or granularity — understood as i) the number of classes, and ii) the classes’ semantic depth in the sense of hypernym and hyponym relations — of the annotation of Personally Identifiable Information (PII) on automatic detection and labeling of such information. We fine-tune a Swedish BERT model on a corpus of Swedish learner essays annotated with a total of six PII tagsets at varying levels of granularity. We also investigate whether the presence of grammatical and lexical correction annotation in the tokens and class prevalence have an effect on predictions. We observe that the fewer total categories there are, the better the overall results are, but having a more diverse annotation facilitates fewer misclassifications for tokens containing correction annotation. We also note that the classes’ internal diversity has an effect on labeling. We conclude from the results that while labeling based on the detailed annotation is difficult because of the number of classes, it is likely that models trained on such annotation rely more on the semantic content captured by contextual word embeddings rather than just the form of the tokens, making them more robust against nonstandard language.

1 Introduction

Personal information is ubiquitous in many text genres, posing a unique challenge for those seeking to create and share corpora. While access to collections of texts is highly desirable from the perspective of researchers in fields such as linguistics, Natural Language Processing (NLP), or

digital humanities, the potential presence of clues indicating the identity of the writer or other natural persons makes them fall under the General Data Protection Regulation (GDPR, [Official Journal of the European Union, 2016](#)). The GDPR itself suggests potential solutions to the problem: de-identification methods such as anonymization — the “[c]omplete and irreversible removal [...] of any information that, directly or indirectly, may lead to a subject’s data being identified” — or pseudonymization, the “[p]rocess of replacing direct identifiers with pseudonyms or coded values,” for which there must exist a mapping between the original data and the pseudonyms, which is securely stored separately from the pseudonymized texts ([Lison et al., 2021](#)).

Both of these privacy-preserving procedures presuppose a stage where the Personally Identifiable Information (PII) found in the data is detected. While this can be done manually, it is time-consuming. While automatic approaches for both anonymization and pseudonymization have been proposed ([Lison et al., 2021](#)), [Szawerna et al. \(2024a\)](#) show that there appears to be very little uniformity in how researchers and corpus creators choose to classify PIIs. The taxonomies range in terms of granularity or detailedness, understood as the number of classes that PIIs are divided into and their semantic depth in terms of hypernym and hyponym relations (as in WordNet ([Miller, 1995](#))). For example, [Pilán et al. \(2022\)](#) utilize only one label, PERSON, to refer to elements such as names, surnames, nicknames, usernames, etc., which can be differentiated in other corpora (e.g. [Volodina et al. 2016, 2019](#); [Eder et al. 2020](#); [Alfalahi et al. 2012](#)). Very little work has been done on determining what level of granularity of PII annotation is the most suitable for subsequent removal or replacement of personal information.

It is worth noting that while the term *detection* often includes labeling in other research on

| General category | Corresponding detailed categories |
|------------------|--|
| personal_name | firstname_male, firstname_female, firstname_unknown, initials, middlename, surname |
| institution | school, work, other_institution |
| geographic | area, city, geo, country, place, region, street_nr, zip_code, foreign |
| transportation | transport_name, transport_nr |
| age | age_digits, age_string |
| date | date_digits, day, month_digit, month_word, year |
| other | phone_nr, email, url, personid_nr, account_nr, license_nr, other_nr_seq, extra, prof, edu, fam, sensitive, gen, def, pl |

Table 1: General and detailed categories in the SWELL PII taxonomy. Tags that can be combined with other categories and therefore were not included in the experiments are crossed out.

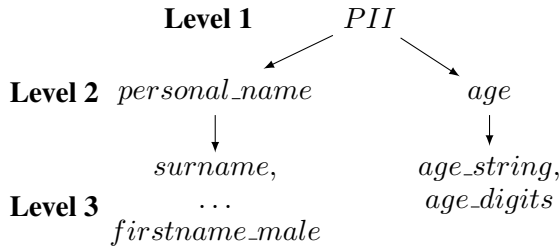


Figure 1: Hierarchical ontological structure of PII categories on the example of selected SWELL categories.

this topic, we choose to differentiate between the two: *PII detection* is the process of determining whether a text span constitutes a piece of Personally Identifiable Information, while *PII labeling* is assigning a PII span a specific class which describes the type of PII it is (this procedure often, by default, detects and assigns a specific PII class at the same time).

In this paper, we set out to investigate what the effect of the class granularity on PII detection and labeling in the learner essay domain. We run our experiments on a set of Swedish texts that are PII-annotated at varying levels of detailedness. A related notion is that of the categories’ ontological structure. As shown in Figure 1, the categories used in this experiment can be hierarchically arranged from the most general (level 1) to the most specific (level 3). Simultaneously, e.g. level 2 categories are semantically broader (include more semantically varied elements) than the more specific level 3 categories. How varied the contents of a category are could have an impact on how easy

it is to automatically detect. While we make an initial assumption that having a larger number of more specific labels means that they will be less internally diverse, labels in one tagset are not necessarily equally internally coherent.

In addition, we are curious to see how various factors pertaining to the class divisions (e.g. the class’s frequency) or the word tokens themselves (e.g. being ungrammatical) influence the performance. While improvement in terms of PII detection on the data with more specific annotation relative to the general one has been previously observed (Sierro et al., 2024), we expect multi-class classification to be more prone to error.

2 Prior Research

Data for research or training language models needs to be free from personal information to protect those who generate it, and the work on automatic de-identification methods, especially for texts belonging to domains other than medical or legal, has gained much traction in the recent years (Lison et al., 2021).

Much research has gone into testing what kinds of models perform best for PII detection or labeling. Eder et al. (2022) evaluate 9 different model architectures and embedding combinations on the PII-annotated corpus of German e-mails, CODE ALLTAG, reaching the best performance with a Transformer-based architecture and embeddings, optionally combined with noncontextual word embeddings. Papadopoulou et al. (2022, 2023) successfully utilize a combination of a generic Named Entity Recognition (NER) model with a gazetteer to detect and classify PIIs in English (the TAB CORPUS and a set of annotated

Wikipedia biographies) and employ privacy risk estimation methods to determine whether a span should be anonymized or not. [Grancharova and Dalianis \(2021\)](#) frame the closely related task of Protected Health Information (PHI) detection as a Named Entity Recognition and Classification (NERC) task and obtain good results on it using two BERT-type language models on Swedish medical data from the STOCKHOLM EPR PHI CORPUS. [Szawerna et al. \(2024b\)](#) also use models of this kind to detect PII in the SWELL corpus, a collection of learner essays in Swedish. Notably, they forego the labeling step, differentiating only between PII and non-PII tokens.

It is worth noting that all of the previously mentioned PII or PHI detection or labeling studies utilized different data, and only the texts used by [Papadopoulou et al. \(2022, 2023\)](#) — representing a vastly different domain and a more general tagset than the texts we work with — are openly available with the original PIIs in place. Additionally, all of the papers employed different categories for the labeling task. As [Szawerna et al. \(2024a\)](#) point out, differences between PII taxonomies employed in the de-identification of corpora can be quite considerable, not only in terms of class granularity but also class overlap. This may be motivated by the specific characteristics of the de-identified domains or the end goal: taxonomies used for pseudonymization seem to feature more classes than those intended for anonymization, likely because the class of the PII is later used to generate a suitable pseudonym. This leads to the results not being fully comparable. The TAB CORPUS features fewer, semantically more general classes (grouping together many different concepts into one category); it also lexical or grammatical correction annotation¹. This makes it unsuitable for addressing our research questions without a considerable amount of time going into manual reannotation.

However, it remains unclear how and to what extent the types of classes used in personal information detection affect the detection step itself. In [Szawerna et al. \(2024a\)](#) we consider a more detailed taxonomy more favorable, but we do not test that. We do, however, point out that what is per-

sonal is context-dependent and may vary between domains, so the choice of the labels can also depend on the domain. To the best of our knowledge, the only study that investigated whether a more diverse class division facilitates better PII detection is the one by [Sierro et al. \(2024\)](#). In this case, the authors adapted the TAB CORPUS by automatically translating it into Spanish and projecting the PII categories back into the text. They later re-annotated the corpus with refined, less ambiguous classes, leading to an increase in the number of classes. Notably, they also discard the MISC class, which is used to annotate very semantically diverse elements. They note an increase in performance on the dataset annotated using the refined tagset, which could be due to the new tagset being easier for their models to train on, but also due to manual re-annotation being more reliable than projection, and some information not being as revealing after translation.

3 Materials and Methods

3.1 Data

The data used in our experiments comes from the SWELL-PILOT (480 texts) and SWELL-GOLD corpora (502 texts) ([Volodina et al., 2016, 2019; Språkbanken Text, 2024b,a](#)), consisting of essays written by adult learners of Swedish as a second language (L2) at varying proficiency levels, with varied essay genres and topics. We chose to work with this data mainly because it is already PII-annotated with a hierarchical PII tagset and because its subset, SWELL-GOLD, features correction annotation which denotes e.g. grammatical variation in the text. The correction annotation was only used in evaluation, and our models were never overtly given that information.

While the released versions of the SWELL corpora² are pseudonymized, we utilize the texts in their original form with the unaltered PIIs in place. We preserve the aforementioned annotation of what spans contain personal information and of what kind. This annotation is done following the SWELL taxonomy ([Megyesi et al., 2018](#)), which consists of 38 types of PIIs (it also includes functional or morphosyntactic tags which we disregard for the sake of this experiment). Every PII token gets assigned an appropriate class and a number used for coreference resolution, which also helps

¹This kind of annotation indicates that a token is in some way at odds with the standard for a given language, e.g. it is misspelled, the wrong word is used, the wrong grammatical form is used, or it is a part of a grammatical construction that is unacceptable from the standard point of view.

²SWELL access can be requested at <https://sunet.artologik.net/gu/swell>

| Class | Bs | Is | Total |
|-------------------|-----|-----|-------|
| firstname.male | 234 | 0 | 234 |
| firstname.female | 289 | 0 | 289 |
| firstname.unknown | 49 | 0 | 49 |
| initials | 0 | 0 | 0 |
| middlename | 1 | 0 | 1 |
| surname | 49 | 2 | 51 |
| school | 44 | 25 | 69 |
| work | 2 | 0 | 2 |
| other.institution | 65 | 24 | 89 |
| area | 0 | 0 | 0 |
| city | 564 | 23 | 587 |
| geo | 17 | 0 | 17 |
| country | 400 | 1 | 401 |
| place | 93 | 19 | 112 |
| region | 37 | 2 | 39 |
| street.nr | 21 | 0 | 21 |
| zip.code | 7 | 2 | 9 |
| transport.name | 5 | 1 | 6 |
| transport.nr | 14 | 0 | 14 |
| age.digits | 82 | 0 | 82 |
| age.string | 12 | 0 | 12 |
| date.digits | 30 | 14 | 44 |
| day | 27 | 0 | 27 |
| month.digit | 9 | 0 | 9 |
| month.word | 46 | 0 | 46 |
| year | 53 | 0 | 53 |
| phone.nr | 7 | 0 | 7 |
| email | 10 | 0 | 10 |
| url | 0 | 0 | 0 |
| personid.nr | 0 | 0 | 0 |
| account.nr | 0 | 0 | 0 |
| license.nr | 0 | 0 | 0 |
| other.nr.seq | 169 | 1 | 170 |
| extra | 37 | 3 | 40 |
| prof | 12 | 2 | 14 |
| edu | 6 | 1 | 7 |
| fam | 464 | 3 | 467 |
| sensitive | 256 | 114 | 370 |

Table 2: Class counts for the detailed PII classes.

| Class | Bs | Is | Total |
|----------------|------|-----|-------|
| personal.name | 622 | 2 | 624 |
| institution | 111 | 49 | 160 |
| geographic | 1139 | 47 | 1186 |
| transportation | 19 | 1 | 20 |
| age | 94 | 0 | 94 |
| date | 165 | 14 | 179 |
| other | 961 | 124 | 1085 |

Table 3: Class counts for the general PII classes.

to define the edges of a PII span. These PII categories can be grouped into 7 general classes (as shown in Table 1). Therefore, the data can have the original SWELL classes (Specific), the overarching SWELL categories (General), or an even more general binary distinction whether the element is personal or not can be made (Basic; this corresponds more to a task of PII detection). It is worth noting that not all of the detailed SWELL classes are present in the data, and some were just theorized by the tagset creators to be possible. Many of the classes are also unlikely to span more than one token. The annotation can be modified to follow the inside-outside-beginning (IOB) schema or not include the distinction between beginning and inside (though the non-PII tokens are still marked as O in that case). This yields six different sets of classes that can be tested (henceforth Specific IOB, Specific, corresponding to Level 3 in Figure 1; General IOB, General, corresponding to Level 2; Basic IOB, Basic, corresponding to Level 1; see also Appendix A for a practical example).

When constructing our samples, we want to include as much context as possible, as we believe that the personal nature of a text span is context-dependent. Many of the essays exceed the maximum input size allowed by the BERT model that we are using.³ We therefore split such essays into several chunks. Such a chunk has a maximum size of 512 BERT sub-word tokens. We ensure that our data consists of equally many samples containing at least one token belonging to a PII category as samples without any and that chunks of the same essay always appear in the same data split. This yields a collection of samples with 217,430 non-PII tokens and 3,348 PII tokens (3,111 B-tokens and 237 I-tokens). The exact counts for the Specific and General class sets can be found in Table 2 and Table 3, respectively. It is worth noting that some classes in the detailed set are not present in the data at all, and are only theoretically permitted by the taxonomy. Having considered discarding some of the data to balance the classes, we have decided against that, since our dataset is small as is, and we are curious to see how the prevalence of certain PII classes influences their labeling.

³Unfortunately, Longformer or a similar model is not available for Swedish.

| Annotation type | Precision | Recall | F1 | F2 |
|-----------------|----------------------|----------------------|----------------------|----------------------|
| Specific IOB | 0.794 ± 0.028 | 0.709 ± 0.059 | 0.748 ± 0.042 | 0.724 ± 0.052 |
| Specific | 0.867 ± 0.020 | 0.733 ± 0.053 | 0.793 ± 0.036 | 0.756 ± 0.047 |
| General IOB | 0.788 ± 0.049 | 0.770 ± 0.061 | 0.770 ± 0.043 | 0.770 ± 0.053 |
| General | 0.858 ± 0.026 | 0.803 ± 0.059 | 0.828 ± 0.037 | 0.813 ± 0.050 |
| Basic IOB | 0.842 ± 0.021 | 0.796 ± 0.050 | 0.808 ± 0.037 | 0.800 ± 0.045 |
| Basic | 0.857 ± 0.019 | 0.817 ± 0.045 | 0.836 ± 0.028 | 0.824 ± 0.038 |

Table 4: Mean results ± standard deviation over the runs evaluated as detection (whether the token was detected as any PII class). Bold indicates the overall best scores. Italicized elements in bold are the best scores if the basic type of annotation were disregarded.

| Annotation type | Precision | Recall | F1 | F2 |
|-----------------|----------------------|----------------------|----------------------|----------------------|
| Specific IOB | 0.497 ± 0.090 | 0.539 ± 0.083 | 0.498 ± 0.086 | 0.519 ± 0.085 |
| Specific | 0.591 ± 0.051 | 0.569 ± 0.062 | 0.550 ± 0.065 | 0.558 ± 0.063 |
| General IOB | 0.719 ± 0.041 | 0.727 ± 0.057 | 0.714 ± 0.049 | 0.720 ± 0.054 |
| General | 0.806 ± 0.039 | 0.761 ± 0.062 | 0.770 ± 0.053 | 0.763 ± 0.059 |
| Basic IOB | 0.842 ± 0.021 | 0.796 ± 0.050 | 0.808 ± 0.037 | 0.800 ± 0.045 |
| Basic | 0.857 ± 0.019 | 0.817 ± 0.045 | 0.836 ± 0.028 | 0.824 ± 0.038 |

Table 5: Mean results ± standard deviation over the runs evaluated as labeling (whether the token was assigned the right class). Bold indicates the overall best scores. Italicized elements in bold are the best scores if the basic type of annotation were disregarded.

3.2 Model and Code

We take the model from Szawerna et al. (2024b) that reports the best results, the Swedish BERT developed by the National Library of Sweden⁴(Malmsten et al., 2020), which is based on the BERT architecture (Devlin et al., 2019), with a regular cross-entropy loss. This is confirmed by our own preliminary testing. Due to the model’s relatively small size and short fine-tuning time, it is possible to conduct cross-validation.

In order to fine-tune KB-BERT we utilize the code for token classification⁵ included in the Transformers library together with the model hosted on HuggingFace (Wolf et al., 2020). This code makes use of HuggingFace’s Trainer class to fine-tune a BERT model for classification by discarding its head and replacing it with a new classification head, which is what is trained for the classification task at hand, while other pre-trained knowledge does not get altered. The only notable change that we make to the default settings of this classification set-up is decreasing the batch size to 8. For each of our 6 sets of data (which differ

by annotation type) we conduct a 10-fold cross-validation.

For the rest of the preprocessing and evaluation we expand the code provided by Szawerna et al. (2024b) for working with SWELL data.⁶

3.3 Evaluation

For each of the runs, we obtain predictions on the held-out fold. We report the mean and the standard deviation across the 10 separate runs for each type of data. Due to the overwhelming prevalence of the non-PII tokens and following the example of e.g. Grancharova and Dalianis (2021), we report the means and standard deviations of the weighted averages of precision, recall, F1, and F2⁷ across all of the PII classes (excluding the scores for non-PII tokens). Consequently, precision reflects the models’ ability to avoid falsely flagging a word token as some PII class, whereas recall illustrates how well PII tokens can be detected instead of slipping through the cracks. The rationale behind reporting an F2 score is that it gives more weight to recall, and Berg and Dalianis (2020) consider recall to be a more important measure (as it reflects how many PII tokens were actually detected, which is a pri-

⁴[KB/bert-base-swedish-cased](#), henceforth KB-BERT.

⁵<https://github.com/huggingface/transformers/tree/main/examples/legacy/token-classification>

⁶<https://github.com/mormor-karl/the-d evils-in-the-details>

⁷ $F_2 = (1 + 2^2) * \frac{precision * recall}{(2^2 * precision) + recall}$

| Annotation type | Correction annotated | Misclassified | Correction-annotated and misclassified | % of misclassified tokens that are correction-annotated |
|-----------------|----------------------|---------------|--|---|
| Specific IOB | 14014 | 405 | 47 | 11.61% |
| Specific | 14014 | 407 | 52 | 12.78% |
| General IOB | 14014 | 334 | 64 | 19.16% |
| General | 14014 | 294 | 64 | 21.77% |
| Basic IOB | 14014 | 277 | 67 | 24.19% |
| Basic | 14014 | 255 | 65 | 25.49% |

Table 6: Counts of the correction-annotated tokens, tokens misclassified during testing (in the labeling task), and the overlap of the two groups per type of PII annotation. Note that the number of correction-annotated tokens does not change across the PII annotation types and that these results concern only the data from SWELL-GOLD, as SWELL-PILOT does not include correction annotations.

ority).

However, we want to highlight that a high precision score is important as well, as avoiding flagging innocuous tokens as PII is essential for preserving as much of the original text as possible, which affects its later usability in linguistic research or NLP applications. Additionally, we evaluate the results both in terms of labeling – whether a token was assigned the correct class – and detection – whether the token was correctly identified as non-PII or any of the PII classes. In the case of the basic-type annotation, these two evaluations are equivalent.

We conduct further analysis, the purpose of which is to study two aspects of label selection: (i) whether grammatical and lexical divergence from the standard has an effect on the labeling of personal information, (ii) how the number of labels used and the depth of their semantics affects the labeling. We approximate the former by analyzing the raw counts of correction-annotated tokens that were misclassified and what percentage of all misclassified tokens they constitute within each annotation type.

4 Results

The mean detection results and the standard deviation over the runs are presented in Table 4. The same is shown for labeling in Table 5.

Both tables show that the IOB-type annotation appears to be more difficult to predict. This is likely due to relatively few PIIs spanning more than one token, leading to the classifiers having more issues determining those boundaries; in our case both the IOB component and the class label have to match for a token to be counted as correctly classified. Yet another aspect worth men-

tioning is that in many cases (`firstname_male`, `month_word`, etc.) the original SWELL annotation is intended to describe only one token, whereas other classes (e.g. `school`) are likely to consist of more than one token (see Table 2 and Table 3). This means that the effect of the IOB annotation is negligible for most classes.

When it comes to PII detection (Table 4), two different kinds of annotation excel in different metrics. Using the Specific annotation leads to the best precision. However, in terms of recall and the F-scores, the Basic annotation performs better.

In the labeling task, the basic type of annotation excels in all evaluation metrics. In the case of Basic annotation, detection and labeling are the same. If we consider the types of annotation where these two tasks are different, then the runs with the best recall, F1, and F2 for detection are the ones fine-tuned on the general type of annotation. In the case of labeling, these runs would be the best on all of the evaluation metrics. This partly contradicts the findings of [Sierro et al. \(2024\)](#), who report that more detailed classes facilitate better PII detection and labeling. However, the detailed classes in their experiment were refined based on what they found to be ambiguous in the original tagset. In our case, we utilized an existing hierarchical tagset. The difference in granularity between General and Specific classes is also much larger, as [Sierro et al. \(2024\)](#) split the original classes into at most 2 new classes, while in our data one General class can correspond to as many as 12 Specific classes.

We can only examine the interplay between the identification of PIIs and the tokens that were labeled for grammatical or lexical errors in different tagsets (Table 6) on the basis of SWELL-GOLD, as SWELL-PILOT does not include any correc-

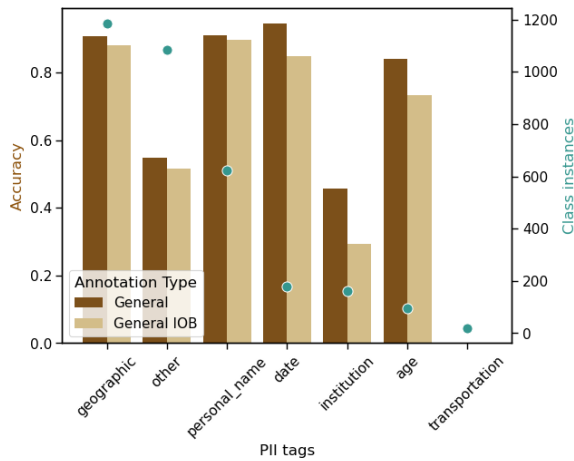


Figure 2: Per class prediction accuracy for the General and General IOB annotations (I and B is merged). The points illustrate the classes’ raw frequencies.

tion annotation. The correction annotations were not visible to the classifier during training, and instead we use them to identify the tokens that were judged to belong to a grammatically or lexically non-standard span. What appears to be influenced by the annotation type is the number of total misclassifications, the percentage of those that consists of correction-annotated tokens, and the raw counts of correction-annotated misclassified tokens. It is clear that the more diverse types of annotation lead to more misclassifications in general; however, there is a reverse trend when it comes to what percent of the misclassified tokens is also correction-annotated. It follows that more diverse annotation is less affected by errors than more general annotation. This could mean that the poorer performance noted for more detailed annotation is caused by the multi-class classification during labeling being inherently more difficult given the number of classes, but that the models learn to connect the more specific tokens better with the word embeddings and their contexts that represent the semantics of the text to determine that the span is a part of some PII. This is also partly reflected in the major improvement of the scores when the predictions are reinterpreted from labeling into detection (as the scores for Specific and Specific IOB then jump by 15 to 30 percentage points).

Figure 2 and Figure 3 show the per-class accuracy (disregarding the I and B distinction). Points indicating the number of instances of the respec-

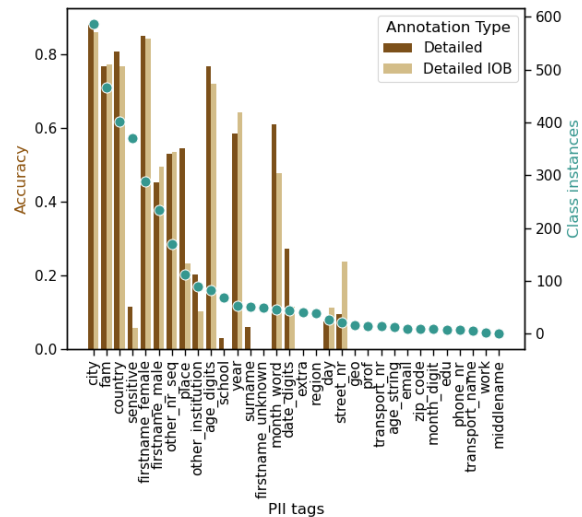


Figure 3: Per class prediction accuracy for the Specific and Specific IOB annotations (I and B is merged). The points illustrate the classes’ raw frequencies.

tive class in the data are overlaid atop the accuracy bar charts.

Figure 2 shows these statistics for the General and General IOB tagsets. For some of the classes, prevalence in the data correlates with accuracy – nearly 1200 tokens belong to the *geographic* class, which has high accuracy, while *institution*, with around 200 tokens, shows worse results and the extremely infrequent *transportation* class practically never gets correctly predicted. However, there are classes that diverge from this trend: despite having almost as many instances as *geographic*, *other* has noticeably lower accuracy, implying that they are difficult to predict. Less frequent classes like *personal_name*, *date*, and *age* achieve high accuracy scores despite not being as numerous as some other classes, indicating that they are easier to predict.

A similar phenomenon can be observed in Figure 3, which represents the Specific and Specific IOB tagsets. Classes like *city*, *fam*, and *country*, have high frequency and high accuracy. Many of the infrequent classes practically never get correctly predicted, and classes with intermediate frequency, like *first_name_male* or *other_nr_seq* have mediocre accuracy. Once again there are also frequent classes with low accuracy (*sensitive*) and less frequent

classes with high accuracy (e.g. `age_digits`, `month_words`) — which, once again, suggests that some classes can be easy or difficult to predict regardless of their frequency.

These results suggest that while having many examples helps the models to learn to predict a given class, some classes are much easier or much more difficult to predict than others. The performance of some classes is high because of their high frequency in the dataset, whereas some other classes are easy to predict despite not being all that frequent. It therefore appears that it is not only class frequency, but also class semantics that influence the accuracy of predictions, with frequent classes and classes with little internal variation in meaning performing better.

This might also explain why [Sierro et al. \(2024\)](#) observed an improvement with a larger number of classes, as their increase in the number of classes happened once they split (and subsequently narrowed down the semantics of) vague classes and disregarded the `MISC` class, their equivalent of our `sensitive` or `other`. This confirms that identifying semantically distinct classes for annotation is crucial for the success of the annotation scheme and its application in classification tasks. Such labeling requires a good understanding and knowledge of the domain.

While the results show what kind of annotation facilitates the best *detection* or *labeling*, the results of the experiments do not allow us to identify the overall best type of PII annotation, as this depends on the subsequent steps. For example, if the final corpus should contain more specific labels for anonymized spans, then it may be worth to split the process into detection followed by labeling, as detection outperforms labeling at this level of tagset detail; there are some results from other tasks which may suggest that such a separation could be beneficial, e.g. [Park and Fung \(2017\)](#). Another related observation is that PII entities tend not to appear directly adjacent to other PII elements belonging to the same class, which suggests that such boundaries (i.e. IOB-type annotation) need not be included, but it may vary for different labels and domains.

5 Conclusions

We have compared the performance of KB-BERT-based classifiers on detecting and classifying Personally Identifiable Information distinguished by a

different number of classes and the semantic depth or specificity of these classes. We have found that for PII detection, Basic, non-IOB annotation yields the best results. When it comes to labeling, more specific classes do not ensure better results, possibly due to some of those classes being under-represented, since frequency does appear to play some role in how well various classes are detected. An IOB-style annotation also results in a decrease in performance versus not differentiating between beginnings and insides of spans.

We have also found that models fine-tuned on more basic annotation tend to misclassify words that are misspelled, misplaced, or syntactically incorrect more often than models fine-tuned with more specific classes. We have also observed that it is not only class imbalance and a low frequency of a number of the classes, but also the classes' semantics that influence the accuracy of the predictions. Semantically less coherent or less constrained classes make it much more difficult for the models to make correct predictions, pointing to the need for well-defined classes. This emphasizes the role of understanding the domain for which the annotation scheme is designed and raises an important issue concerning the cross-domain transfer of annotation schemes as different classes will have different frequencies and semantics across these classes.

While the choice of PII taxonomy is likely to depend on the needs of the specific case, the results suggest that using over-detailed classes for automatic PII detection and labeling may not lead to optimal performance, at least not without a large dataset for the model to learn from. The same applies to the differentiating between the beginning and the inside of a PII span in IOB-type annotation, which does not lead to better performance, and therefore should only be included if required in the specific case.

In these experiments we have shown what kinds of annotation facilitate PII detection and labeling, the final choice also depends on the subsequent task, such as generating pseudonyms or removing PII spans. As long as the classes are required by the subsequent steps in a pipeline (e.g. pseudonym generation) or desired in the final version of the text (e.g. as placeholders in the anonymized text), there is a need for a more detailed annotation than the basic one utilized in our experiment. This also signals a need for investigating whether the label-

ing step can be separated from the detection step, and how the performance of such a setup compares to classifying the PII in a single step.

The overt class imbalance (including the lack of any PII of certain kinds of Specific labeling in the data) highlights the need for well-curated training datasets that feature a sufficient number of PII of each kind, either by collecting more data or adjusting the annotation; alternatively, one could also opt to combine machine learning and rule-based detection methods (many of the absent Specific classes, such as `account_nr`, could be more easily identified using e.g. regular expressions).

6 Future Work

To strengthen our results, these methods should be applied to larger amounts of training data, potentially resolving issues pertaining to some of the classes being very difficult for our models to learn to predict due to their low frequency. Since we also observe that the semantic vagueness of certain classes is problematic for the models, it would be interesting to split those classes into more coherent subclasses and examine what effect that has – however, this requires manual re-annotation of those tokens. Equally, we would like to see how these results compare for different domains where labels have different distributions in the text or are entirely different.

The question related to the variability of data (here in terms of non-standard spelling in the form of grammatical errors but also other variability such as unconstrained communication) and its interaction with the selected annotation scheme is also open for further exploration. An alternative route would be trying to utilize synthetic data, and, especially, comparing the performance of models trained on larger amounts of synthetic data with models that were only trained on a smaller corpus of authentic data. An intermediate step would be augmenting the training data using e.g. manually or automatically pseudonymized versions of the same texts.

It can also be worth exploring whether the same trends occur when using other BERT-type models for this task — although KB-BERT has been shown to perform the best on PII detection in Swedish texts, perhaps other models do not show the same trends as it does in these experiments.

We also aim to construct PII detection and PII labeling models which we plan to release without

any privacy risks. Comparing an approach where we separate detection and labeling versus where they are combined in a single step is also an interesting path. Since more granular tagsets seem to be used for pseudonym generation in many cases, we consider it worth exploring alternative methods for pseudonym generation that are not as dependent on the PII taxonomy used, e.g. using language models.

Acknowledgments

This work was possible thanks to the funding of several grants from the Swedish Research Council.

All of the authors are supported by the research environment project *Grandma Karl is 27 years old: Automatic pseudonymization of research data* with the funding number 2022-02311 for the years 2023-2029.

The first, third, and fourth authors are also receiving support from the Swedish national research infrastructure *Språkbanken*, which is jointly funded by its 10 partner institutions and the Swedish Research Council (2018–2024; dnr 2017-00626).

The second author is also supported by a grant from the Swedish Research Council (VR project 2014-39) for the establishment of the *Centre for Linguistic Theory and Studies in Probability (CLASP)* at the University of Gothenburg.

This work has also been aided by the Swedish national research infrastructure *Huminfra*, funded for the years 2022-2024, contract 2021-00176, and the participating partner institutions.

Limitations

One major limitation in our experiments is the relatively small amount of training data. However, the particular hierarchical PII taxonomy that we analyze is only used in the SWELL corpora, and SWELL-GOLD’s correction annotation sets it apart from other corpora with hierarchical annotation, such as CODE ALLTAG (Eder et al., 2020). Unfortunately, SWELL-PILOT is not correction-annotated, meaning that we can only conduct certain result analyses on a subset of our data.

Despite the small amount of data, a qualitative analysis of the errors made by the models was deemed to be beyond the scope, as it would require a manual inspection of almost 1000 texts in six different annotation versions.

Since it takes a considerable amount of time to train a BERT-based classifier, we trained on 6 different kinds of annotation, we limited ourselves to 10 runs per annotation type, which does not satisfy the requirements of applying statistical tests on the overall performance results.

Ethical Considerations

Since the data that we use to fine-tune our models includes Personally Identifiable Information, it cannot be openly shared. We choose not to share our models to avoid any risks of leakage of personal information. However, we provide the code (see subsection 3.2) from which the results can be generated provided one has access to the data in the appropriate SWELL format.

References

- Alyaa Alfalahi, Sara Brissman, and Hercules Dalianis. 2012. [Pseudonymisation of Personal Names and other PHIs in an Annotated Clinical Swedish Corpus](#). In *Proceedings of the Third Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTextM 2012) held in conjunction with LREC 2012*.
- Hanna Berg and Hercules Dalianis. 2020. [A semi-supervised approach for de-identification of Swedish clinical text](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4444–4450, Marseille, France. European Language Resources Association.
- 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.
- Elisabeth Eder, Ulrike Krieg-Holz, and Udo Hahn. 2020. [CodE alltag 2.0 — a pseudonymized German-language email corpus](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4466–4477, Marseille, France. European Language Resources Association.
- Elisabeth Eder, Michael Wiegand, Ulrike Krieg-Holz, and Udo Hahn. 2022. [“beste grüße, maria meyer” — pseudonymization of privacy-sensitive information in emails](#). In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 741–752, Marseille, France. European Language Resources Association.
- Mila Grancharova and Hercules Dalianis. 2021. [Applying and sharing pre-trained BERT-models for named entity recognition and classification in Swedish electronic patient records](#). In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 231–239, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- Pierre Lison, Ildikó Pilán, David Sanchez, Montserrat Batet, and Lilja Øvrelid. 2021. [Anonymisation models for text data: State of the art, challenges and future directions](#). 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 4188–4203, Online. Association for Computational Linguistics.
- Martin Malmsten, Love Börjeson, and Chris Haffenden. 2020. [Playing with Words at the National Library of Sweden – Making a Swedish BERT](#).
- Beáta Megyesi, Lena Granstedt, Sofia Johansson, Julia Prentice, Dan Rosén, Carl-Johan Schenström, Gunlög Sundberg, Mats Wirén, and Elena Volodina. 2018. [Learner corpus anonymization in the age of GDPR: Insights from the creation of a learner corpus of Swedish](#). In *Proceedings of the 7th workshop on NLP for Computer Assisted Language Learning*, pages 47–56, Stockholm, Sweden. LiU Electronic Press.
- George A. Miller. 1995. WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11):39–41.
- Official Journal of the European Union. 2016. [Consolidated text: Regulation \(EU\) 2016/679 of the European parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/EC \(general data protection regulation\) \(text with EEA relevance\)](#). *Official Journal*, (Document 02016R0679-20160504).
- Anthi Papadopoulou, Pierre Lison, Mark Anderson, Lilja Øvrelid, and Ildikó Pilán. 2023. [Neural text sanitization with privacy risk indicators: An empirical analysis](#).
- Anthi Papadopoulou, Yunhao Yu, Pierre Lison, and Lilja Øvrelid. 2022. [Neural text sanitization with explicit measures of privacy risk](#). 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 217–229, Online only. Association for Computational Linguistics.
- Ji Ho Park and Pascale Fung. 2017. [One-step and two-step classification for abusive language detection on Twitter](#). In *Proceedings of the First Workshop on Abusive Language Online*, pages 41–45,

- Vancouver, BC, Canada. Association for Computational Linguistics.
- Ildikó Pilán, Pierre Lison, Lilja Øvrelid, Anthi Papadopoulou, David Sánchez, and Montserrat Batet. 2022. [The text anonymization benchmark \(TAB\): A dedicated corpus and evaluation framework for text anonymization](#). *Computational Linguistics*, 48(4):1053–1101.
- Maria Sierro, Begoña Altuna, and Itziar Gonzalez-Dios. 2024. [Automatic detection and labelling of personal data in case reports from the ECHR in Spanish: Evaluation of two different annotation approaches](#). In *Proceedings of the Workshop on Computational Approaches to Language Data Pseudonymization (CALD-pseudo 2024)*, pages 18–24, St. Julian’s, Malta. Association for Computational Linguistics.
- Språkbanken Text. 2024a. [SweLL-gold](#).
- Språkbanken Text. 2024b. [SweLL-pilot](#).
- Maria Irena Szawerna, Simon Dobnik, Therese Lindström Tiedemann, Ricardo Muñoz Sánchez, Xuan-Son Vu, and Elena Volodina. 2024a. [Pseudonymization categories across domain boundaries](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 13303–13314, Torino, Italia. ELRA and ICCL.
- Maria Irena Szawerna, Simon Dobnik, Ricardo Muñoz Sánchez, Therese Lindström Tiedemann, and Elena Volodina. 2024b. [Detecting personal identifiable information in Swedish learner essays](#). In *Proceedings of the Workshop on Computational Approaches to Language Data Pseudonymization (CALD-pseudo 2024)*, pages 54–63, St. Julian’s, Malta. Association for Computational Linguistics.
- Elena Volodina, Lena Granstedt, Arild Matsson, Beáta Megyesi, Ildikó Pilán, Julia Prentice, Dan Rosén, Lisa Rudebeck, Carl-Johan Schenström, Gunlög Sundberg, and Mats Wirén. 2019. [The SweLL Language Learner Corpus: From Design to Annotation](#). *Northern European Journal of Language Technology*, 6:67–104.
- Elena Volodina, Ildikó Pilán, Ingegerd Enström, Lorena Llozhi, Peter Lundkvist, Gunlög Sundberg, and Monica Sandell. 2016. SweLL on the rise: Swedish Learner Language corpus for European Reference Level studies. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), May 23-28, 2016, Portorož, Slovenia*, Paris. European Language Resources Association.
- 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](#).

A Appendix

Example (1), shows what all the annotation schemes used in this paper look like on sample text. The annotation schemes a-f correspond to the Specific IOB, Specific, General IOB, General, Basic IOB, and Basic annotations, respectively.

- (1) a. My name is **Maria** . I
 O O O B-firstname_female O O
 come from **Wroclaw** (that is in
 O O B-city O O O O
Poland) . I work at the
 B-country O O O O O O
University of Gothenburg .
 B-work I-work I-work O
- b. My name is **Maria** . I come
 O O O firstname_female O O O
 from **Wroclaw** (that is in **Poland**) .
 O city O O O O country O O
 I work at the **University of**
 O O O O work work
Gothenburg .
 work O
- c. My name is **Maria** . I come
 O O O B-personal_name O O O
 from **Wroclaw** (that is in
 O B-geographic O O O O
Poland) . I work at the
 B-geographic O O O O O O
University of Gothenburg
 B-institution I-institution I-institution
 .
 O
- d. My name is **Maria** . I come
 O O O personal_name O O O
 from **Wroclaw** (that is in **Poland**
 O geographic O O O O geographic
) . I work at the **University**
 O O O O O O institution
of Gothenburg .
 institution institution O
- e. My name is **Maria** . I come from
 O O O B O O O O
Wroclaw (that is in **Poland**) . I
 B O O O O B O O O
 work at the **University of Gothenburg**
 O O O B I I
 .
 O
- f. My name is **Maria** . I come from
 O O O S O O O O
Wroclaw (that is in **Poland**) . I
 S O O O O S O O O
 work at the **University of Gothenburg**
 O O O S S S
 .
 O

Braxen 1.0

Christina Tånnander

Swedish Agency for Accessible Media
KTH Royal Institute of Technology
christina.tannander@mtm.se

Jens Edlund

KTH Royal Institute of Technology
edlund@speech.kth.se

Abstract

With this paper, we release a Swedish pronunciation lexicon resource, Braxen 1.0, which is the result of almost 20 years development carried out at the Swedish Agency for Accessible Media (MTM). The lexicon originated with a basic word list, but has continuously been expanded with new entries, mainly acquired from university textbooks and news text. Braxen consists of around 850 000 entries, of which around 150 000 are proper names. The lexicon is released under the CC BY 4.0 license and is accessible for public use.

1 Introduction

The mission of the Swedish Agency for Accessible Media (MTM) includes the production of accessible materials for individuals with print impairments, such as low vision or dyslexia. This work primarily involves converting books into accessible formats such as Braille and talking books. The talking books are produced through either human narration or text-to-speech synthesis (TTS). MTM uses TTS to produce approximately 1 500 university textbooks annually and more than 120 newspapers on a near-daily basis. While commercial TTS voices are used in this production, the complexity of non-fiction texts often necessitates additional support to ensure accuracy, mainly through pronunciation instructions. The pronunciation dictionary used at MTM, from which Braxen is derived, is referred to as MTM-lex for clarity.

The starting point of MTM-lex was CentLex, a generalised Swedish lexicon for speech technology developed at the academic-industrial centre of excellence CTT in the early 2000s (Jande, 2006). In 2005, as CTT approached the end of its 10-year run, MTM made the decision to develop an

in-house TTS system for the production of talking books (Tånnander, 2018). The pronunciation lexicon in the MTM TTS started with 55 000 entries from CentLex, and was supplemented with around 35 000 entries acquired from Svenska språknämndens uttalsordbok (SUO), *67 000 ord i svenskan och deras uttal* (Garlén, 2003). SUO was made publicly available by the Institute for Language and Folklore under the CC-0 license in 2023 (Isof, 2023). MTM has since made significant changes and expansions to the lexicon to meet the substantial demands placed on a pronunciation lexicon used for TTS synthesis of long and information-rich text, such as university textbooks. An in-house format for a phone alphabet used for phonetic transcriptions was developed and inflections of baseforms were added along with hundreds of thousands of new entries, mainly proper names and domain-specific vocabulary.

As part of an active production process, MTM-lex is continuously updated, primarily with words from Swedish newspapers and university textbooks. MTM produces over 120 newspapers in spoken form, read aloud by TTS. The lexicon is updated weekly with the 100+ most frequent news words that are not yet part of MTM-lex. These pronunciations are then forwarded to the TTS system, either as a user lexicon or as SSML insertions in the newspaper document. In addition, MTM produces around 1 500 Swedish and English university textbooks with speech synthesis annually. Frequency lists are computed individually for most books, and new high-frequency words are added to MTM-lex. In this way, the lexicon is kept up-to-date with the current vocabulary of the news world, as well as with vocabularies from specific domains, such as medicine or law.

The sharing of Braxen has been approved by MTM's legal team. The lexicon can be downloaded here: <http://www.github.com/sprakbankental/braxen>.

2 Initial release: Braxen 1.0

Braxen is not identical to the original MTM-lex and does not include all of its entries or information. Firstly, only 5 of the original 27 fields are included in the first release (see section 4). The remaining fields are excluded for one of the following reasons: they are internal to MTM, unavailable for most of the lexicon entries, lacking in quality or consistency, or simply mere placeholders for future information.

Secondly, not all entries are included in the release. English proper names are included, but approximately 35 000 general English words are not. These words were originally transcribed to match the English variety of a specific Swedish speaker, which was incorporated into the Swedish TTS system mentioned in section 1. As a result, the current pronunciations differ from more established transcription conventions of English.

As with any lexicon, it is virtually impossible to guarantee complete accuracy.

For example, all entries do not have complete PoS information, partially due to the purpose of the resource. Features that are less important in speech-oriented dictionaries, such as whether a word is an adjective or a perfect participle, have been given less attention. We are also aware that there are a small number of incorrect entries.

To the best of our knowledge, this remains the best Swedish resource of its kind available by some margin.

The release includes full documentation and Perl scripts for conversion between the native transcription format and IPA, as well as validation scripts.

3 Statistics

This section presents statistics on a selection of features of general words (852 000 entries, Table 1) and proper names (151 000 entries, Table 2).

Examining the baseforms of the open part-of-speech classes of general words, we count approximately 129 000 nouns, 8 000 verbs, and 16 000 adjectives, present and perfect participles.

4 Fields

This section describes the five fields included in the initial release.

| Words | Number | Example |
|-------------|----------|-------------|
| Baseforms | 318 000 | lexikon |
| Inflections | 534 000 | lexikonen |
| Swedish | 679 000 | lexikon |
| English | (35 000) | lexicon |
| Latin | 3 500 | humanitatis |
| Norwegian | 2 600 | langrenn |
| German | 2 000 | Krankheit |
| French | 1 700 | ouvrière |
| Other | 7 800 | áhkkku |

Table 1: Word statistics. Note that the English entries are not part of this initial release.

| Proper names | Number | Example |
|--------------|---------|------------|
| Baseforms | 151 000 | Stockholm |
| Inflections | 23 000 | Stockholms |
| Swedish | 91 000 | Göteborg |
| English | 16 000 | Gothenburg |
| Other | 44 000 | København |

Table 2: Proper name statistics.

4.1 Orthography

The orthography is displayed in the letter casing that reflects the most common form of the word.

4.2 Part-of-speech (PoS)

The PoS field contains part-of-speech tags and morphological information, following the SUC standard (Ejerhed and Ridings, 2010).

4.3 Language

The language field generally follows the ISO 639-2 standard (Library of Congress, 2017) and indicates which language the pronunciation refers to. Consequently, the same orthography can occur multiple times and have different pronunciations depending on language. Detailed identification of language properties is not a primary task in this work, but words and proper names are classified as belonging either to a specific language or to a pragmatic placeholder category indicating for example a continent associated with the word, such as 'afr' (Africa) or 'asi' (Asia). These placeholder categories are not linguistically accurate, but pave the way for a more refined classification in a forthcoming edition by providing accessible classes for untrained annotators.

4.4 Pronunciation

The pronunciation field contains the standard pronunciation of the orthography. Only one pronunciation variant is present in this first release of Braxen. The details of the symbol set, stress and boundary information are explained in section 5.

4.5 ID

Finally, the ID field contains a unique identification number.

5 Phonetic-phonemic transcription

This section describes the Braxen transcriptions and the symbol set used to encode them. As Braxen is primarily a pronunciation resource for real-world Swedish speech technology, and particularly for TTS synthesis of long, information-rich texts, much care has been taken to create transcriptions that are *useful* for this purpose. This goal takes precedence over strict adherence to any specific speech or language theory, and even over generality in terms of language independence.

The Braxen transcriptions are encoded using a symbol set based on four main design principles, some of which are language-specific. The symbol set can be converted to its IPA equivalent using tools included with the release.

Principle 1: Programming compatibility

Symbols that complicate programming should be excluded.

Principle 1 primarily rejects characters which often serve as control characters in programming languages, such as the SAMPA symbols `/ { /` and `/ @ /`. It also excludes IPA stress and accent notations such as `/ ˈ /` and `/ ˌ /`, which can complicate the splitting and parsing of pronunciations. The principle also underpins the decision to separate all phonemes by space, as this facilitates splitting words and longer entities into phonemes and makes pronunciation easier to read.

Principle 2: Keyboard accessibility

All symbols should be easily accessible on a Swedish keyboard without compromising ergonomics.

This principle bluntly excludes most IPA symbols and prohibits keyboard combinations (e.g. combinations involving Shift, Alt, or Ctrl). Consequently, it limits the symbols to lowercase characters but allows the inclusion of Swedish alphabetic characters such as “ä” and “ö”.

Principle 3: Visual transparency

Each symbol should preferably resemble its typical orthographic counterpart or its IPA equivalent.

Principle 3 has various implications, such as using the colon “:” as the vowel length marker and “u” for the closed rounded back vowel.

Principle 4: Internal coherence

Each symbol representation should aim for internal coherence, both within the symbol set and within individual symbols.

This principle is especially important for multi-character symbols, where it favours systematic compositionality and mnemonically sound choices.

5.1 Phones

The symbol set consists of 65 phones, 15 of which are xenophones. These are used for speech sounds that are not inherently Swedish, for example `/ ð - dh /` or `/ õ - on /`. In this section, we describe the rationale behind the notation but refer the reader to the documentation for a complete list of phones and their IPA counterparts.

Following Principle 4, we aim for a consistent use of multi-character symbols when a single-symbol notation is not feasible using the keyboard alone. The additional characters used are presented in Table 3 and include the following:

- The colon marks long vowels: `/ i: /`.
- “h” is attached to speech sounds to signal some kind of modification of the single symbol, e.g. `/ ʃ - sh /` and `/ ð - dh /`. This means that we end up with three-character notations of some English diphthongs, e.g. `/ ɛə - eeh /`.
- Nasal vowels are followed by “n”: `/ an, on /`.
- Retroflex speech sounds are preceded by “r”: `/ rd, rt, rn, rs, rl /`.

| Symbol | Meaning | Example |
|-----------------|-----------|---------------------|
| <code>_:</code> | long | <code>i:</code> |
| <code>_h</code> | modified | <code>dh, oh</code> |
| <code>_n</code> | nasalised | <code>an</code> |
| <code>r_</code> | retroflex | <code>rt</code> |
| <code>_x</code> | more back | <code>rx</code> |
| <code>--</code> | modified | <code>uu:</code> |
| <code>_0</code> | silent | <code>r0</code> |

Table 3: Meaning of control characters placed before or after the main part of the phone.

- "x" involves a more back pronunciation of the original speech sound, e.g. /R- rX/.
- Similar to attached "h", a double notation of a symbol signals a similar, but different phoneme, e.g. /u: - u:/ and /uu: - u:/.
- "Silent speech sounds", such as R.P. English /r/ are followed by "0": /r0/.

5.2 Stress

We use three stress and accent symbols (see Table 4): the primary stress with its two accent variations, and secondary stress. Note that all accent 2 words in Braxen are assigned secondary stress. In most Swedish phonetic transcriptions, the secondary stress is assigned compounds only, and left out in simplex words such as /h "o . p a/. Here, we acknowledge that we have violated both Principle 1 (programming compatibility: the single and double quotes) and Principle 2 (keyboard accessibility: e.g., the Shift key is used for typing accent 2). However, these symbols are justified by their clear connection to stress symbols of other symbol sets: /' (primary stress, accent 1) and /, (secondary stress) are visually similar to the IPA symbols, and /'' (accent 2) visually resembles two primary stress symbols combined.

5.3 Boundaries

Three types of boundaries are used: word, compound and syllable boundaries, as shown in Table 4. Again, Principle 2 is violated, this time by the word boundary "|". We find some reassurance in the fact that this symbol is rarely needed in a pronunciation dictionary, although it is more frequently used in input for applications such as speech synthesis. In these cases, word boundaries are typically inserted automatically.

| Symbol | Meaning | Orthography | Pronunciation |
|--------|------------------|-------------|-----------------------------------|
| ' | accent 1 | boll | b 'o l |
| " | accent 2 | fotboll | f "u: t - b ,o l |
| , | secondary stress | fotboll | f "u: t - b ,o l |
| | unstressed | bollen | b 'o . l ex n |
| | word | 7-eleven | s 'e . v ex n e . l 'e . v ex n |
| - | compound | fotboll | f "u: t - b ,o l |
| . | syllable | bollen | b 'o . l ex n |

Table 4: Stress and boundaries.

6 Conclusions and future work

This release of Braxen 1.0 marks a step forward for Swedish speech technology in that it provides an accessible and high-quality pronunciation lexicon for Swedish speech technology applications.

With its comprehensive symbol set tailored to Swedish language needs and adherence to practical design principles, Braxen is well-suited for TTS synthesis and other real-world applications. While the current release offers robust functionality, several areas remain open for enhancement and expansion, and a range of activities are already on the list for upcoming releases:

- Consolidate the excluded 35 000 English MTM-lex entries, and/or add entries from an existing English pronunciation dictionary.
- Implement validation that conforms that pronunciations are plausible given their associated orthography (a complement to existing validation).
- Correct and include other MTM-lex fields that might be of interest to others, such as compound decomposition, pronunciation variations, and word origin.
- Establishing procedures for regular updates to the dictionary, in particular automated transfer of valid additions from MTM-lex to Braxen.
- Release the full symbol set specification.
- Release a free-standing conversion tool between the symbol set used in Braxen and other widespread symbol sets (e.g. SAMPA) in addition to the existing IPA conversion.

Acknowledgments

MTM-lex was developed by the Swedish Agency for Accessible Media, MTM. Its refactoring and partial release as Braxen has partly taken place in collaboration in the Vinnova project Deep learning based speech synthesis for reading aloud of lengthy and information rich texts in Swedish (2018-02427). The resource will be maintained and accessible through the Swedish Research Council funded national infrastructure Språkbanken Tal (2017-00626).

References

- Library of Congress. 2017. Iso 639-2. Accessed: 2024-08-22.
- Eva Ejerhed and Daniel Ridings. 2010. Suc - parole.
- Claes Garlén. 2003. *Svenska språknämndens uttalsordbok, 67 000 ord i svenskan och deras uttal*. Norstedts Förlag AB.
- Isof. 2023. Svenska språknämndens uttalsordbok.
- Per-Anders Jande. 2006. *Modelling phone-Level pronunciation in discourse context*. KTH Computer Science and Communication.
- Christina Tännander. 2018. Speech synthesis and evaluation at mtm. In *Proc. of Fonetik 2018*.

Temporal Relation Classification: An XAI Perspective

Sofia Elena Terenziani
IT University of Copenhagen
seterenziani@gmail.com

Abstract

Temporal annotations are used to identify and mark up temporal information, offering definition into how it is expressed through linguistic properties in text. This study investigates various discriminative pre-trained language models of differing sizes on a temporal relation classification task. We define valid reasoning strategies based on the linguistic principles that guide commonly used temporal annotations. Using a combination of saliency-based and counterfactual explanations, we examine if the models' decisions are in line with the strategies. Our findings suggest that the selected models do not rely on the expected linguistic cues for processing temporal information effectively¹.

1 Introduction

Temporal information processing is a fundamental aspect of natural language and is essential for NLP applications including question answering (Chen et al., 2021; Ko et al., 2023), text summarization (Daiya, 2020), and information retrieval (Gade and Jetcheva, 2024). Transformer-based pre-trained language models have shown impressive performance in such tasks (Xiong et al., 2024; Ko et al., 2023; Tai, 2024; Shi et al., 2023). Yet, their interpretation of time diverges from human interpretation (Callender, 2011), making it challenging to evaluate their temporal processing, and whether they indeed interpret the temporal information as expected (Qiu et al., 2023; Jain et al., 2023).

While temporal benchmarks (Tan et al., 2023a; Zhou et al., 2019; Ning et al., 2020; Zhou et al., 2021) have been extensively developed, performance metrics alone do not reveal the under-

lying mechanisms or explain how conclusions are reached (Chakraborty et al., 2017). This study contributes a methodology and an evaluation dataset for evaluating NLP models on temporal relation classification. We define valid reasoning strategies, and use a combination of saliency-based and example-based explainability methods to assess whether a model follows these strategies when making decisions.

Our framework extends the work introduced by Ray Choudhury et al. (2022). We explore discriminative models of varying sizes to determine if larger models, trained more extensively on more data, are also more likely to base their decisions on valid information retrieval processes. Our findings suggest that while larger models show better performance on the task, they frequently deviate from expected reasoning strategies. These results align with broader concerns about the reliability of current popular benchmarks, where high accuracy can mask a reliance on shortcuts or spurious correlations. We discuss the limitations of this framework, together with the opportunities and challenges of extending it to generative models.

2 Related Work

Temporal Relation Classification. Temporal relation classification (TRC) was first introduced in TempEval-3 (UzZaman et al., 2013) and gained popularity with dedicated corpora and annotations for temporal information processing. Modern TRC methods predominantly use discriminative pre-trained language models, to generate robust contextual representations for pairs of event mentions (Yang et al., 2019; Lin et al., 2019). Further advancements include graph-based methods (Mathur et al., 2021; Zhang et al., 2022; Zhou et al., 2022) and prompt and masking techniques (Han et al., 2021; Yang et al., 2024). Despite the recent surge in the generative models, they still underperform compared to fine-tuned smaller mod-

¹<https://github.com/sofitere/TRC-XAI>

| | Reasoning Step | Relevant Features |
|--|-----------------------------------|---|
| Context:
Leon <u>won</u> the marathon years after he <u>underwent</u> surgery in 2011. | Identify temporal information | Expression: <i>years, 2011</i>
Preposition: <i>after</i> |
| | Map temporal information to event | <i>underwent := 2011</i>
<i>won := (years, after)</i> |
| Relation: $\langle \text{won}, ?, \text{underwent} \rangle$ | Determine temporal relationship | <i>won := year after 2011</i>
$\langle \text{won}, \text{AFTER}, \text{underwent} \rangle$ |

Table 1: Valid reasoning steps for determining the temporal relation between a given event pair.

els (Roccabruna et al., 2024; Yuan et al., 2023).

Temporal Annotation. TimeML (Mani et al., 2006) remains the most widespread format for temporal annotation, and it became the basis for ISO standard (Pustejovsky et al., 2010). TimeML includes conventions to identify and describe temporal elements in text, including temporal expressions (TIMEX), events, temporal relations (T-LINKS), signals (SIGNAL), and relation types. TimeBank corpus (Pustejovsky et al., 2003) has been re-annotated in several projects to increase the density of T-LINKs (Verhagen et al., 2007; Rogers et al., 2022; Naik et al., 2019) and improve its consistency. Its texts have been utilized in subsequent projects providing additional annotation in other formats, including MATRES (Ning et al., 2018).

Benchmarks. Benchmarks for temporal processing vary widely in format and scope. TimeQA (Chen et al., 2021) and Tempreason (Tan et al., 2023b) focus on temporal question answering, Torque (Ning et al., 2020) on temporal reading comprehension, adopting question/answering as format, and MCTACO (Zhou et al., 2019) on temporal commonsense reasoning, adopting multiple-choice as format. Commonly used benchmarks have shown some limitations, also here ranging from task and scope. Temporal question-answering (QA) benchmarks tend to be biased in their coverage of time spans and question types, leading to models performing well due to format biases rather than actual language processing skills (Tan et al., 2023c). Additionally, benchmarks with focus on temporal expressions, such as numeric years, have shown to not represent the full range of diversity of temporal expressions (Qin et al., 2021). Benchmarks for reading comprehension often assume that performing well necessitates engaging with cognitive processes of language understanding (Sugawara et al., 2019; Weston et al., 2015), implying that higher scores

reflect advances in general language processing (Ray Choudhury et al., 2022). Performance on benchmarks alone, while useful, does not necessarily tell us whether the model is right for the right reasons; if it is not, the benchmark results may be misleading and not generalize to other data (Dehghani et al., 2021; Bowman and Dahl, 2021).

Explainability. Explainability methods can account for some of the limitations of the current benchmarks by highlighting what information the model relies on, or where it fails to perform. They can thus provide means to check to what degree the models are reliable, i.e. they perform correctly and consistently for the right reasons (McCoy et al., 2019; Christianson, 2016). For this line of research, local and post-hoc methods have been used to evaluate pre-trained language models on tasks that demand specific linguistic skills. Ray Choudhury et al. (2022) apply a combination of these methods to analyze and evaluate models on two linguistic skills required for a reliable reading comprehension system, finding that models use shortcuts rather than valid inference strategies. In the context of LLMs, explainability methods are both important and challenging. Research efforts are also put into examining the utility (González et al., 2021), interpretability (González et al., 2021; Schuff et al., 2022) and reliability (Harbecke and Alt, 2020; Spreitzer et al., 2022; Rahimi and Jain, 2022) of explainability methods.

Contribution. To date, relevant NLP work on temporal processing has focused on modeling, benchmarks and annotation schemes. The use of explainability methods to explore how models handle temporal data is largely unexplored. To the best of our knowledge, this is the first study to apply saliency-based and example-based interpretability methods to assess whether models rely on the expected reasoning patterns for temporal relation classification. We evaluate the validity

| | |
|--|--|
| <p>Context: Leon <u>won</u> a marathon years after he <u>underwent</u> surgery.</p> <p>Relation: ⟨won, AFTER, underwent⟩</p> | <p>Context: Leon won a marathon years after he underwent surgery.</p> <p>Relation: ⟨won, AFTER, underwent⟩</p> |
|--|--|

(a) Example of temporal annotation: TIMEX (blue), and SIGNAL (orange). These are temporal elements relevant for expressing the relationship between 'won' and 'underwent'.

(b) Example of token partitioning into positive (green) and negative (red). The models are expected to rely more on the tokens in the 'positive' set.

Figure 1: A sample question from the MATRES (Ning et al., 2018) dataset. A model is asked to predict the temporal relationship between winning a marathon and having brain surgery. Token partitioning is delivered from the features defined as relevant for determining the temporal relation between two events.

of these methods (via examining their alignment), and discuss the challenges of evaluating the latest generative models on temporal relation classification.

3 Defining Success Criteria for TRC

Evaluating whether models follow expected reasoning involves testing if their decision-making process are based on valid information retrieval and inference strategies rather than superficial patterns in the data. Ray Choudhury et al. (2022) defines three success criteria for NLP systems: a system must (1) accurately perform on a specific task, (2) rely on information deemed pertinent to the task, and (3) maintain consistency under distribution shifts. We evaluate a model’s performance in TRC against these criteria. We first define the expected reasoning processes (§ 3.1). We then assess the model’s adherence to these reasoning steps by verifying its reliance on valid information (§ 4.7), and by evaluating its performance consistency across variations in data distribution (§ 4.6).

3.1 What reasoning should a model perform?

To correctly extract and classify temporal relations, a model must identify linguistic features that express temporal information, map these features to the events they describe or modify, and use this information to deduce the temporal relationship between the pair of events. We define these as valid reasoning steps² (see Table 1 for an exam-

²We recognize that this represents only the minimal information on which models (or humans) might rely. For the example shown in Table 1, if the context includes details about Leon breaking his leg, this information could reasonably influence the understanding of Leon’s chances of winning the marathon. Nonetheless, the minimally necessary information in the immediate context would still be salient, and it is a reasonable expectation that either models or humans should rely on it.

ple). Temporal annotation schemes and guidelines can be used to clarify which linguistic features are essential for identifying the temporal relation between an event pair. We focus on two types of annotations from the TimeML guidelines (Mani et al., 2006):

- **TIMEX3** tags are utilized for annotating explicit temporal expressions within text. These expressions can be absolute (“December 2025”, “5PM”) or relative (“Mondays”, “monthly”). They serve to anchor events to specific times or durations.
- **SIGNAL** tags mark words or phrases that cue the relationships between two entities (e.g. *timex to event*, *timex to timex*, *event to event*). Common linguistic features are adverbs (“again”, “late”, “eventually”) detailing the timing of events, conjunctions (“before”, “since”, “while”) relating events to each other and subordinate conjunctions (“because”, “if”, “therefore”) expressing conditional or causal relationships. These features indicate the sequence or structure of events, showing their interactions over time.

Essentially, while **TIMEX3** tags are used to identify temporal entities, **SIGNAL** annotations establish the links between these entities within the text. Together, they provide the foundational information necessary to understand the temporal relationships among events in texts.

3.2 What reasoning does a model perform?

Having established the reasoning processes a model should follow, the next step is to assess whether a specific model adheres to these. Ray Choudhury et al. (2022) uses a combination of example-based and saliency-based interpretability methods. These methods are categorized as local and post-hoc (Molnar, 2022): they focus on individual instances and they are applied after model

| | Purpose | # Docs | #Events | #TLinks |
|--------------|------------|--------|---------|---------|
| TimeBank | Training | 162 | 6.6k | 6.5k |
| Aquaint | Training | 73 | 4,3k | 6.4k |
| Platinum | Validation | 20 | 748 | 837 |
| <i>Total</i> | | 275 | 6k | 13.5k |

Table 2: Summary of purpose and statistics of the MATRES (Ning et al., 2018) dataset subsets.

has been trained.

Saliency-based Methods. Saliency-based methods are a family of methods that offer feature-centered explanations (Molnar, 2022; Ding and Koehn, 2021a). These methods offer different ways of computing a score for each token, indicating how individual features (token) affect a model’s decision. By comparing the saliency scores to a predefined partition of tokens, these explanations can be used to determine whether a model is relying on the right information for correct predictions. Following Ray Choudhury et al. (2022), we define a partition of the token space as: tokens a model should find important (positive), and tokens a model should not find important (negative) (§ 4.5). If saliency scores show that a model consistently has higher scores on the positive compared to the negative partition of tokens, it suggests that the model focuses on the ‘right’ information.

Counterfactual Explanations. Counterfactual explanations offer data-centred explanations by analyzing how changes in the input data can lead to different model predictions (Molnar, 2022). By changing parts of the input with alternative valid tokens that would change the type of temporal relation, these explanations can help determine if a model is relying on the expected reasoning strategies (§ 4.6). If a model predicts the correct temporal relationship for both original and altered inputs, it suggests that the model consistently relies on the correct information.

Explanation Alignment. For a model to demonstrate valid reasoning, both saliency and counterfactual explanations must align across many instances, suggesting that a model consistently relies on the right information for accurate predictions.

| Label | # | % |
|--------|-------|-----|
| BEFORE | 6.886 | 50% |
| AFTER | 4.576 | 34% |
| VAGUE | 1.644 | 12% |
| EQUAL | 471 | 4% |

Table 3: Label distribution in the MATRES (Ning et al., 2018) dataset.

4 Methodology

4.1 Data

Our experiments are conducted on the MATRES dataset (Ning et al., 2018). In total, MATRES includes 275 news articles from TempEval3 (UzZaman et al., 2013), annotated for temporal relations between pair of events. For experimental consistency, we follow the original split for training and evaluation (Ning et al., 2019), as shown in Table 2. MATRES is annotated for four different temporal relation classes. The label distribution is shown in Table 3.

4.2 Models

We experiment with transformer-based encoders of different sizes from three families: BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), and LUKE (Yamada et al., 2020). BERT and RoBERTa are the classical models to use for this task; they share a similar architecture but differ in pre-training scope and optimization (with RoBERTa also receiving more extensive training, but without optimization for the next-sentence-prediction task). They have been used extensively for temporal relation classification (Liu et al., 2019).

We also add LUKE (Yamada et al., 2020): the model enhancing the RoBERTa framework with entity-aware self-attention, improving contextual understanding. Since entities are crucial to temporal relation classification (e.g. for recognizing dates and events), this model could be expected to improve on base BERT/RoBERTa. For all models, we experiment with ‘base’ and ‘large’ versions. For some cases, larger models have shown to generalise better (Zhong et al., 2021; Desai and Durrett, 2020). Part of this project is set to investigate whether they are also more likely to rely on the right information. We focus on discriminative models, as they are known for their robust performance in TRC (§ 2). While incorporating gener-

| | |
|--|--|
| <p>Original: Leon <u>won</u> a marathon few years after he <u>underwent</u> surgery.</p> <p>Relation: ⟨won, AFTER, underwent⟩</p> <hr/> <p>Altered: Leon <u>won</u> a marathon few years <u>before</u> he <u>underwent</u> surgery.</p> <p>Relation: ⟨won, <u>AFTER</u>, underwent⟩</p> | <p>Original: Computers, about to be <u>deployed</u>, are taking over (..)</p> <p>Relation: ⟨deployed, AFTER, taking⟩</p> <hr/> <p>Altered: Computers, <u>already</u> <u>deployed</u> <u>for</u> <u>months</u>, are <u>now</u> taking (..)</p> <p>Relation: ⟨deployed, <u>BEFORE</u>, taking⟩</p> |
| (a) Simple reversal of temporal conjunctions. | (b) Label reversal with more extensive editing |
| <p>Original: If it <u>performs</u> as (..), the design could be <u>used</u> to (..)</p> <p>Relation: ⟨performs, BEFORE, used⟩</p> <hr/> <p>Altered: If it <u>is</u> <u>used</u> to (..), the design <u>currently</u> <u>performs</u> as (..)</p> <p>Relation: ⟨performs, <u>EQUAL</u>, used⟩</p> | <p>Original: He <u>took</u> part in the mission. He also <u>made</u> expeditions to (..)</p> <p>Relation: ⟨took, VAGUE, made⟩</p> <hr/> <p>Altered: He <u>made</u> expeditions to (..). He <u>later</u> <u>took</u> part in the mission.</p> <p>Relation: ⟨took, <u>AFTER</u>, made⟩</p> |
| (c) Changing a conditional relationship | (d) Sentence reordering |

Figure 2: Examples of counterfactual alterations changing the original temporal relation label, with altered tokens highlighted in yellow.

ative models could be insightful, their limitations within this framework are addressed in Section 7.

4.3 Fine-Tuning

Each encoder is fine-tuned for TRC using the tokenization strategy proposed by Yanko et al. (2023) and Baldini Soares et al. (2019). The strategy consists in explicitly marking the boundaries of each event in an input sentence with special tokens. We define these as [a1], [/a1], [a2], [/a2] and process each input sentence as following:

Leon [a1] won [/a1] a marathon years after he [a2] underwent [/a2] surgery.

When a given input is processed by each encoder, the embeddings of the special tokens are adjusted based on surrounding tokens. This results in a context-specific representation for each event. We concatenate the embedding vectors of the special tokens and use them for classification by feeding them into a linear layer on top of each encoder. All code to reproduce our results, including hyperparameters, is included with the submission and will be made public upon acceptance of the paper.

4.4 Evaluation Metrics

We evaluate each encoder using standard evaluation metrics for classification: F1 and exact-match. Given the significant class imbalance in the MATRES dataset (see Table 3), the F1-score is particularly important. We report both weighted and macro-average F1-score. Although exact-

match is less reliable for imbalanced datasets, we include it for its straightforward interpretability.

4.5 Token partition

We previously defined linguistic features essential for expressing the temporal relationships between events (§ 3.1). Token partitioning is guided by this definition. The positive token partition is defined as all individual tokens that express or clarify the temporal relationship between two events, such as temporal expressions, prepositions, conjunctions, and verbs demonstrating tense and aspect. The negative token partition is defined as tokens that are not part of the positive partition and do not match the relevant tokens for the event pair, deemed irrelevant for expressing the temporal relationship. Figure 1 shows the relevant tokens for an instance, and how these define the partition of tokens.

4.6 Counterfactual Explanations

Counterfactual explanations are crafted from 300 instances randomly selected from the validation dataset, with minimal modifications to the original input. The queried event pair to the temporal relation is kept intact³, and changes are limited

³Alterations often involve reversing verb tenses. Since event pairs are defined by the verb’s base form and English verb tenses are structured flexibly, most instances can be altered without changing the original event pair. However, shifting to perfect tenses (e.g., “will finish,” “had finished”), which useful to indicate completed events isn’t always possible.

| | F1 M/avg | F1 W/avg | EM |
|-------------------------------|----------|----------|------|
| LUKE _{large} | 0.54 | 0.70 | 0.70 |
| LUKE _{base} | 0.55 | 0.67 | 0.68 |
| RoBERTa _{large} | 0.58 | 0.70 | 0.72 |
| RoBERTa _{base} | 0.56 | 0.69 | 0.69 |
| BERT _{large-uncased} | 0.58 | 0.69 | 0.69 |
| BERT _{base-uncased} | 0.52 | 0.66 | 0.66 |

Table 4: Performance of different models on the MATRES (Ning et al., 2018) dataset.

to the surrounding context. The alteration process involves a two-stage approach: (a) identifying the positive partition of tokens (§ 4.5), likely to impact predictions significantly, and (b) modifying these to change the temporal relationship.

We made alterations of four types, presented in Table 2. About 67% of instances are altered by reversing temporal conjunctions (e.g., modifying ”before” or ”after”), or adding modifiers or temporal expressions. This strategy is often applied to alter BEFORE-AFTER relationships, aligning with the dataset’s label distribution, where these are the most common labels. Less frequent methods like reversing phrase order ($\approx 12\%$) and changing conditional relationships ($\approx 21\%$) targeted the rarer EQUAL and VAGUE labels.

4.7 Saliency Scores

We obtain saliency scores from two different methods: Occlusion and Integrated Gradients (IG).

Occlusion (DeYoung et al., 2020) is a perturbation-based method. It works by systematically replacing the input token with a baseline token and observing the changes in the model’s output probabilities. The occlusion score for each token represents the change in the model’s output probability when the token is occluded. We select [MASK] as the baseline token to represent the absence of a specific feature. By replacing each token one at a time with [MASK], we remove the specific information provided by that specific token and observe how its absence affects the model’s output.

Integrated gradient (Sundararajan et al., 2017; Molnar, 2022) is a gradient-based method. This family of methods work by quantifying

| | Original | Counterfactual |
|-------------------------------|----------|----------------|
| LUKE _{large} | 0.66 | 0.45 |
| LUKE _{base} | 0.60 | 0.43 |
| RoBERTa _{large} | 0.67 | 0.43 |
| RoBERTa _{base} | 0.63 | 0.41 |
| BERT _{large-uncased} | 0.62 | 0.40 |
| BERT _{base-uncased} | 0.61 | 0.44 |

Table 5: Performance on counterfactual vs. original instances (measured as F1 W/avg).

how much each token in an input contributes to the gradient being propagated downstream. Tokens that have larger impact on the output will impact the gradient more, and are considered more influential. IG work by comparing the actual input against a baseline. We again select [MASK] as the baseline token, and create baselines based on the length of the original input. Gradients are computed along a linear path, from baseline to actual input, representing a transition from absence of features to the actual input. The gradients are accumulated at multiple steps along the path. The result is a vector for each token, representing a separate gradient value for each of a feature’s dimension. We convert these vectors into a single score per token by applying $L2$ normalization (Ray Choudhury et al., 2022).

Applying each saliency method results in four scores per token, representing the individual token’s impact on a specific class of the MATRES dataset. We aggregate these scores into a single value by summing⁴ over each score. The resulting score indicates the token’s overall significance across all classes. Special tokens, introduced during fine-tuning (§ 4.3), must be carefully considered. For IG, the special tokens are included in the baseline inputs, to ensure the integrity of the input. For Occlusion, they are not perturbed, allowing to measure the impact of regular tokens on the representation of the special tokens, which in turn affects the model’s predictions.

⁴Summing or averaging are common approaches for representing the influence of a token across classes (Molnar, 2022; Atanasova et al., 2020a). Both might overlook the importance of tokens that are particularly influential for a specific class.

| | Alignment | |
|-------------------------------|-----------|-----------|
| | IG | Occlusion |
| LUKE _{large} | 0.19 | 0.20 |
| LUKE _{base} | 0.21 | 0.18 |
| RoBERTa _{large} | 0.11 | 0.21 |
| RoBERTa _{base} | 0.27 | 0.17 |
| BERT _{large-uncased} | 0.52 | 0.25 |
| BERT _{base-uncased} | 0.56 | 0.28 |

Table 6: Alignment score between correctly predicted portions of counterfactual instances and saliency methods for each model.

4.8 Explanation Alignment Score

Recalling § 3.2, explanation alignment happens when a model accurately predicts both counterfactual and original instances, using the right cues, as indicated by saliency scores. We calculate an alignment score from the 300 instances where both original and counterfactual predictions are accurate. The score reflects the proportion of instances where the positive partition of tokens has a statistically significant higher average saliency score than the negative partition, suggesting reliance on correct information⁵. We use a one-tailed independence T-test at a 0.05 p-value to assess statistical significance, testing the null hypothesis that positive tokens do not have higher average saliency scores than negative ones, as per Ray Choudhury et al. (2022).

5 Results & Analysis

5.1 Model Evaluation

Table 4 shows the performance of fine-tuned models on the MATRES dataset. Across all models, weighted F1-scores consistently exceed macro F1-scores, indicating challenges in predicting minority classes, such as VAGUE. LUKE and RoBERTa models exhibit similar performance metrics, with their larger variants showing marginal improvements. However, these improvements are limited. BERT models show similar trend in improved performance when scaled, but they underperform relative to other models. This suggests that the notion, that larger models might perform

⁵For a single instance with a random partition of tokens, the positive and negative partitions should have similar saliency scores. For a dataset this translates to them being significantly different in $\approx 0\%$ of cases.

[CLS] in competitions against the clock , some athletes display an ability to seize control . think of the clark - kent - to - superman routines that john el ##way and michael jordan often pulled in the final seconds . but ira ##m leon stands on the side ##lines of his own race against time . (...) medical science is advancing at a rate that doesn't pre ##cl ##ude the development of a treatment , but it 's not clear if it will come in time ! ! no one knows what technology will be available in five years , ' said allan friedman , duke university hospital ne ##uro ##sur ##ge ##on in chief , who in 2011 removed as much of leon 's brain tumor as possible . (...) " but leon can still run . two years after his brain - cancer diagnosis , he recently ran a sub - five - minute mile for the first time since high school . what has startled the medical community even more is what leon did this month in beaumont , texas : he [a1] won [a1] the gus ##her marathon , finishing in 3 : 07 : 35 : that was one second slower than his personal record in the 26 . 2 : mile event , [a2] set [a2] days before he underwent brain surgery in early 2011 . [SEP]

Figure 3: Visualization of saliency scores obtained by occlusion for one instance, performed on BERT-large. The model correctly predicts the temporal relation between "won" and "set". More saturated tokens indicate higher saliency.

better for some use cases (Zhong et al., 2021; Desai and Durrett, 2020), only partially holds true for a temporal relation classification on the MATRES dataset.

5.2 Counterfactual Evaluation

Table 5 shows a comparison of F1-weighted average scores for the selected models on 300 original versus counterfactual instances. For all models, we observe a significant decrease in performance on counterfactual instances compared to the original instances, with an average performance drop of 20%. This indicates overall challenges in maintaining expected reasoning when the conditions change. Contrary to expectations, larger model variants show a bigger performance drop. This indicate that larger models are less likely to perform well on altered inputs than their smaller variants. Future work could consider relaxing the criteria that a model's prediction on a counterfactual scenario must perfectly align with the true class. Instead, by analyzing prediction probabilities, we might show that models appropriately adjust their probabilities in response to counterfactual changes. This is particularly valuable for classification with unbalanced distribution of labels (Molnar, 2022).

5.3 Explanation Alignment

Table 6 shows the explanation alignment score between correctly predicted counterfactual instances against the two selected saliency-based methods. We observe that IG and Occlusion do not agree on the alignment scores. This lack of agreement between the two methods is consistent with previous findings (Ray Choudhury et al., 2022; Atanasova

et al., 2020a), and it must be addressed to draw appropriate conclusions.

The alignment scores with IG indicate that smaller models, when making correct predictions for both original and counterfactual cases, are more likely to rely on relevant information compared to larger models. In contrast, Occlusion shows no consistent trend across model sizes, with scores that do not favor either smaller or larger models.

Potential interpretations for this inconsistency have been suggested. One interpretation is that IG may struggle to compute accurate saliency scores due to the discrete nature of text data (Harbecke and Alt, 2020), as the intermediate representations required do not align well with discrete word embeddings (Zhao et al., 2023), and therefore the computed gradients might not produce truthful saliency scores. Occlusion, potentially more stable, demonstrates no clear trend favoring model sizes. Another possible interpretation is that IG are in fact more faithful (Ray Choudhury et al., 2022). The trend shown by IG suggests that as the model’s size increases, the features we define as important do not align with the model’s strategies for correct predictions. Larger models, with their increased capacity, might be more likely to learn complex statistical patterns in the training data, including spurious ones. If the training data contain many such correlations, a larger model might be more prone to learn them and use them for predictions (Linzen, 2020). This could explain the higher accuracy of larger models compared to smaller ones (§ 5.1), but also indicates that larger models might depend on spurious patterns instead of relevant information (essentially, being right for the wrong reasons).

Overall, while the reasons behind inconsistencies remain unclear, the findings question the reliability of the selected saliency-based methods in evaluating model reasoning. Further work might include alternatives for computing saliency scores, such as surrogate models LIME (Ribeiro et al., 2016) and SHAP (Lundberg and Lee, 2017).

6 Discussion

This study investigates selected discriminative models on a temporal relation classification task. While numerous benchmarks have been developed to evaluate models’ temporal processing abilities, our experiments highlight limitations in these

evaluations. Specifically, we adopted one commonly used benchmark dataset and found that models can achieve high accuracy without following the expected reasoning patterns. The framework used in this study offers a step toward improving evaluation methodologies by emphasizing whether models make correct predictions for the right reasons. It establishes clear success criteria for the task and highlights the role of validating “reasoning” to accurately assess model performance.

Post-hoc and local explainability methods are commonly used to determine if model decisions are justifiable from a human perspective, yet their reliability and utility is often questioned (Dasgupta et al., 2022; Saini and Prasad, 2022). Counterfactual explanations are considered as more truthful (Zhao et al., 2023), but require careful handling to prevent unreliable conclusions. Saliency scores, on the other hand, may not reflect the model’s decision-making process. Different saliency methods can produce conflicting results, meaning that they inconsistently reflect the model’s decision process (Jukić et al., 2023; Ding and Koehn, 2021b; Atanasova et al., 2020b). Moreover, the lack of a ground truth for saliency evaluation makes it challenging to evaluate whether they correctly approximate the model’s processes (Molnar, 2022).

Having addressed the truthfulness of these methods, the question of their utility remains. For this study, we must conclude that the models follow some other strategy for correct prediction (rather than relying on the expected reasoning). Explainability methods should aim to make a model’s decisions understandable to humans. However, this is challenging when a model’s reasoning processes do not align with human reasoning (González et al., 2021). Identifying alternative reasoning strategies or shortcuts through these explanations is challenging because they are not necessarily human interpretable (see Figure 3⁶), raising questions about the practical value of these methods, as they only provide a partial interpretable view of a model’s processes, and fail to provide actionable insights.

⁶Similar work (Ray Choudhury et al., 2022; Du et al., 2021) report both negative and positive impacts on saliency scores, which we consider as positive contributions regardless of probability direction.

7 Extending to Generative Models

Extending the experiments to adapt modern generative models, such as LLaMA (Touvron et al., 2023), GPT (Yenduri et al., 2023), and OLMO (Groeneveld et al., 2024), presented challenges, particularly in interpreting saliency scores.

Zhao et al. (2023) provides a taxonomy of explainability methods for transformer-based language models, categorizing them based on training paradigms (e.g., fine-tuning and prompting), which influence their goals and effectiveness. Generative models, primarily prompt-based, leverage their extensive scale and learned prompts for task execution. These complex processing strategies (Wei et al., 2023) make it difficult to isolate specific components of the model responsible for particular decisions. Localized and example-based explainability methods become less meaningful (Zhao et al., 2023). Moreover, differences in training objectives (e.g. autoregressive versus masked language), make it challenging to apply explainability methods that work reliably across all model types. Trustworthiness of explanations is both task and model-dependent (Bastings et al., 2022). Variations in how models process and prioritize input can result in inconsistencies in the effectiveness of these methods. This variability underscores that no single explanation method can be universally treated as a standard across all contexts. Consequently, conducting meaningful comparisons between different architectures becomes challenging, as the results may be unreliable or even misleading. Further research is needed to validate the robustness of such comparative analyses.

In contrast, counterfactual explanations provide a promising approach for evaluating generative models. Assessments centered on counterfactual instances could help determine whether these models maintain consistent reasoning when confronted with alternative scenarios. We leave the adaption of the presented counterfactual explanations (§ 4.6) to generative models to future work.

Of particular relevance, Roccabruna et al. (2024) highlights the performance gap between generative and discriminative models in temporal relation classification tasks. Encoder-only models based on RoBERTa consistently outperform generative models like LLaMA. This performance gap is attributed to RoBERTa’s ability to fully utilize input context via masked language modeling, in contrast to LLaMA’s autoregressive objective,

which tends to prioritize final tokens in the input sequence. This underscores the significance of discriminative models for TRC and reinforces the value of evaluating whether their decisions are based on valid and expected reasoning patterns.

8 Conclusion

Temporal annotations are used to mark all linguistic features that express temporal information in text. We evaluate selected discriminative models on a temporal relation classification task, examining whether they rely on these features for correct predictions. Experiments involve a combination of counterfactual explanations and saliency-based methods. High alignment between these two explanations indicates that a model is following a valid processing strategy. We find that this is not the case for the selected models, meaning that they might learn spurious correlations or shortcuts rather than relying on the defined linguistic features that form temporal meaning. We evaluate the limitations of this framework by examining the utility of the explainability methods used, together with challenges and potential directions for extending the framework to generative models.

Limitations

This study focuses on a single dataset and task, which limits the generalizability of its findings. Future work could expand the scope by exploring additional benchmark datasets and tasks to assess the broader applicability of the proposed framework. Generating and testing a larger number of counterfactual and original instance pairs would also provide a more robust evaluation.

Our approach to saliency scores may attract additional attention. The current methodology does not account for the potential negative impact of individual tokens on predictions, and it aggregates all scores without identifying specific tokens that are particularly influential for a given class.

Acknowledgment

Many thanks to Anna Rogers for her invaluable guidance and support.

References

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020a. A diagnostic

- study of explainability techniques for text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3256–3274, Online. Association for Computational Linguistics.
- Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020b. A diagnostic study of explainability techniques for text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3256–3274, Online. Association for Computational Linguistics.
- Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2895–2905, Florence, Italy. Association for Computational Linguistics.
- Jasmijn Bastings, Sebastian Ebert, Polina Zablotskaia, Anders Sandholm, and Katja Filippova. 2022. “will you find these shortcuts?” a protocol for evaluating the faithfulness of input salience methods for text classification. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 976–991, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Samuel R. Bowman and George E. Dahl. 2021. What will it take to fix benchmarking in natural language understanding?
- C. Callender. 2011. *The Oxford Handbook of Philosophy of Time*.
- Supriyo Chakraborty, Richard Tomsett, Ramya Raghavendra, Daniel Harborne, Moustafa Alzantot, Federico Cerutti, Mani Srivastava, Alun Preece, Simon Julier, Raghuvier M. Rao, Troy D. Kelley, Dave Braines, Murat Sensoy, Christopher J. Willis, and Prudhvi Gurram. 2017. Interpretability of deep learning models: A survey of results. In *2017 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computed, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCOM/IOP/SCI)*, pages 1–6.
- Wenhu Chen, Xinyi Wang, and William Yang Wang. 2021. A dataset for answering time-sensitive questions.
- Kiel Christianson. 2016. When language comprehension goes wrong for the right reasons: Good enough, underspecified, or shallow language processing. *Quarterly journal of experimental psychology (2006)*, 69:1–29.
- Divyanshu Daiya. 2020. Combining temporal event relations and pre-trained language models for text summarization. In *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 641–646.
- Sanjoy Dasgupta, Nave Frost, and Michal Moshkovitz. 2022. Framework for evaluating faithfulness of local explanations.
- Mostafa Dehghani, Yi Tay, Alexey A. Gritsenko, Zhe Zhao, Neil Houlsby, Fernando Diaz, Donald Metzler, and Oriol Vinyals. 2021. The benchmark lottery.
- Shrey Desai and Greg Durrett. 2020. Calibration of pre-trained transformers.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.
- 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.
- Shuoyang Ding and Philipp Koehn. 2021a. Evaluating saliency methods for neural language models.
- Shuoyang Ding and Philipp Koehn. 2021b. Evaluating saliency methods for neural language models.
- Mengnan Du, Varun Manjunatha, Rajiv Jain, Ruchi Deshpande, Franck Dernoncourt, Jiuxiang Gu, Tong Sun, and Xia Hu. 2021. Towards interpreting and mitigating shortcut learning behavior of nlu model.
- Anoushka Gade and Jorjeta Jetcheva. 2024. It’s about time: Incorporating temporality in retrieval augmented language models.
- Ana Valeria González, Anna Rogers, and Anders Søgaard. 2021. On the interaction of belief bias and explanations. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2930–2942, Online. 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.

- Rujun Han, Xiang Ren, and Nanyun Peng. 2021. ECONET: Effective continual pretraining of language models for event temporal reasoning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5367–5380, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- David Harbecke and Christoph Alt. 2020. Considering likelihood in NLP classification explanations with occlusion and language modeling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 111–117, Online. Association for Computational Linguistics.
- 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 common-sense 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. Association for Computational Linguistics.
- Josip Jukić, Martin Tutek, and Jan Šnajder. 2023. Easy to decide, hard to agree: Reducing disagreements between saliency methods.
- Dohwan Ko, Ji Lee, Woo-Young Kang, Byungseok Roh, and Hyunwoo Kim. 2023. Large language models are temporal and causal reasoners for video question answering. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4300–4316, Singapore. Association for Computational Linguistics.
- 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, Minnesota, USA. Association for Computational Linguistics.
- Tal Linzen. 2020. How can we accelerate progress towards human-like linguistic generalization? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5210–5217, Online. 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.
- Scott Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions.
- Inderjeet Mani, Marc Verhagen, Ben Wellner, Chong Min Lee, and James Pustejovsky. 2006. Machine learning of temporal relations. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 753–760, Sydney, Australia. Association for Computational Linguistics.
- Puneet Mathur, Rajiv Jain, Franck Dernoncourt, Vlad Morariu, Quan Hung Tran, and Dinesh Manocha. 2021. TIMERS: Document-level temporal relation extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 524–533, Online. Association for Computational Linguistics.
- R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference.
- Christoph Molnar. 2022. *Interpretable Machine Learning*. LeanPub.
- Aakanksha Naik, Luke Breitfeller, and Carolyn Rose. 2019. TDDiscourse: A dataset for discourse-level temporal ordering of events. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 239–249, Stockholm, Sweden. Association for Computational Linguistics.
- Qiang Ning, Sanjay Subramanian, and Dan Roth. 2019. An improved neural baseline for temporal relation extraction. 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 6203–6209, Hong Kong, China. Association for Computational Linguistics.
- Qiang Ning, Hao Wu, Rujun Han, Nanyun Peng, Matt Gardner, and Dan Roth. 2020. TORQUE: A reading comprehension dataset of temporal ordering questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1158–1172, Online. Association for Computational Linguistics.
- Qiang Ning, Hao Wu, and Dan Roth. 2018. A multi-axis annotation scheme for event temporal relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1318–1328, Melbourne, Australia. Association for Computational Linguistics.
- James Pustejovsky, Patrick Hanks, Roser Saurí, Andrew See, Rob Gaizauskas, Andrea Setzer, Dragomir Radev, Beth Sundheim, David Day, Lisa Ferro, and Marcia Lazo. 2003. The timebank corpus. *Proceedings of Corpus Linguistics*.
- James Pustejovsky, Kiyong Lee, Harry Bunt, and Laurent Romary. 2010. ISO-TimeML: An international standard for semantic annotation. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC’10)*, Valletta, Malta. European Language Resources Association (ELRA).

- Lianhui Qin, Aditya Gupta, Shyam Upadhyay, Luheng He, Yejin Choi, and Manaal Faruqui. 2021. Temporal: Temporal commonsense reasoning in dialog.
- Yifu Qiu, Zheng Zhao, Yftah Ziser, Anna Korhonen, Edoardo M. Ponti, and Shay B. Cohen. 2023. Are large language models temporally grounded?
- Adel Rahimi and Shaurya Jain. 2022. Testing the effectiveness of saliency-based explainability in nlp using randomized survey-based experiments. *ArXiv*, abs/2211.15351.
- Sagnik Ray Choudhury, Anna Rogers, and Isabelle Augenstein. 2022. Machine reading, fast and slow: When do models “understand” language? In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 78–93, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.
- Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. “why should i trust you?”: Explaining the predictions of any classifier.
- Gabriel Roccabruna, Massimo Rizzoli, and Giuseppe Riccardi. 2024. Will LLMs replace the encoder-only models in temporal relation classification?
- Anna Rogers, Marzena Karpinska, Ankita Gupta, Vladislav Lialin, Gregory Smelkov, and Anna Rumshisky. 2022. Narrativetime: Dense temporal annotation on a timeline.
- Aditya Saini and Ranjitha Prasad. 2022. Select wisely and explain: Active learning and probabilistic local post-hoc explainability.
- Hendrik Schuff, Alon Jacovi, Heike Adel, Yoav Goldberg, and Ngoc Thang Vu. 2022. Human interpretation of saliency-based explanation over text. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT ’22. ACM.
- Xiaoming Shi, Siqiao Xue, Kangrui Wang, Fan Zhou, James Y. Zhang, Jun Zhou, Chenhao Tan, and Hongyuan Mei. 2023. Language models can improve event prediction by few-shot abductive reasoning.
- Nina Spreitzer, Hinda Haned, and Ilse van der Linden. 2022. Evaluating the practicality of counterfactual explanations. In *Workshop on Trustworthy and Socially Responsible Machine Learning, NeurIPS 2022*.
- Saku Sugawara, Pontus Stenetorp, Kentaro Inui, and Akiko Aizawa. 2019. Assessing the benchmarking capacity of machine reading comprehension datasets.
- Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks.
- Xia Sitt Fankhauser Chicas-Mosier Monteith Tai, Bentley. 2024. An examination of the use of large language models to aid analysis of textual data.
- Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2023a. Towards benchmarking and improving the temporal reasoning capability of large language models.
- Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2023b. Towards benchmarking and improving the temporal reasoning capability of large language models.
- Qingyu Tan, Hwee Tou Ng, and Lidong Bing. 2023c. Towards benchmarking and improving the temporal reasoning capability of large language models.
- 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.
- Naushad UzZaman, Hector Llorens, Leon Derczynski, James Allen, Marc Verhagen, and James Pustejovsky. 2013. SemEval-2013 task 1: TempEval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9, Atlanta, Georgia, USA. Association for Computational Linguistics.
- Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Graham Katz, and James Pustejovsky. 2007. SemEval-2007 task 15: TempEval temporal relation identification. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 75–80, Prague, Czech Republic. Association for Computational Linguistics.
- Jerry 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.
- Jason Weston, Antoine Bordes, Sumit Chopra, Alexander M. Rush, Bart van Merriënboer, Armand Joulin, and Tomas Mikolov. 2015. Towards ai-complete question answering: A set of prerequisite toy tasks.
- Siheng Xiong, Ali Payani, Ramana Kompella, and Faramarz Fekri. 2024. Large language models can learn temporal reasoning.
- Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. Luke: Deep contextualized entity representations with entity-aware self-attention.
- Jing Yang, Yu Zhao, Linyao Yang, Xiao Wang, Long Chen, and Fei-Yue Wang. 2024. Temprompt: Multi-task prompt learning for temporal relation extraction in rag-based crowdsourcing systems.

- Sen Yang, Dawei Feng, Linbo Qiao, Zhigang Kan, and Dongsheng Li. 2019. Exploring pre-trained language models for event extraction and generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5284–5294, Florence, Italy. Association for Computational Linguistics.
- Guy Yanko, Shahaf Pariente, and Kfir Bar. 2023. Temporal relation classification in Hebrew. In *Findings of the Association for Computational Linguistics: IJCNLP-AACL 2023 (Findings)*, pages 261–267, Nusa Dua, Bali. Association for Computational Linguistics.
- Gokul Yenduri, Ramalingam M, Chemmalar Selvi G, Supriya Y, Gautam Srivastava, Praveen Kumar Reddy Maddikunta, Deepti Raj G, Rutvij H Jhaveri, Prabadevi B, Weizheng Wang, Athanasios V. Vasilakos, and Thippa Reddy Gadekallu. 2023. Generative pre-trained transformer: A comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions.
- 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. Association for Computational Linguistics.
- Shuaicheng Zhang, Qiang Ning, and Lifu Huang. 2022. Extracting temporal event relation with syntax-guided graph transformer. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 379–390, Seattle, United States. Association for Computational Linguistics.
- Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2023. Explainability for large language models: A survey.
- Ruiqi Zhong, Dhruva Ghosh, Dan Klein, and Jacob Steinhardt. 2021. Are larger pretrained language models uniformly better? comparing performance at the instance level. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3813–3827, Online. Association for Computational Linguistics.
- Ben Zhou, Daniel Khashabi, Qiang Ning, and Dan Roth. 2019. “going on a vacation” takes longer than “going for a walk”: A study of temporal commonsense understanding. 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 3363–3369, Hong Kong, China. Association for Computational Linguistics.
- Ben Zhou, Kyle Richardson, Qiang Ning, Tushar Khot, Ashish Sabharwal, and Dan Roth. 2021. Temporal reasoning on implicit events from distant supervision. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1361–1371, Online. Association for Computational Linguistics.
- Jie Zhou, Shenpo Dong, Hongkui Tu, Xiaodong Wang, and Yong Dou. 2022. RSGT: Relational structure guided temporal relation extraction. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2001–2010, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Appendix A

BERT (bert-base-uncased, bert-large-uncased), RoBERTa (FacebookAI/roberta-base, FacebookAI/roberta-large), LUKE (studio-ousia/luke-base, studio-ousia/luke-large) are sourced from the Hugging Face Transformers library. Each encoder model is fine-tuned for the task of temporal relation classification using the architectural and tokenisation strategies presented by Yanko et al. (2023) and Baldini Soares et al. (2019). All models are fine-tuned for the duration of 10 epochs with a batch-size of 8, using AdamW optimizer. The learning rate was kept at 1e-05.

Appendix B

| | Relaxed F1
M/avg | Relaxed F1
W/avg | EM |
|-------------------------------|---------------------|---------------------|------|
| LUKE _{large} | 0.61 | 0.81 | 0.80 |
| LUKE _{base} | 0.61 | 0.80 | 0.78 |
| RoBERTa _{large} | 0.67 | 0.82 | 0.81 |
| RoBERTa _{base} | 0.65 | 0.81 | 0.79 |
| BERT _{large-uncased} | 0.66 | 0.81 | 0.78 |
| BERT _{base-uncased} | 0.63 | 0.79 | 0.77 |

Table: Performance evaluation on MATRES (Ning et al., 2018) dataset, using the "relaxed" F1 metric proposed by Yanko et al. (2023).

VAGUE class was initially introduced in MATRES dataset to account for disagreements that arise during the annotation process (Ning et al., 2018). Yanko et al. (2023) introduces a "related F1" metric to address the complexities associated with the class. This evaluation metric excludes errors where non-VAGUE predictions are made on VAGUE samples, based on the argument that VAGUE inherently encompasses both temporal directions (BEFORE and AFTER). Errors in this class are considered less critical and can be partially disregarded. Similarly, Roccabruna et al. (2024) take this notion further by completely excluding the VAGUE class from analysis, arguing that it does not represent a true temporal relation. We chose to keep the VAGUE class due to its potential value in generating counterfactual explanations. The class can serve as a middle ground that can be modified into more definitive temporal relations (BEFORE, AFTER or EQUAL) or created by introducing ambiguities into otherwise clear relationships.

Appendix C

This section provides a detailed overview of the methods used to generate counterfactual explanations, including how alterations were identified and implemented to ensure semantic correctness. Four types of possible and semantically correct alterations were employed to generate counterfactual explanations:

1. We consider simple temporal relationships those that contain explicit temporal conjunctions (e.g. "before", "after" and "while"). For simple temporal relationships, revering the temporal conjunction and/or changing verb tenses were sufficient as semantically correct alterations. This strategy most often resulted in reversing BEFORE and AFTER relationships.
2. For instances where a direct reversal of temporal conjunction or verb tense change was not possible, temporal conjunctions or adverbs (e.g. "subsequently", "already") and temporal expressions (e.g. "months", "years") were added or removed. This strategy often resulted in altering BEFORE or AFTER relationships to an EQUAL relationship, or vice-versa.
3. We consider more complex relationships those that include conditional or causal relationships between the two events. Focus was put in not altering the nature of such relationships. For these cases, reversing the temporal relationship involved reversing the cause with the effect or vice-versa.
4. For actions described in separate sentences, reordering the sentences was considered as a valid semantic alteration. This alteration is possible and particularly relevant for the dataset at hand, which is based on news snippets. For the news domain, the order of mention often dictates the sequence of events. This strategy often resulted in altering to or from a VAGUE relationship. Reordering sentences within the text, by placing them closer or further apart, either increased or decreased the contextual dependency between a pair of actions.

| | Common Features | Examples |
|--|--|---|
| Temporal Expressions:
Tokens that specify points in time | Absolute expressions, such as
<i>December 2025, at 5PM</i> | She started a new job on September 1st , after moving to the city. |
| | Relative expressions, such as
<i>week, Mondays, annually</i> | If it rains tomorrow , the picnic will be postponed until Sunday . |
| Temporal Prepositions and Adverbs: Tokens used to connect actions or events to specific times. | Prepositions such as
<i>at, on, in, during, for, over, by</i> | She started a new job on September 1st, after moving to the city. |
| | Adverbials such as
<i>again, late, now, then eventually, previously, recently</i> | Recently , he has taken up running before breakfast at 8AM. |
| Temporal Conjunctions: Tokens used to related events to each one another. | Conjunctions such as
<i>before, after, while, until, since when, as soon as, as long as</i> | She started a new job on September 1st, just after moving to the city. |
| | | Recently, he has taken up running before breakfast every morning. |
| Subordinate Conjunction: Tokens used to express conditional or causal relationship between events or actions. | References to causality such as
<i>because, therefore, as</i> | Because you didn't reply in time, I only bought tickets for two. |
| | References to conditions such as
<i>if, unless, then, so</i> | If it rains tomorrow, then the picnic will be postponed until Sunday at noon. |

Appendix D: Examples of features that express temporal information. The table is designed to demonstrate how relevant and important tokens are identified and retrieved in accordance with the annotation guidelines. Color coding follows the annotation guidelines from TimeML (Mani et al., 2006): **orange** is used for signal tokens (SIGNAL), providing cues for how events and temporal expressions are related to each other; **blue** is used for specific time expressions (TIMEX3).

Benchmarking Abstractive Summarisation: A Dataset of Human-authored Summaries of Norwegian News Articles

Samia Touileb¹, Vladislav Mikhailov², Marie Kroka¹, Lilja Øvrelid², Erik Velldal²

¹University of Bergen, ²University of Oslo,
samia.touileb@uib.no, vladism@ifi.uio.no,
liljao@ifi.uio.no, erikve@ifi.uio.no

Abstract

We introduce a dataset of high-quality human-authored summaries of news articles in Norwegian¹. The dataset is intended for benchmarking the abstractive summarisation capabilities of generative language models. Each document in the dataset is provided with three different candidate gold-standard summaries written by native Norwegian speakers, and all summaries are provided in both of the written variants of Norwegian – Bokmål and Nynorsk. The paper describes details on the data creation effort as well as an evaluation of existing open LLMs for Norwegian on the dataset. We also provide insights from a manual human evaluation, comparing human-authored to model-generated summaries. Our results indicate that the dataset provides a challenging LLM benchmark for Norwegian summarisation capabilities.

1 Introduction

One of the key practical use cases of large language models (LLMs), is to generate condensed summaries of texts. Several news publishers already include LLM-generated summaries as part of the news stories they publish. Evaluating such generated summaries, however, remains a challenge. For Norwegian, one important reason for this is the lack of gold-standard summaries to compare to. The current paper introduces a new and open dataset of high-quality human-authored summaries of news articles in Norwegian, covering both of the official written variants; Bokmål (BM) and Nynorsk (NN). Aiming to make benchmarking as robust as possible, each document in

the dataset is provided with three different candidate gold-standard summaries (for each variant, BM and NN, resulting in six summaries in total for each news article).

The remainder of the paper is structured as follows. We first describe the creation of the human-authored summaries, including the underlying data sources, the annotator guidelines, and corpus statistics. We then move on to describe a first set of experiments with using pre-trained LLMs to generate summaries, and then evaluate them using our new dataset. Importantly, we here also present the methodology and framework we use, including factors like prompts and metrics. We thereafter discuss in detail the setup and results of our manual human evaluation.

2 Related work

Summarisation datasets are foundational for advancing the development of techniques for automatic summarisation, as well as for benchmarking LLMs. There are various approaches developed to address diverse summarisation challenges, along with influential datasets to benchmark both extractive and abstractive methods (Dong et al., 2022; El-Kassas et al., 2021). Most works on benchmark datasets have been done for English, and we here mention some of the works that focus on summarising news articles.

The CNN/Daily Mail dataset (Hermann et al., 2015) is one of such influential works. This dataset was created for the task of reading comprehension, but is widely used for summarisation-related tasks. The dataset consists of news articles accompanied by a set of bullet points representing (abstractive) summaries. Subsequent works have focused on creating resources for various domains, contexts, and summarisation styles. For instance, Gigaword (Rush et al., 2015) is extracted from the Gigaword news corpus and contains sentences paired with short summaries (headlines). This

¹<https://github.com/SamiaTouileb/NorSumm/tree/main> and <https://huggingface.co/datasets/SamiaT/NorSumm/tree/main>

is also an abstractive dataset enabling sentence-level summarisation. It has however been criticised for only including headlines instead of full summaries (El-Kassas et al., 2021). The extreme summarisation (XSum) dataset (Narayan et al., 2018) was also created from news articles, sourced from BBC. Each article in this dataset is paired with a one-sentence summary representing a concise and abstractive summary. The CNN-corpus (Lins et al., 2019) contains news articles from CNN paired with highlights and gold-standard abstractive summaries. However, the corpus is mostly used for extractive summarisation tasks (El-Kassas et al., 2021). Efforts have also been made for multi-document summarisation, such as Multi-News (Fabbri et al., 2019), which contains relatively long summaries of multi-news articles covering the same topic.

Resources for news summarisation in Norwegian are notably scarce. Some efforts to introduce summarization datasets in Norwegian have relied on machine translation, e.g. based on the CNN/DailyMail data (Liu et al., 2024). However, failing to adequately capture nuances of the target language, as machine translation may produce non-idiomatic and non-natural-sounding language. Another concern is that, being based on English sources, the original texts are typically not geared towards issues of primary salience to a Norwegian context (whether socially, politically, geographically, or otherwise), which is unfortunate if the goal is to benchmark Norwegian LLMs.

To our knowledge, no freely available, manually curated summarization dataset, created from scratch for Norwegian news data exists, making this work a valuable contribution to advancing research in this field.

3 Human authored summaries

Data sources We use a subset of the news articles in the Norwegian event extraction dataset EDEN (Touileb et al., 2024) as the data source for summarisation. EDEN contains articles in BM only, and because creating summaries based on news articles is a time and effort intensive task, we here only make use of the dev and test splits of EDEN, which respectively contain 30 and 33 news articles. EDEN was chosen due to its high-quality, as it comprises news articles from the Norwegian Dependency Treebank (Solberg et al., 2014; Øvrelid and Hohle, 2016), and is

a richly annotated dataset covering event triggers and arguments (Touileb et al., 2024), named entities (Jørgensen et al., 2019), morphosyntactic annotation, and co-reference information (Mæhlum et al., 2022).

Annotators We hired three annotators with strong academic backgrounds related to journalism, all Norwegian native speakers. The annotators were fairly compensated following an hourly contract, and were hired for a period of 6 months. All annotators have a background in media science or journalism. The first annotator, has a bachelor’s degree in media and communication science, and has worked as a freelance journalist. The second annotator has a bachelor’s degree in journalism, and was finishing up a master’s degree in investigative journalism while doing an internship in a leading Norwegian news broadcasting company. The third annotator, a journalism student, who also worked part-time as a journalist in a local Norwegian newspaper.

All hired annotators have experience writing news articles, including the identification of key information that should be selected to write the article. As the task was to create natural-sounding summaries that preserved the original meaning of the news articles, we believe that these annotators can be referred to as domain experts. In addition, as we wanted the summaries to be as natural-sounding as possible, we asked the annotators to write in their preferred variant of Norwegian. This has resulted in two annotators writing in BM, and one annotator writing in NN.

Guidelines The annotators received a detailed set of guidelines outlining the steps to follow when authoring the summaries. The guidelines were inspired by concrete prompts, shared with us, and which were used to automatically generate summaries of news articles by one of the biggest media companies in Norway.

We asked the annotators to write summaries that reflect the main content of the news articles, but without providing strong limitations to their language use or formulations. We aimed to create summaries that are as natural-sounding as possible, and as diverse as possible. Each annotator was free to write their own summaries, without consulting or discussing details about the content of the summaries. However, we provided the annotators with an example consisting of a news article

| News article | | | |
|---|---|---|---|
| <p>Mer frukt, men mindre norsk Forbruket økt med 20 prosent på ti år. Forbruket av frukt og grønt har økt med over 20 prosent i løpet av de siste ti årene. Men den norske produksjonen faller. Hele veksten og mer til av frukt og grønt kommer fra import. Den norske produksjonen har nemlig falt med 10 prosent siden 1998, skriver Nationen. Ifølge landbruks- og matminister Lars Peder Brekk (Sp) må de norske kjedene bli flinkere til å samarbeide med norske produsenter og bøndene må bli flinkere til å produsere det kundene vil ha.</p> | | | |
| Summary 1 | | Summary 2 | Summary 3 |
| Bokmål | <p>Forbruket av frukt og grønt har økt med 20 prosent på ti år. Hele veksten og mer til av frukt og grønt kommer fra import da den norske produksjonen har falt med 10 prosent siden 1998, skriver Nationen. De norske kjedene må bli flinkere til å samarbeide med norske produsenter og bøndene må bli flinkere til å produsere det kundene vil ha ifølge landbruks- og matminister Lars Peder Brekke (Sp).</p> | <p>Nordmenn kjøper mer frukt og grønnsaker, samtidig som de norske bøndene produserer mindre. Frukt- og grønt-forbruket har økt med over 20 prosent de 10 siste årene. Den norske produksjonen har falt med 10 prosent siden 1998. Import av varer dekker den økte etterspørselen i det norske markedet.</p> | <p>Forbruket av frukt og grønt har økt med 20 prosent i Norge de siste ti årene. Likevel falt den norske eksporten, og veksten kommer fra stadig mer import. Siden 1998 har den norske produksjonen falt med 10 prosent, opplyser Nationen. Landbruksministeren sier at kjedene må bli flinkere til å samarbeide med norske produsenter, og at bøndene i større grad må produsere det kundene ønsker.</p> |
| | <p>Forbruket av frukt og grønt har auka med 20 prosent på ti år. Heile veksten og meir av frukt og grønt kjem frå import då den norske produksjonen har falle med 10 prosent sidan 1998, skriv Nationen. Dei norske kjedane må bli flinkare til å samarbeide med norske produsenter og bøndene må bli flinkare til å produsera det kundene vil ha, ifølge landbruks- og matminister Lars Peder Brekke (Sp).</p> | <p>Nordmenn kjøper meir frukt og grønnsaker, samtidig som dei norske bøndene produserer mindre. Frukt- og grønt-forbruket har auka med over 20 prosent dei 10 siste åra. Den norske produksjonen har gått ned med 10 prosent sidan 1998. Import av varer dekker den auka etterspørselen i den norske marknaden.</p> | <p>Forbruket av frukt og grønt har auka med 20 prosent i Noreg dei siste ti åra. Likevel fell den norske eksporten, og veksten kjem frå meir og meir import. Sidan 1998 har nemleg den norske produksjonen falle med 10 prosent, opplyser Nationen. Landbruksministeren seier at kjedene må bli flinkare til å samarbeida med norske produsentar, og at bøndene må i større grad produsera kva kundane ynskjer.</p> |

Table 1: Example of a news article and the summaries written by three different native speakers in either Bokmål (BM) or Nynorsk (NN), and translated into the other respective variety.

paired with its summary to discuss the format and exemplify the concrete guidelines.

More concretely, the guidelines we provided the annotators are as follow:

- Make a short and precise summary.
- The summary should be formatted as a bulleted list, with each point on a single line.
- The language must be clear, precise, concise, and easy to understand.
- Journalistic integrity must be maintained, ensure that no errors are introduced.
- The summary must address the following questions: who, what, where, when, and why it is important to have knowledge of the case or event presented in the news article.
- The summary must be engaging and highlight key information from the article.

- The summary should have a maximum character count of 700, including spaces.

We intentionally decided to keep the annotation guidelines simple to give annotators the freedom to write in a natural and authentic style. Rather than imposing strict constraints, we provided them with general and broad instructions on the importance of maintaining journalistic integrity while clearly, precisely, and concisely creating an informative summary. We believe that this flexibility allowed annotators to create more natural and engaging summaries. Our choice of enforcing summaries formatted as bullet-points was in part based on how news outlets present machine-generated summaries in the Norwegian news. But also because we planned to perform a human evaluation where human-authored summaries will be compared to machine-generated summaries. See Section 6 for more details about this analysis.

Generation and evaluation The annotation process was carried out using a simple text editing platform, to provide the annotators a more straightforward and user-friendly interface. We had several meetings with the annotators to discuss the process and the progression of the task. However, we never aimed for aligning the content of the human-authored summaries. This was an intentional decision to create a benchmark dataset with diversity, as we believe that in the case of summarisation, there is no unique gold summary version. We wanted to create a resource that would provide three diverse summaries for each news article, in each of the written variants BM and NN.

The annotation was conducted in two rounds: (i) creating human-authored summaries, (ii) translating human-authored summaries. As previously mentioned, we gave the annotators the liberty to write in their preferred Norwegian written variant. This was to both ensure the creation of naturally-sounding summaries, but also to create a benchmark for both BM and NN.

In the first round of annotation, our three annotators authored 63 summaries each (30 from the dev split of EDEN and 33 from the test split), following our annotation guidelines. For the second round of annotation, two of our annotators translated all summaries from BM to NN, and vice versa. Here again, the annotators translated summaries to their preferred Norwegian variant.

Since translations between the two written variants were performed by another annotator, each human-authored summary has been seen and analysed by two different annotators. We believe that this enhances the quality of the summaries, as potential ambiguity or errors could be discovered and corrected in both versions. This process again allowed us to create additional human-authored summaries for each of BM and NN. We provide more details about the resulting dataset bellow.

Examples Table 1 shows three summaries originally written in either Bokmål or Nynorsk, and translated into the other respective variety.

Each summary varies in both content and length, with *Summary 1* being the longest and *Summary 2* being the shortest (in terms of tokens). We believe that this diversity contributes to a benchmark dataset that more accurately reflects the complexities of generated summaries. Each summary presents the news article in a unique way, emphasising different important aspects of

| | Ann. | #Summ. | #Sent | #Tokens | Avg. |
|--------------|------|--------|-------|---------|--------|
| BM | A1 | 63 | 365 | 6,695 | 106.26 |
| | A2 | 63 | 280 | 6,221 | 98.74 |
| | A3 | 63 | 312 | 6,472 | 102.73 |
| | | 189 | 957 | 19,042 | 102.58 |
| NN | A1 | 63 | 365 | 6,843 | 108.61 |
| | A2 | 63 | 280 | 6,280 | 99.68 |
| | A3 | 63 | 312 | 6,459 | 102.52 |
| | | 189 | 957 | 19,582 | 103.60 |
| Total | | 378 | 1,914 | 38,624 | 102.17 |
| #Doc. | | | 3,136 | 49,003 | 778.92 |

Table 2: Dataset statistics of the human-authored summaries. Left to right, the columns show language variety (Bokmål/Nynorsk), total number of summaries, documents, sentences, and tokens, and finally average token length of summaries. The bottom row shows the corresponding numbers for the original news articles.

the case discussed in the news.

The human-authored summaries exhibit differences in style and news interpretation. Some summaries are more concise, presenting only essential facts (*Summary 1*), while others have a more narrative style (*Summary 2* and *Summary 3*) providing more contextual details. Furthermore, the summaries emphasise on varying aspects, with some focusing on key events (*Summary 1* and *Summary 2*), while other highlight implications or underlying causes (*Summary 3*). We believe that this variation make our summarisation benchmark dataset more representative, and enables model evaluation on a diverse set of summaries.

Dataset statistics As previously mentioned, our dataset uses the dev and test splits of the EDEN dataset (Touileb et al., 2024) comprising documents written in Norwegian BM. Given the limited number of documents in each split (30 in dev and 33 in test) we present the dataset statistics as a whole, for the entire summarisation benchmark, disregarding original splits. This decision aligns with the intended usage of the dataset as a comprehensive benchmark, where treating these splits separately is not meaningful.

Table 2 shows the main statistics of our summarisation benchmark datasets, in terms of number of summaries, sentences, tokens, and average number of tokens, broken down by annotator (A1, A2, and A3) and variety (BM or NN). We also provide the total number of sentences, tokens, and average number of tokens in the original news ar-

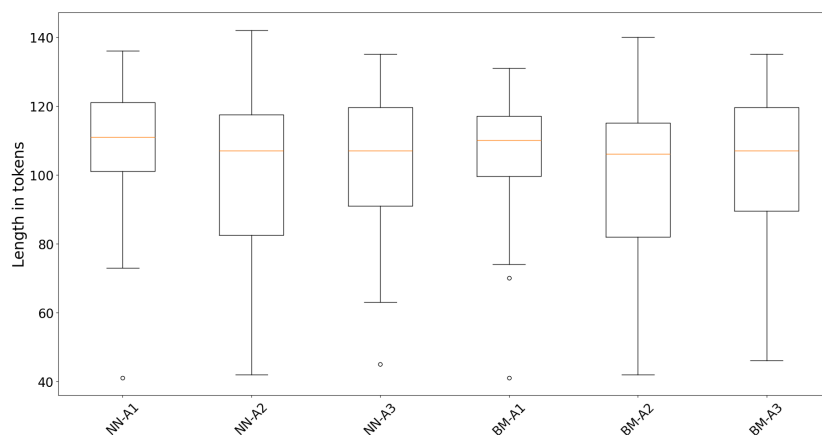


Figure 1: Box plots of summary lengths authored by three different annotators (referred to as A1, A2, and A3) in either Bokmål (BM) or Nynorsk (NN).

ticles for comparison. As can be seen, the total number of summaries and sentences is equal across annotators and Norwegian variety, as all annotators created summaries (and their translations) of every news article from EDEN dev and test splits. However, the number of tokens and the average number of tokens per summary varies between the language varieties and the human annotators. The first annotator (A1 in the table) has authored longer summaries than the other annotators, with annotator 2 creating the shortest ones.

These observations are clearer in the box plot in Figure 1, where A1, A2, and A3 refer to our three human annotators, and BM and NN are the two Norwegian varieties. The figure presents the distribution of summary (token-) lengths across the three annotators, and across the BM and NN varieties. Each annotator’s summaries exhibit a range of lengths, allowing us to observe both individual tendencies and variations. The longest summary was written by annotator 1, while the shortest was written by annotator 2. The median lengths across all summaries are relatively similar, with lengths around 100-token. This we believe suggests a level of consistency in summary length, that also aligns with the guidelines given to the annotators.

There are also clear differences in term of ranges. For instance, NN-A2 has a broader range in summary lengths compared to the others, which might indicate variance in the level of details provided in the summaries. In contrast, both NN-A1 and NB-A1 display narrower ranges, implying that these summaries are more uniform in length with fewer cases of extreme variations.

The whiskers also vary in length, with NB-A3

exhibiting particularly long whiskers. This suggest a broader range of token counts in the summaries, potentially reflecting a less standardised approach to summarisation. Outliers are observed in NN-A2, NN-A2, and NB-A1, and which indicate the presence of significantly shorter summaries than the main distribution. These outliers might represent instances of summaries that are either very condensed, lacking details or depth, or simply based on shorter original news articles.

Overall, the differences between the summaries are subtle, but still noteworthy. Summaries written by annotator 1 appear to have less variability in length, indicating greater consistence in the summarisation style. Annotator 2 seems to have a less strict and rigid way of writing summaries, which might be depending on the original length of the news article. This diversity in summary length and variability makes the datasets more natural. This suggests that models evaluated on this benchmark would need to handle varying levels of details and conciseness that necessitate the ability to meet different summarisation styles effectively.

Annotators’ experience and feedback At the end of the annotation work, we invited annotators to reflect on their main observations and to discuss the specific aspects of the summarisation process, as well as particular news articles that they found most challenging. More concretely, we asked them to reflect on the annotation process, challenges and ambiguities in annotation, consistency in annotation, and adherence to the guidelines. With regards to the annotation process, the annotators had different strategies where for ex-

ample one annotator always started by highlighting named entities, events, facts, and actions to identify the articles’ main points, while another annotator read each article twice to verify accuracy and to avoid excluding details.

Concise, bulletin-like news articles were straightforward to summarise, as their structured formats closely aligned with what they believed would constitute a good summary. The annotators had a clear consensus regarding which articles were relatively straightforward to summarise and which posed greater difficulties. Sports articles and disaster-related news, injuries, or investigations were easier to summarise as they tend to contain clear and concise information.

The annotators noted that increased complexity within certain articles directly correlated with the time required to produce high-quality summaries, highlighting the impact of article complexity on the annotation process. Annotators experienced that presence of subjectivity in the article was a factor indicating increased complexity. This led the annotators to make more choices, increasing the risk of making a misrepresentative summary. Examples of such “difficult” pieces of text are: portrait interviews, feature articles, interviews, opinion pieces, and reviews. Some articles lacked sufficient content, which required external research and made the creation of a summary more tedious. Annotators also particularly struggled with long opinion-based articles, as it was difficult for them to summarise these texts without misrepresenting opinions as facts. The longer and the more complex the article, the more difficult it was for the annotators to reduce the contents to their essence within the maximum summary size.

All annotators reported their focus on journalistic priorities, where the aim was to convey the most relevant facts from the original news articles. While they also report a strict adherence to the guidelines, they still prioritised content accuracy over strict compliance in some cases. With regards to the translation part of the process, the annotators felt that the process was smooth and that it was easy to translate consistently.

4 Evaluation Design

In the following, we illustrate the use of our summarisation dataset as an evaluation benchmark for a range of openly available Norwegian and multilingual LLMs.

Models We evaluate nine pretrained Transformer LLMs as our baselines: NorwAI-Mistral-7B², NORA.LLM (NorBLOOM-7B-scratch³, NorMistral-7B-scratch⁴, and NorMistral-7B-warm⁵; Samuel et al., 2025), NorwAI-Llama2-7B⁶, Viking-7B⁷, Viking-13B⁸, Mistral-7B-v.01⁹ (Jiang et al., 2023), and falcon-7b¹⁰ (Almazrouei et al., 2023). All the LLMs’ weights are taken from the Transformers library (Wolf et al., 2020).

Setup We conduct a zero-shot evaluation of the previously mentioned LLMs using NorEval¹¹, an open-source framework for evaluating Norwegian generative LLMs. We integrate our dataset into NorEval together with 12 diverse prompts written by Norwegian native speakers, who are authors of this paper. Table 3 illustrates the prompts – 6 prompts per language variety. As can be seen, we use a variety of prompting styles to generate summaries, varying both the placement of the source article, as well as the verbosity and precise wording of the instruction. The LLMs’ summaries are generated via the greedy search decoding method.

Performance Metrics We measure the performance using standard summarisation evaluation metrics: ROUGE-L (Lin, 2004) and BERTScore (Zhang et al., 2020). Our result aggregation procedure accounts for prompt sensitivity (Voronov et al., 2024; Lu et al., 2024) and includes two steps: (i) for each prompt, we compute the maximum performance scores between the LLM’s output and each of three human-written references (our human-authored summaries); (ii) we then maximize the BERTScore across all prompts and average the resulting ROUGE-L and BERTScore values over all BM/NN examples.

5 Evaluation Results

Table 4 presents the zero-shot evaluation results on concatenated development and test sets. In addition to this evaluation, we conducted a human-

²hf.co/NorwAI/NorwAI-Mistral-7B

³hf.co/norallm/norbloom-7b-scratch

⁴hf.co/norallm/normistral-7b-scratch

⁵hf.co/norallm/normistral-7b-warm

⁶hf.co/NorwAI/NorwAI-Llama2-7B

⁷hf.co/LumiOpen/Viking-7B

⁸hf.co/LumiOpen/Viking-13B

⁹hf.co/mistralai/Mistral-7B-v0.1

¹⁰hf.co/tiiuae/falcon-7b

¹¹github.com/lrgoslo/noreval

| Bokmål (BM) |
|--|
| 1. Skriv en oppsummering av følgende artikkel med kun noen få punkter: {{article}}\nOppsummering: |
| 2. Oppsummer følgende artikkel med noen få setninger: {{article}}\nOppsummering: |
| 3. {{article}}\nSkriv en kort og presis oppsummering av teksten over. Språket må være klart og lett å forstå. Sørg for å ikke introdusere feil. Oppsummeringen må dekke følgende spørsmål: hvem, hva, hvor, når, og hvorfor er denne saken viktig å vite om. Oppsummeringen må være engasjerende og fremheve nøkkelinformasjon fra artikkelen. Oppsummeringen skal inneholde maksimalt 700 tegn, inkludert mellomrom. |
| 4. Gi et kortfattet sammendrag av følgende tekst: {{article}}\n |
| 5. Lag en kort oppsummering som sammenfatter den følgende teksten i noen få punkter:\n{{article}}\n\nOppsummering: |
| 6. Hele artikkelen:\n{{article}}\n\nHovedpunkt: |
| Nynorsk (NN) |
| 1. Skriv ei oppsummering av følgande artikkel med berre nokre få punkt: {{article}}\nOppsummering: |
| 2. Oppsummer følgande artikkel med nokre få setningar: {{article}}\nOppsummering: |
| 3. {{article}}\nSkriv ein kort og presis oppsummering av teksten over. Språket må vere klart og lett å forstå. Sørg for å ikkje introdusere feil. Oppsummeringa må dekkje følgande spørsmål: kven, kva, kor, når, og kvifor er denne saka viktig å vite om. Oppsummeringa må vere engasjerande og framheve nøkkelinformasjon frå artikkelen. Oppsummeringa skal innehalde maksimalt 700 tegn, inkludert mellomrom. |
| 4. Gje eit kortfatta samandrag av følgande tekst: {{article}}\n |
| 5. Lag ein kort oppsummering som samanfattar den følgande teksten i nokre få punkt:\n{{article}}\n\nOppsummering: |
| 6. Hele artikkelen:\n{{article}}\n\nHovedpunkter: |
| English translation |
| 1. Write a summary of the following article in just a few points: {{article}}\nSummary: |
| 2. Summarise the following article in a few sentences: {{article}}\nSummary: |
| 3. {{article}}\nWrite a short and precise summary of the text above. The language must be clear and easy to understand. Ensure not to introduce errors. The summary must cover the following questions: who, what, where, when, and why this matter is important to know about. The summary must be engaging and highlight key information from the article. The summary should contain a maximum of 700 characters, including spaces. |
| 4. Provide a concise summary of the following text: {{article}}\n |
| 5. Create a short summary that encapsulates the following text in a few points:\n{{article}}\n\nSummary: |
| 6. The entire article:\n{{article}}\n\nMain point: |

Table 3: Six prompts in BM and NN from NoREval used in our zero-shot evaluation experiments (§5).

| Model | BM | | NN | | Overall | |
|-----------------------|--------------|--------------|--------------|--------------|--------------|-------------|
| | ROUGE-L | BERTScore | ROUGE-L | BERTScore | ROUGE-L | BERTScore |
| NorwAI-Mistral-7B | 12.14 | 50.06 | 10.62 | 50.78 | 11.38 | 50.42 |
| NorwAI-Llama2-7B | 13.58 | 54.44 | 12.24 | 54.04 | 12.91 | 54.24 |
| norbloom-7b-scratch | 20.00 | 52.40 | 13.29 | 49.16 | 16.6 | 50.78 |
| normistral-7b-scratch | 25.32 | 58.25 | 15.28 | 48.32 | 20.3 | 53.28 |
| normistral-7b-warm | 17.38 | 49.86 | 9.93 | 41.86 | 13.6 | 45.86 |
| Viking-7B | <u>30.56</u> | <u>69.65</u> | <u>25.82</u> | 70.34 | <u>28.19</u> | <u>70.0</u> |
| Viking-13B | 33.76 | 70.90 | 27.38 | <u>69.96</u> | 30.57 | 70.4 |
| Mistral-7B-v0.1 | 9.60 | 52.36 | 8.70 | 47.28 | 9.15 | 49.82 |
| falcon-7b | 10.61 | 44.40 | 9.80 | 44.06 | 10.2 | 44.23 |

Table 4: Zero-shot evaluation results on concatenated development and test sets by BM and NN. The best score is in bold, second-best is underlined. The LMs with more limited abilities in Norwegian are separated by a dashed line.

based evaluation (see §6) to analyse the LLMs’ behaviour in more detail given the limitations of the automatic performance metrics (Gehrmann et al., 2023; Colombo et al., 2023).

Overall Results We find that all LLMs achieve acceptable performance on both BM and NN. Viking-7B and Viking-13B perform the best, reaching the ROUGE-L of up to 33.76 and BERTScore of up to 70.34. The larger version is insignificantly better than the smaller one. We also observe that Norwegian monolingual LLMs (NorwAI-Mistral-7B, NorwAI-Mistral-7B-pretrain, and NorMistral-7B-warm) can perform on par with LLMs with more limited abilities in

Norwegian (Mistral-7B-v0.1 and Falcon-7b).

Comparison of BM & NN Comparing the results between BM and NN, we find that most LLMs performs better on BM in terms of ROUGE-L (e.g., the δ -score ranges from 1 to 10 for NorwAI-Mistral-7B-pretrain and NorBLOOM-7B-scratch, respectively). However, the BERTScore difference is less pronounced.

The relatively low performance scores suggest that our summarisation dataset presents a challenging benchmark. One could argue that using more advanced, proprietary LLMs, which have demonstrated higher effectiveness in summarisation tasks, could yield better results than the mod-



Figure 2: Screenshot of the interface used during human evaluation. We present a news article on top, and two suggestions for summaries. The goal for the evaluator is to choose the summary they prefer based on simple criteria (see §6).

els we have evaluated here. However, we chose to rely exclusively on open-source models with Norwegian language support to ensure accessibility and reproducibility for future research.

6 Human evaluation

In addition to model and metric-based evaluations, we conducted a manual evaluation. For this purpose, a research assistant was hired to develop an interface where evaluators were shown a news article, followed by two summaries beneath it. An example of this simple interface is shown in Figure 2. The volunteer evaluators were asked to choose their preferred summary from a selection of two summaries: one human-authored and one generated by a model. However, the evaluators were not aware of the provenance of each summary.

To ensure that evaluators rank summaries consistently, we provided them with a set of very simple criteria inspired by evaluations presented in (Fabbri et al., 2021):

- **Relevance:** Selection of essential content from the original news article.
- **Consistency:** Alignment between the summary and the source article, ensuring that the summary contains only factual statements that can be directly inferred from the source.
- **Fluency:** Quality of individual sentences, with particular attention to grammatical correctness to ensure readability.

We also asked the evaluators to prioritise these criteria in the following order: relevance > con-

sistency > fluency, with relevance being the most important and fluency the least. This approach was designed to assess the quality of the summaries based on the primary functions of summarisation: accurately and concisely conveying essential content. The prioritisation we chose reflects a deliberate emphasis on accuracy and factuality over style.

The link to this evaluation interface was shared with volunteer colleagues, resulting in a total of 146 responses. In 138 cases, evaluators preferred the human-authored summaries, while only 8 responses favoured a machine-generated summary. These preferred machine-generated summaries were produced by the three models Viking-13B (4 of the preferred summaries), NorBLOOM-7b-scratch (2 of the preferred summaries), and NorMistral-7b-warm (2 of the preferred summaries), using prompt nr. 1 (BM) and prompt nr. 2 (BM) in Table 3.

Similarly to the results in Table 4, the best model metric-wise, Viking-13B, seem to also be the model most favoured by human evaluators. Although this preference remains limited compared to the preference of human-authored summaries, it provides an indication of the quality of summaries generated by this model compared to the others.

Several issues were identified during the human evaluation of summaries. These were primarily related to those generated by the models. We give a summary of the types of errors that commonly appeared in what follows.

Issues related to relevance the generated summaries often reproduce the initial part of the original article, not including important information presented later, and sometimes even cutting off mid-sentence. Some summaries were direct copy-paste of the original article, or were too lengthy, and occasionally repeating (parts of) the prompts (e.g. “Skriv en oppsummering av følgende artikkel med kun noen få punkter: Tilbake til hverdagen | Helse. Vandrehall [...]”, eng: *Write a summary of the following article in just a few points: Back to Everyday Life | Health. Walking hall [...]*). Some other summaries were too short, providing incomplete contexts or unnatural-sounding sentences.

Issues related to consistency generally, evaluators reported that the summaries were consistent with the source material. However, some summaries did exhibit repetitions of phrases. Minor

but significant alterations in the texts, like adding or omitting words, were also observed. In some instances, the model-generated summaries invented quotes (e.g. a citation in the summary that did not occur in the original news text “- Jeg er veldig glad for at jeg har fått et nytt hjerte, sier Per Arne Olsen til Tønsbergs Blad.” (eng: ‘- I am very happy that I have received a new heart, says Per Arne Olsen to Tønsbergs Blad.’)). However, a simple internet search led us to finding a similar quote in another news article which seemingly the model had access to during training, or confused entities (e.g., mixing between Bill and Hillary Clinton when mentioned jointly in a news article).

Issues related to fluency similarly to what we already have mentioned, despite fluency being largely maintained, certain summaries repeated identical or similar sentences continuously (more than 10 times). Additionally, in some cases we observed missing function words (e.g. the function word “av” (eng: *by*) in the sentence “Malis statsminister Cheick Modibo Diarra har gått av etter å ha blitt pågrepet soldater” (eng: *Mali’s Prime Minister Cheick Modibo Diarra has resigned after being arrested soldiers*) not being included in the same sentence in the generated summary.)

7 Conclusion and Outlook

This paper introduces a novel dataset of human-authored summaries of Norwegian news articles for benchmarking abstractive summarisation. Our dataset is of high quality and provides for each news article a set of diverse summaries written in both Norwegian varieties Bokmål and Nynorsk. Through comprehensive evaluations using human evaluators and generative models, we have demonstrated the robustness and complexity of this benchmark.

As this is the first freely available human-authored Norwegian summarisation datasets, we believe that the impact it will have on benchmarking current and future LLMs is considerable. Looking ahead, we see several avenues for developing models that leverage the particularities of this dataset to build more robust summarisation techniques. This dataset allows us to compare the output of generative models to a distinct set of human-authored summaries, which will allow us to generate more naturally-sounding summaries.

Acknowledgments

We would like to thank our annotators Marie I. Kroka, Frida Måseidvåg, and Lidvard Sandven for their great work on producing the human summaries and their translations. This work was supported by industry partners and the Research Council of Norway with funding to MediaFutures: Research Centre for Responsible Media Technology and Innovation, through the centers for Research-based Innovation scheme, project number 309339.

References

- Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Merouane Debbah, Etienne Goffinet, Daniel Hestlow, Julien Launay, Quentin Malartic, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. Falcon-40B: an open large language model with state-of-the-art performance.
- Pierre Colombo, Maxime Peyrard, Nathan Noiry, Robert West, and Pablo Piantanida. 2023. [The glass ceiling of automatic evaluation in natural language generation](#). In *Findings of the Association for Computational Linguistics: IJCNLP-AACL 2023 (Findings)*, pages 178–183, Nusa Dua, Bali. Association for Computational Linguistics.
- Chenhe Dong, Yinghui Li, Haifan Gong, Miaoxin Chen, Junxin Li, Ying Shen, and Min Yang. 2022. A survey of natural language generation. *ACM Computing Surveys*, 55(8):1–38.
- 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.
- Alexander Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. 2019. [Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy. Association for Computational Linguistics.
- Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. [SummEval: Re-evaluating summarization evaluation](#). *Transactions of the Association for Computational Linguistics*, 9:391–409.
- 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.

- Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Su-leyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, L  lio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timoth  e Lacroix, and William El Sayed. 2023. [Mistral 7B](#).
- Fredrik J  rgensen, Tobias Aasmoe, Anne-Stine Ruud Husev  g, Lilja   vrelid, and Erik Velldal. 2019. Norne: Annotating named entities for norwegian. *arXiv preprint arXiv:1911.12146*.
- 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.
- Rafael Dueire Lins, Hilario Oliveira, Luciano Cabral, Jamilson Batista, Bruno Tenorio, Rafael Ferreira, Rinaldo Lima, Gabriel de Fran  a Pereira e Silva, and Steven J Simske. 2019. The cnn-corpus: A large textual corpus for single-document extractive summarization. In *Proceedings of the ACM Symposium on Document Engineering 2019*, pages 1–10.
- Peng Liu, Lemei Zhang, Terje Farup, Even W. Lauvrak, Jon Espen Ingvaldsen, Simen Eide, Jon Atle Gulla, and Zhirong Yang. 2024. [Nlebench+norglm: A comprehensive empirical analysis and benchmark dataset for generative language models in norwegian](#).
- Sheng Lu, Hendrik Schuff, and Iryna Gurevych. 2024. [How are prompts different in terms of sensitivity?](#) 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 5833–5856, Mexico City, Mexico. Association for Computational Linguistics.
- Petter M  hlum, Dag Trygve Truslew Haug, Tollef Emil J  rgensen, Andre K  sen, Anders N  klestad, Egil R  nningstad, Per Erik Solberg, Erik Velldal, and Lilja   vrelid. 2022. Narc–norwegian anaphora resolution corpus. In *International Conference on Computational Linguistics (ICCL)(COLING)*, volume 29, pages 48–60.
- Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. [Don’t give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.
- Lilja   vrelid and Petter Hohle. 2016. Universal dependencies for norwegian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 1579–1585.
- Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. [A neural attention model for abstractive sentence summarization](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.
- David Samuel, Vladislav Mikhailov, Erik Velldal, Lilja   vrelid, Lucas Georges Gabriel Charpentier, and Andrey Kutuzov. 2025. Small Languages, Big Models: A Study of Continual Training on Languages of Norway. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, Tallinn, Estonia.
- Per Erik Solberg, Arne Skj  rholt, Lilja   vrelid, Kristin Hagen, and Janne Bondi Johannessen. 2014. The norwegian dependency treebank.
- Samia Touileb, Jeanett Murstad, Petter M  hlum, Lubos Steskal, Lilja Charlotte Storset, Huiling You, and Lilja   vrelid. 2024. [EDEN: A dataset for event detection in Norwegian news](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5495–5506, Torino, Italia. ELRA and ICCL.
- Anton Voronov, Lena Wolf, and Max Ryabinin. 2024. Mind Your Format: Towards Consistent Evaluation of In-context Learning Improvements. *arXiv preprint arXiv:2401.06766*.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pi  ric 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.
- 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 (ICLR)*.

Efficient Elicitation of Fictitious Nursing Notes from Volunteer Healthcare Professionals

Jesper Vaaben Bornerup

IT University of Copenhagen
jesper.bornerup@live.dk

Christian Hardmeier

IT University of Copenhagen
chrha@itu.dk

Abstract

Reliable automatic solutions to extract structured information from free-text nursing notes could bring important efficiency gains in healthcare, but their development is hampered by the sensitivity and limited availability of example data. We describe a method for eliciting fictitious nursing documentation and associated structured documentation from volunteers and a resulting dataset of 397 Danish notes collected and annotated through a custom web application from 98 participating nurses. After some manual refinement, we obtained a high-quality dataset containing nurse notes with relevant entities identified. We describe the implementation and limitations of our approach as well as initial experiments in a named entity tagging setup.

1 Introduction

With the emergence of Electronic Health Records (EHR), the way nurses document their work has changed drastically. Printed schemas and handwritten notes were supplanted by computer-based systems like the Danish Sundhedsplatformen (SP), aiming to reduce data redundancy and errors (Ambinder, 2005). To simplify automatic processing and data reuse, EHR systems emphasize structured documentation. This choice has been described as “Technological somnambulism” (Johnson, 2016) and tends to be at odds with the preferences of the clinical professionals, who value usability and flexibility (Rosenbloom et al., 2011) and experience structured documentation as time-consuming and inefficient (Brinkmann et al., 2020; Baumann et al., 2018), frequently leading to inadequate documentation (Tram, 2017).

Automatic generation of structured documentation from free-text nurse notes would offer an at-

tractive solution to this dilemma. However, the development of such systems across countries and languages is frustrated by the lack of training data due to the stringent privacy constraints surrounding all forms of medical notes (Landolsi et al., 2023). While some relevant datasets are available (Johnson et al., 2016), they are specific to the context in which they were produced and may be of limited use in another location characterised by a different language, different social context or different healthcare procedures.

In this paper, we describe and evaluate a method to elicit fictitious nurse notes from volunteering healthcare professionals based on visual stimuli. The collected notes closely mirror real free-text nursing documentation without suffering from the privacy restrictions of authentic notes. Emphasising a low time commitment for the volunteers, our method enabled us to collect a high-quality dataset of 397 notes from 98 participating nurses. We describe our procedures for eliciting and curating the dataset and annotating it for information extraction as well as initial experiments on automatic extraction of structured data. Our dataset is in Danish, but the procedure would be easily generalisable to other languages.

2 Data collection framework

We collected fictitious examples of nursing notes, together with structured annotations of their content, with two goals in mind: 1. The notes collected should mimic authentic nursing notes as much as possible. 2. The entry threshold for participants should be minimal to make recruitment easier. We used visual stimuli to minimize the influence of the stimuli on the participants’ word choice, and imposed a time limit on the text entry to simulate real-life time pressure.

Figure 1 shows the structure of our web application, whose core parts are the stimulus presentation, note capture and structured annotation. Dif-

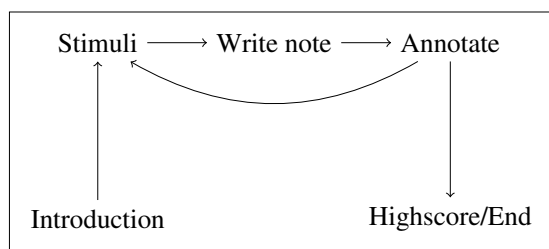


Figure 1: Data Collection Process. After annotating the participant gets the option to repeat or stop.

ferent sets of test participants were used to evaluate the design and offer feedback on the web application during the design process. Some of the test participants were observed doing the process, other were interviewed afterwards.



(a) ©Bangkok Click Studio / Adobe Stock
Example notes: "Pt. only slept around 4 hours, despite medication" and "Pt. is awake and restless"



(b) ©Andrius Gruzdaitis / Adobe Stock
Example notes: "Pt. feeling better and is ready to get discharged later today" and "Pt. happy with the plan and will contact the department in case of worsening in symptoms"

Figure 2: Stimuli examples

As we considered a denser and more focused dataset more useful than a sparse dataset covering many areas, some of the nurse-relevant problem areas were omitted in the our data collection to increase the number of items per category.

Introduction. The introduction page consists of a 4-step guide, including three small video clips demonstrating the process of seeing a stimulus, writing a note and annotating it.

Initially the introduction included detailed instructions to the participants. However, during testing, most of the test participants did not read the text and quickly pressed "next" to move on to the next step, which led to confusion about the process. To mitigate this, the text was cut significantly and the introduction page was redesigned with three GIF animations demonstrating the process. The Facebook post advertising this study also described that the purpose was to create fictitious free-text nursing documentation.

Stimulus presentation. The stimulus display page features an image or video, a 60-second countdown timer and a button to manually progress. The stimulus is drawn uniformly at random from 23 unique items (16 pictures, 7 videos), each chosen to inspire the participants to write relevant nursing documentation. Examples of stimuli and associated notes are shown in Figure 2.

Note capture. The write note page consists of 6 fields in which the participants can write notes based on the 12 nursing-related problem areas (*sygeplejefaglige problemområder*) defined by Styrelsen for Patientsikkerhed (Danish Patient Safety Authority) (Styrelsen for Patientsikkerhed (SFPS), 2023), which defines minimum requirements for nursing documentation. Given the anticipated limited volume of collected data, certain problem areas, including pain and sexuality, were excluded to ensure a more targeted dataset.

A time limit, randomly selected in 9 steps from 20–135 seconds, was imposed on the participants.

Structured annotation. The structured annotation page, shown in Figure 3, is composed of three sections. On the left, the note intended for annotation is displayed for the participant. The right section presents the completed annotations, while the central area houses the module responsible for managing the annotation process. The design of this system adopts a similar layered struc-

Tekst: KAD anlagt og kvitterer straks 100ml mørk grumset urin.

Overkategori: Indgift/Udskillelse

Underkategori: Registrer urin

Underunderkategori: Antal ml

Vælg/indtast værdi: 100

Tilføj et til svar

HUSK AT DU KAN TILFØJE FLERE END ET SVAR. TILFØJ KUN SVAR DER PASSER TIL TEKSTEN.

Indsend svar

Husk du kan tilføje flere svar

Dine svar fremgår her:

Kategori: Indgift/Udskillelse

Underkategori: Registrer urin

Underunderkategori: Udseende

Værdi: Uklar

Remove

Figure 3: Annotate page

ture found in the schemes of EHRs, with a categories, subcategories and subsubcategories to narrow down the options for the final selected value. There is a one-to-one relationship between the highest-level categories and the six fields in the note capture section.

Highscore. The highscore page showed the top contributors and gave the participants a choice to end the process or take one more cycle.

3 Collected data

The study was advertised four times in a Facebook group with 30,000 nurses, and three medical wards were visited once each to recruit participants. A total of 98 nurses participated in the study, producing 407 notes and 594 annotations. We expect that this number could be increased by offering economic incentives for participation. Every note and annotations was manually reviewed for quality control.

3.1 Notes

Most participants produced 1 note (n=34), and the average number of notes per person is 3.75. Typical notes are short and concise with an average length of about 8 words per note, focusing on one category per note. 16 out of 407 notes (3.9%) had to be removed, because they had a length of 1 word, because they directly described the stimulus shown or because they were spam.

The length of the notes shows a very slight upward trend as the time limit was increased, but the effect is not very strong (Figure 5). This might be attributed to participants having the option to proceed by clicking “next” at their discretion, before the timer ran out.

Hjem Om Kontakt

Udskillelse Søvn og hvile Ernæring

Funktionsniveau Kommunikation Psykosocial

Næste

45 sekunder tilbage

Figure 4: Note capture page

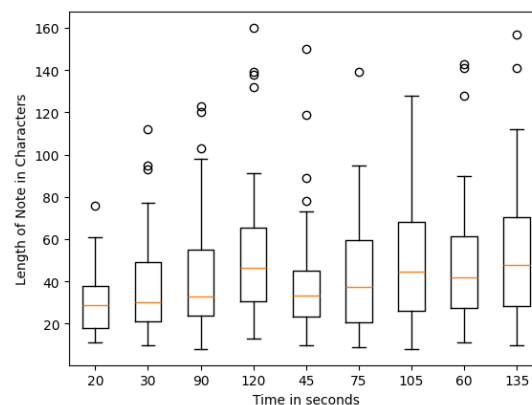


Figure 5: Average Note Length per Timer

3.2 Structured annotations

Each annotation consists of a category, a subcategory, a subsubcategory and a value. Figure 6 shows a note with 4 annotations. The subcategory is not only used to navigate to the right subsubcategory, it also carries information that relates to the final value.

Unannotated. 64 of the notes were submitted without any annotations. 14 were impossible to annotate, as there was no type of annotation which would fit the note, 12 were either 1 character long or cut short, probably because of the time limit, and 38 were possible to annotate.

Annotated. The remaining 343 notes had annotations. The annotations can be divided into 4 groups, all represented in Figure 6.

1. **Exact match:** The selected value in the annotation has an exact match in the note.

| Annotation Type | Count | Percentage |
|----------------------|-------|------------|
| Total annotations | 594 | 100.0% |
| Exact match | 297 | 50.0% |
| Partial match | 106 | 17.8% |
| Interpretation | 78 | 13.2% |
| Incorrect irrelevant | 39 | 6.5% |
| Incorrect relevant | 74 | 12.5% |

Table 1: Annotation Statistics

2. **Partial match:** The selected value in the annotation has partial overlap with the note. This could happen because of two reasons.

- (a) The choices offered by the annotation process forced the use of another word, than was in the note. The structured part enforces the use of the Bristol Stool Scale (Lewis and Heaton, 1997) (which defines consistencies of stools) where “type 4” amounts to “soft”.
- (b) The entity in the note was misspelled or in plural form, causing a mismatch with the structured category.

3. **Interpretation/classification:** The selected value can be interpreted by the note. In Figure 6, the amount of persons is not mentioned, however operating a ceiling hoist requires two people, making the annotation correct.

4. **Incorrect:** The annotation fits in none of the above categories. These annotations can be divided into two categories:

- (a) Relevant, where the annotation fits the theme, but is not present in the note. In Figure 6 the size of the stool is annotated, but is not present in the note.
- (b) Irrelevant, where the annotation is completely unrelated.

Missing Annotations. Missing annotations occur when an *Exact match* or *Partial match* annotation is possible, but missing. Omitted possible *Interpretation* annotations are not considered missing due to the subjectivity of this category. A total of 107 annotations were missing. The distribution among the types of annotation can be seen in Table 1.

A total of 64 different *subcategory/subsubcategory* pairs were used by the

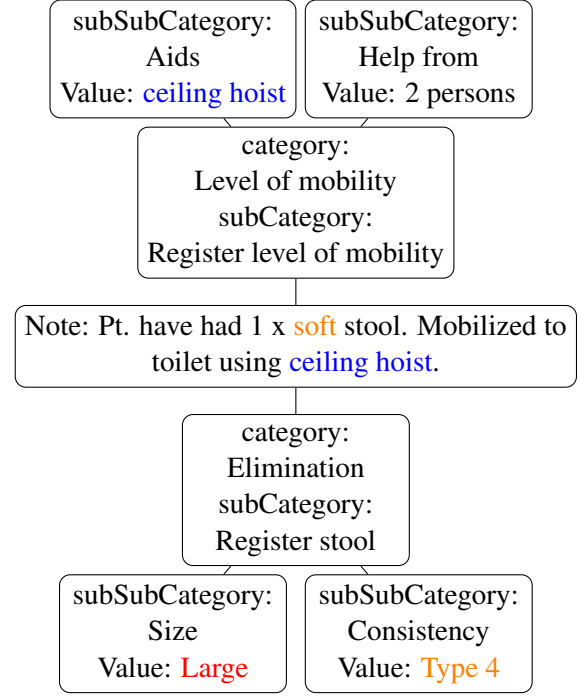


Figure 6: Top annotations: Left Exact match, right Interpretation

Bottom annotations: Left Incorrect, right Partial match .

participants, with the 5 highest having from 22 to 55 entries and the lowest 5 having one entry each.

3.3 Data evaluation

Four people replied to the Facebook post advertising the study that they did not understand the task, and another wrote the interface was too confusing. No other feedback from participants was received.

3.3.1 Notes

A manual review of the notes showed good variety in word choice (e.g., ‘murky’ and ‘unclear’ used interchangeably) and a realistic feel, suggesting they could have been real nurse documentation. The goal was to balance stimuli uniformly across the 6 main categories, but the resulting dataset is not balanced (Figure 2). This could be because some stimuli were harder to understand and therefore harder to write a note to or because some stimuli could be interpreted in multiple ways. For example, a picture of a diaper could both represent *elimination* and *mobility*.

3.3.2 Annotations

The structured annotation part posed a greater challenge for the participants, resulting in 64 unannotated notes (18.6%). However, 12 of those

| Category | Count | Percentage |
|--------------------------|-------|------------|
| Elimination | 145 | 24.4% |
| Mobility | 133 | 22.4% |
| Psychological and social | 83 | 14.0% |
| Sleep and rest | 81 | 13.6% |
| Communication | 78 | 13.1% |
| Nutrition | 74 | 12.5% |

Table 2: Category distribution

were errors or probably cut short because of the time limit, which can be expected. 14 were impossible to annotate with the options given to the participants. This leaves 38 (11%) of the notes which were possible to annotate, but had no annotations.

Incorrect annotations amount to 19% of all annotations, with 66% of them relevant to the topic and the rest completely irrelevant. These were removed from the dataset.

Missing annotations also pose a significant problem. Missing annotations and unannotated notes may be due the interface of the annotation process. While the interface mimics a real EHR, it is not exactly the same. They may also reflect the restrictions of structured documentation: It is time consuming, and finding the right category can be difficult (Brinkmann et al., 2020; Baumann et al., 2018). With no tangible incentive to spend time on it, participants may just click next and move on if they cannot find the right category immediately.

Users had the ability to add their own entity, if it was not among the options provided by the web application. This was however not utilized and that could be the reason for some of the missing annotations.

64 distinct subCategory/subSubCategory pairs were utilized by participants, with the majority being used less than 8 times. This posed a significant challenge for the experimental part of our study (extracting structured information from free-text nurse documentation). To simplify the problem, the classification part of the annotations was discarded as they represented a very small part of the annotation. The remaining annotations were either an exact or partial match, enabling us to reframe the task as a Named Entity Recognition (NER) challenge. Here, the subSubCategory represents the *entity type*, while the value represents an *instance* of the entity type.

4 Entity tagging

Exact matches only needed the start and end positions of the instance to make a complete tag, which was done automatically using regular expressions. Tags for the partial matches were done manually as the value in the original annotation did not match the instance in the note exactly.

Some annotations were straightforward, while others required additional work. For example, participants could choose the color "yellow" for urine. However, since the relation to urine was conveyed in the subcategory, this relation was lost. To address this, additional entity types were created. For example the entity type "OUT" (as something leaving the body), was created for words like "urine" and "stool". The resulting tagset was designed to ensure that, if all entities were accurately identified and appropriately combined, the original structured annotation could be reconstructed. After settling on a tagset the process of tagging all notes began.

One person tagged the dataset, using approximately 20 hours. Every note was looked at four times. Beyond the notes that already had an annotations, every non-annotated note were tagged as well. A total of 23 entity types were used (Table 3).

5 Experiments

Extracting entities from the dataset could prove to be difficult. Some verbs, like "walk", belongs to different categories based on the tense of the word and the surrounding words. The word "big" ("store" in Danish) is used both as a description of an AMOUNT "*The patient consumed two big portions of food*" or as a MODIFIER "*The patient have big problems eating*" (directly translated from Danish). Additionally, some entity types appear much less frequently than others, resulting in an unbalanced dataset where entities occur between 13 and 201 times. Lastly words like "nasogastric tube" (nasalsonde in Danish) and "Foley cathether" (KAD in Danish) are not common words and very specific to the medical domain, which might affect the results in a negative way.

5.1 Data split

Due to the size of the dataset, we used k-fold cross-validation for the evaluation. A value of k=6 was chosen, ensuring each entity type appears at

| Tag | Description |
|------------------|---|
| PSYCHOLOGICAL | A psychological symptom (e.g. <i>sad, happy, angry, frustrated, confused</i>) |
| PHYSIOLOGICAL | A physiological symptom or condition (e.g. <i>constipated, nauseous, bound to bed</i>) |
| STATE | A state a patient can be in (e.g. <i>sleeping, sleepy, relaxed, awake</i>) |
| ASSISTIVE DEVICE | Items such as <i>walker, lift, hearing aids, diaper</i> |
| QUANTITY | A quantity defined numerically or textually (e.g. <i>4, 600, one, two</i>) |
| AMOUNT | A non-numerical amount (e.g. <i>big, small, large, huge, several</i>) |
| PERSONNEL | Any hospital personnel or outside personnel (e.g. <i>nurse, doctor, porter, ergo-therapist, interpreter, he</i>) |
| PATIENT | Any mention of a patient (e.g. <i>Jack, William, pt, patient, him, her</i>) |
| IN | Anything that goes into a patient (e.g. <i>water, food, tubefood</i>) |
| OUT | Anything that goes out of a patient (e.g. <i>aspiration, stool, urine</i>) |
| CONSISTENCY | The consistency of OUT and IN (e.g. <i>soft, hard, liquid, gratin</i>) |
| UNIT | Units of measurement (e.g. <i>ml, mg, x</i>) |
| COMMUNICATION | Everything related to communication with the patient (e.g. <i>Danish, French, German, deaf, mute, reduced hearing</i>) |
| COLOR | Color of something (e.g. <i>brown, orange, red, green, yellow</i>) |
| APPEARANCE | The appearance of something (e.g. <i>clear, murky, dark</i>) |
| ACCESS | Access on the patient's body (e.g. <i>catheter, feeding tube, nasogastric tube</i>) |
| SOCIAL | Family members and friends (e.g. <i>daughter, son, neighbor, friend</i>) |
| MODIFIER | A word that modifies the meaning of a word (e.g. <i>much, less, very, good</i>) |
| NEGATION | A word that negates another word (e.g. <i>not, no</i>) |
| LOCATION | A location something can be (e.g. <i>bed, chair, toilet, leaf ear</i>) |
| TIME | An indication of time (e.g. <i>night shift, day shift, upon inspection, yesterday, tomorrow, after rounds</i>) |
| ACTION | An event that has happened (e.g. <i>eaten, mobilized, instructed, helped</i>) |
| ACTIVITY | An activity the patient can do or can be done to the patient (e.g. <i>walks, eats, drinks</i>) |

Table 3: List of entities

least twice in every split. The data was stratified based on the entity tags for each note, maintaining roughly equal occurrences of entity tags and notes across splits.

5.2 Models

As the notes are in Danish, the number of models available for testing is limited.

5.2.1 BERTs

Four BERT models and one RoBERTa model will be tested.

- **bert-base-cased** (Devlin et al., 2019): An English BERT model not trained on Danish, tested here for comparison with Nordic language models.
- **danishBERT-uncased** (Certainly, 2023): A Danish BERT model trained on 9.5GB of text.
- **bert-base-swedish-cased** (KB (Kungliga Biblioteket), 2023): A Swedish BERT model trained on 15GB of text. Although Swedish, it has more training data than Danish models and it is cased.
- **nb-bert-base-cased** (Kummervold et al., 2021): A Norwegian model trained on the 48.9GB Norwegian Colossal Corpus, showing strong results for Danish tasks.
- **xlm-roberta-base-cased** (Conneau et al., 2019): A multilingual model based on RoBERTa, trained on 2.5TB of Common Crawl data, outperforming mBERT.

A token classification head was attached on top of the BERT/RoBERTa models, whereafter they were fine-tuned with the AdamW algorithm (Loshchilov and Hutter, 2019).

All models underwent a hyperparameter grid search optimization. The hyperparameters finetuned for included epoch [15, 20, 25, 30, 35, 40, 45], learning rate [$2 \cdot 10^{-5}$, $3 \cdot 10^{-5}$, $5 \cdot 10^{-5}$] and weight decay [0.01, 0.1]. Class weights were used in the loss function to handle the unbalanced classes.

5.2.2 Conditional Random Field

The Conditional Random Field (CRF) model developed for this study is supplied with a range of automatically computable features. These features include:

- Capitalization status of the current word, the preceding word, and the following word (uppercase and title case).
- Numeric status, identifying if the word consists of digits.
- Word2Vec embeddings from a Danish model (Sørensen, 2020), providing semantic representations for each word.

Additionally, the model identifies whether a word is at the beginning or end of a sentence, and it receives the same entity tags as the BERT models receive. The hyperparameters we optimized were c1 and c2 (the ℓ_1 and ℓ_2 regularization coefficients) [0.01, 0.1, 0.5, 1.0] and the maximum number of iterations [50, 75, 100].

5.3 Evaluation strategy

The BERT models and CRF model use the BIO (Beginning, inside, outside) tag scheme and a prediction is only correct if the model predicts all B and I tags associated with an entity. A micro, macro and weighted avg f1 score is calculated for each model.

5.4 Results

Table 4 shows the average performance across all entities on the CRF model and the BERT models. The results for individual entity types and all tested models can be seen in Appendix A, Table 6. Not shown in any of the tables is the bert-based model which achieved a macro f1 score of 0.613.

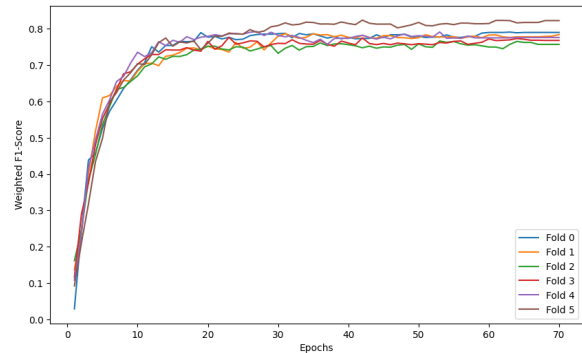


Figure 7: f1 for each epoch, with all 6 folds for DanishBERT

6 Discussion

6.1 Data collection and annotation

The note-writing aspect was successful, with most notes being of high quality and nuanced, indicating the web application’s effectiveness. However, the annotation phase presented challenges, requiring significant effort to address low data quality, a common risk with crowdsourcing (Travis and Burton, 2023).

There are many reasons which could explain why the annotation part was less of a success and unfortunately the only feedback from the participants after the web application launched were a few comments on Facebook. Potential reasons for the troubles with the annotation part could be:

- The participants did not understand the task.
- The participants found the interface provided too difficult to use.
- The inherent problems in structured documentation (time consuming, hard to find the right categories) (Baumann et al., 2018).
- Too much to be expected from volunteers.

Our expectation was that the participants would quickly learn how to fill in the structured annotations, as the interface matched what is used in a real EHR, but the low quality of the annotations and notes without annotations suggested that this part remained difficult to use successfully.

There are several options to mitigate these issues:

- Improve the interface of the annotation process and put it through a more rigorous testing before beginning the data collection. This is time consuming, but could lead to better results.

| | CRF | danishBERT | nb-BERT | xlm-roBERTa-base | swedishBERT |
|--------------|---------------|----------------------|---------------|------------------|---------------|
| micro avg | 0.740 ± 0.033 | 0.779 ± 0.018 | 0.750 ± 0.033 | 0.763 ± 0.037 | 0.725 ± 0.029 |
| macro avg | 0.704 ± 0.044 | 0.744 ± 0.018 | 0.739 ± 0.042 | 0.732 ± 0.030 | 0.699 ± 0.031 |
| weighted avg | 0.726 ± 0.038 | 0.783 ± 0.016 | 0.771 ± 0.032 | 0.772 ± 0.031 | 0.736 ± 0.029 |

Table 4: A comparison between CRF and the BERT models, with average f1 score over a 6-fold-cross validation run and standard deviation between those runs. The best results are bolded.

- Pay nurses and give more detailed instructions. This is expensive, but would provide better quality as the annotators are better instructed.
- Lastly the annotation part of the process could be removed, leaving only the write note part, which could lead to more notes as it is an easier task and thus more encouraging for the participants. However, doing this would lead to more work, as some of the annotations done by the participants were directly usable.

The dataset does not cover all nurse-relevant problem areas, and even the represented nurse-relevant problem areas are incomplete. This limitation poses a challenge in evaluating the results, as there might be nuances of nurse documentation that is harder to capture than others.

Furthermore, the decision to discard annotations based on interpretation in favor of framing the task as a NER task, inadvertently contributes to the incompleteness in capturing the full spectrum of nursing documentation.

6.2 Information extraction

This section will discuss the results in regards to extracting entities from the dataset. When observing the results, one should take into consideration the high variance in the f1 scores between folds. Some folds, as illustrated in Figure 7 had a big difference in f1 score, which both highlights the importance of using a cross-validation strategy, but also indicates that the results might look different if the dataset were larger and more balanced. When looking at the results of this study, these things should be kept in mind.

The best model was the DanishBERT achieving a macro f1 of 0.744. As expected the nb-BERT, which has been shown to have solid performance on danish, showed similar performance with a macro f1 of 0.739 and achieved best performance on 8 entities, compared to the danish

which had the best score on only 4 entities. The xlm-roBERTa-base (multilingual) had a solid performance as well with a macro f1 of 0.732 and best performance on 6 entities. SwedishBERT only managed a macro f1 of 0.699.

The CRF model performed well and performed best of all models in 7 entity types and only having a slightly lower macro f1 of 0.704. However, it did fall short completely on more entities than the BERT models, indicating that the more computational BERT models are more robust in their performance.

7 Conclusion

This study aimed to bridge the gap between structured and free-text documentation in healthcare using NLP techniques. The initial step involved constructing a dataset, which was necessary due to the absence of pre-existing suitable datasets in this domain. Following dataset construction, the study focused on extracting relevant information from nursing documentation within this newly created dataset.

The creation of a synthetic dataset of annotated nurse notes was accomplished through a web application. This application presented various stimuli to participants, prompting them to write corresponding notes. Subsequently, participants annotated their notes using categories reflective of those used in actual EHRs. Overall, the quality of the notes was high, although not all annotations were usable. A manual process was employed to eliminate incorrect annotations and convert the annotations into pairs of (entity type, entity). Additional support entities were manually added, ensuring that every word relevant to nurse documentation was properly tagged.

The process of extracting meaningful information from nurse documentation was approached as a NER task. Performance evaluation revealed that the the Danish, Norwegian and multilingual models had similar performances, with the best being

the Danish which achieved a macro f1 score of 0.744, surpassing the CRF model, which scored 0.704. This performance difference highlights the necessity and efficiency of more advanced models like BERT in handling complex NER tasks.

However, it is important to note that the entity type/entity instance pairs extracted through this NER process do not directly correspond to the structured format which is used in EHRs. This gap underscores a potential area for future research, where the focus could be on transforming these pairs into EHR-compatible triples. Such a transformation is crucial for the practical application of this research in real-world EHR systems, potentially facilitating smoother integration of automated NLP-based documentation tools into healthcare workflows. Nevertheless, this study demonstrates that it is possible to generate synthetic nurse notes and extracting information relevant to nurse documentation from them.

8 Ethical Considerations

Our approach mitigates privacy concerns by using fictitious data, thereby reducing the risk associated with real patient information. However, there is a potential concern regarding the applicability of findings derived from this synthetic dataset, as the data may not accurately reflect real-world.

9 Limitations

With only 98 nurses participating in the study, the dataset is relatively small and only encompass a subset of possible nurse-related categories, potentially limiting its representativeness. Additionally, the lack of multiple reviewers for note quality assessment and the absence of inter-annotator agreement values for the entities diminish the robustness of the results. Lastly it is important to note that all of the participants' status as nurses cannot be verified, as the Facebook group used does not authenticate group members credentials.

References

- Edward P. Ambinder. 2005. Electronic health records. *J Oncol Pract*, 1(2):57–63.
- Lisa Ann Baumann, Jannah Baker, and Adam G. Elshaug. 2018. The impact of electronic health record systems on clinical documentation times: A systematic review. *Health Policy*, 122(8):827–836.
- Maj-Britt Brinkmann, Birgitte Brask Skovgaard, and Raymond Kolbæk. 2020. Dokumentation er en væsentlig del af sygeplejen. *Fag & Forskning*, nr. 1:64–69.
- Certainly. 2023. Certainly has trained the most advanced danish bert model to date. <https://certainly.io/blog/danish-bert-model/>. Accessed: 2023-12-12.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.
- 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.
- Alistair Johnson, Tom Pollard, Lu Shen, and et al. 2016. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3:160035.
- Ralph Johnson. 2016. A comprehensive review of an electronic health record system soon to assume market ascendancy: Epic®. *Journal of Healthcare Communications*, 01.
- KB (Kungliga Biblioteket). 2023. Bert-base, swedish, cased. <https://huggingface.co/KB/bert-base-swedish-cased>.
- Per E Kummervold, Javier De la Rosa, Freddy Wetjen, and Svein Arne Brygfeld. 2021. Operationalizing a national digital library: The case for a Norwegian transformer model. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 20–29, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- MY Landolsi, L Hlaoua, and L Ben Romdhane. 2023. Information extraction from electronic medical documents: State of the art and future research directions. *Knowledge and Information Systems*, 65(2):463–516. Epub 2022 Nov 8.
- S. J. Lewis and K. W. Heaton. 1997. Stool form scale as a useful guide to intestinal transit time. *Scandinavian Journal of Gastroenterology*, 32(9):920–924. PMID: 9299672.
- Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *arXiv:1711.05101 [cs.LG]*. Published as a conference paper at ICLR 2019.

- S Trent Rosenbloom, Joshua C Denny, Hua Xu, Nancy Lorenzi, William W Stead, and Kevin B Johnson. 2011. Data from clinical notes: a perspective on the tension between structure and flexible documentation. *Journal of the American Medical Informatics Association*, 18(2):181–186.
- Styrelsen for Patientsikkerhed (SFPS). 2023. Sygeplejefaglig journalføring.
- Nicolai Hartvig Sørensen. 2020. Word2vec-model for danish. <https://korpus.dsl.dk/resources/details/word2vec.html>. Accessed: [2023-12-12].
- Emma Tram. 2017. Sygeplejerskers dokumentationsspraksis. *Sygeplejersken*, 2017(11):26–27.
- Jack Travis and Scot Burton. 2023. Do you trust what the survey says? examining data quality on online crowdsourcing platforms. *Walton College Insights*.

A Tables

| | Models | | | | |
|---------------|-------------------------------|----------------------------|----------------------|----------------------|-------|
| | XLM-BERT | DanishBERT | swedishBERT | nb-BERT | CRF |
| dropout | 0.1 | 0.1 | 0.1 | 0.1 | - |
| architecture | RoBERTaForTokenClassification | BertForTokenClassification | | | - |
| embedding | RoBERTa _{base} | BERT _{base} | BERT _{base} | BERT _{base} | - |
| parameters | | | | | |
| epoch | 35 | 35 | 45 | 45 | - |
| learning_rate | $5 \cdot 10^{-5}$ | $5 \cdot 10^{-5}$ | $5 \cdot 10^{-5}$ | $3 \cdot 10^{-5}$ | - |
| batch_size | 8 | 8 | 8 | 8 | - |
| weight_decay | 0.1 | 0.01 | 0.01 | 0.1 | - |
| c1 | - | - | - | - | 0.01 |
| c2 | - | - | - | - | 0.01 |
| max_iter | - | - | - | - | 50 |
| algorithm | - | - | - | - | lbfgs |

Table 5: Training parameters

| Entity type | Models | | | | average support |
|------------------|----------------------|----------------------|----------------------|----------------------|-----------------|
| | CRF | DanishBERT | nb-BERT | xlm-roBERTa | |
| PATIENT | 0.948 ± 0.028 | 0.963 ± 0.035 | 0.942 ± 0.021 | 0.926 ± 0.072 | 33.5 |
| PSYCHOLOGICAL | 0.579 ± 0.034 | 0.782 ± 0.063 | 0.805 ± 0.058 | 0.789 ± 0.098 | 17.0 |
| ASSISTIVE DEVICE | 0.777 ± 0.064 | 0.814 ± 0.050 | 0.821 ± 0.051 | 0.836 ± 0.081 | 16.8 |
| QUANTITY | 0.921 ± 0.104 | 0.908 ± 0.022 | 0.931 ± 0.059 | 0.955 ± 0.031 | 16.8 |
| ACTION | 0.764 ± 0.058 | 0.713 ± 0.081 | 0.675 ± 0.084 | 0.642 ± 0.155 | 14.5 |
| PHYSIOLOGICAL | 0.368 ± 0.151 | 0.551 ± 0.045 | 0.615 ± 0.080 | 0.511 ± 0.135 | 14.2 |
| TIME | 0.502 ± 0.129 | 0.604 ± 0.147 | 0.608 ± 0.101 | 0.602 ± 0.085 | 11.3 |
| UNIT | 0.968 ± 0.044 | 0.941 ± 0.054 | 0.885 ± 0.109 | 0.926 ± 0.085 | 11.2 |
| OUT | 0.712 ± 0.084 | 0.804 ± 0.041 | 0.863 ± 0.085 | 0.869 ± 0.079 | 11.0 |
| MODIFIER | 0.510 ± 0.214 | 0.688 ± 0.115 | 0.592 ± 0.132 | 0.580 ± 0.148 | 9.8 |
| ACTIVITY | 0.661 ± 0.127 | 0.650 ± 0.125 | 0.643 ± 0.099 | 0.714 ± 0.094 | 9.0 |
| STATE | 0.822 ± 0.111 | 0.903 ± 0.108 | 0.908 ± 0.081 | 0.904 ± 0.132 | 7.5 |
| PERSONNEL | 0.761 ± 0.180 | 0.774 ± 0.178 | 0.852 ± 0.116 | 0.794 ± 0.138 | 7.2 |
| IN | 0.635 ± 0.184 | 0.766 ± 0.109 | 0.013 ± 0.030 | 0.630 ± 0.125 | 6.7 |
| AMOUNT | 0.617 ± 0.114 | 0.702 ± 0.116 | 0.761 ± 0.115 | 0.765 ± 0.095 | 6.3 |
| CONSISTENCY | 0.713 ± 0.111 | 0.707 ± 0.143 | 0.721 ± 0.101 | 0.770 ± 0.111 | 5.5 |
| COMMUNICATION | 0.513 ± 0.287 | 0.588 ± 0.289 | 0.760 ± 0.181 | 0.643 ± 0.312 | 4.8 |
| ASSIS/LOCATION | 0.745 ± 0.203 | 0.843 ± 0.033 | 0.850 ± 0.150 | 0.736 ± 0.187 | 4.0 |
| ACCESS | 0.825 ± 0.108 | 0.806 ± 0.196 | 0.708 ± 0.220 | 0.747 ± 0.221 | 3.5 |
| COLOR | 0.900 ± 0.200 | 0.856 ± 0.245 | 0.883 ± 0.186 | 0.867 ± 0.221 | 3.3 |
| SOCIAL | 0.960 ± 0.080 | 0.867 ± 0.094 | 0.952 ± 0.067 | 0.875 ± 0.191 | 3.2 |
| LOCATION | 0.280 ± 0.232 | 0.000 ± 0.000 | 0.436 ± 0.261 | 0.000 ± 0.000 | 2.8 |
| NEGATION | 0.867 ± 0.163 | 0.778 ± 0.050 | 0.704 ± 0.137 | 0.721 ± 0.134 | 2.8 |
| APPEARANCE | 0.560 ± 0.285 | 0.856 ± 0.151 | 0.800 ± 0.224 | 0.759 ± 0.214 | 2.2 |
| micro avg | 0.740 ± 0.033 | 0.779 ± 0.018 | 0.750 ± 0.033 | 0.763 ± 0.037 | 222.333 |
| macro avg | 0.704 ± 0.044 | 0.744 ± 0.018 | 0.739 ± 0.042 | 0.732 ± 0.030 | 222.333 |
| weighted avg | 0.726 ± 0.038 | 0.783 ± 0.016 | 0.771 ± 0.032 | 0.772 ± 0.031 | 222.333 |

Table 6: A comparison between different models, with average f1 score over a 6-fold-cross validation run and standard deviation between those runs. The best result being bolded. swedishBERT not shown.

| Category | SubCategory | SubSubCategory |
|------------------|---------------------------|---|
| Functional Level | Current functional level | Mobility aids: 37 |
| | | Mobility assistance: 18 |
| | | Assistance with elimination: 3 |
| | | Mobility restrictions: 3 |
| | | Personal hygiene assistance: 2 |
| | Habitual functional level | Habitual mobility: 3 |
| | | Mobility aids: 2 |
| | | Personal hygiene assistance: 2 |
| | Mobilization activity | Mobility aids: 30 |
| | | Mobility assistance: 17 |
| | | Mobilization (number of times): 6 |
| | | Mobilization (where the patient is mobilized to): 6 |
| | | Mobilization (distance) in meters: 3 |
| | | Mobilization (time): 1 |
| Sleep and rest | Habitual sleep | Sleep pattern: 2 |
| | | Sleep disturbances: 6 |
| | Rest | Resting state: 9 |
| | Sleep registration | Hours slept during shift: 18 |
| | | Sleep quality: 8 |
| | | Current state: 8 |
| | Sleep/Rest issues | Problems: 23 |
| | | Measures taken: 7 |

Table 7: Annotations 1/3

| Category | SubCategory | SubSubCategory |
|--------------------------|---------------------------|--------------------------|
| Communication | Barriers | Language: 12 |
| | | Hearing: 10 |
| | | Cognitive: 8 |
| | Communication assistance | Technical aids: 22 |
| | | Need for interpreter: 17 |
| | | Need for relatives: 9 |
| Psychological and social | Psychological | Current mental state: 55 |
| | | Reaction to illness: 11 |
| | | Habitual mental state: 4 |
| | | Illness insight: 4 |
| | | Perception of health: 1 |
| | Social | Network: 8 |
| Elimination | Aspiration | Amount: 7 |
| | | Frequency: 5 |
| | | Color: 3 |
| | Stool registration | Consistency: 16 |
| | | Amount: 15 |
| | | Frequency: 14 |
| | | Color: 12 |
| | | Location: 2 |
| | Stool status registration | Stool status: 8 |
| | Urination registration | Amount in ml: 13 |
| | | Source: 12 |
| | | Appearance: 11 |
| | | Color: 10 |
| | | Amount: 9 |
| | Regular bowel movements | Frequency: 4 |
| | | Consistency: 3 |

Table 8: Annotations 2/3

| Category | SubCategory | SubSubCategory |
|-----------|-----------------------------|-------------------------------|
| Nutrition | Current nutritional status | Weight in kg: 2 |
| | | Height in cm: 1 |
| | Assistance to eat and drink | Assistance to drink: 3 |
| | | Assistance to eat: 2 |
| | Diet | Consistency food: 10 |
| | | Diet: 5 |
| | | Consistency liquids: 4 |
| | Issues | Nausea: 11 |
| | | Appetite: 9 |
| | | Swallowing difficulties: 1 |
| | Meal registration | Percentage of intake: 13 |
| | | Problems: 5 |
| | | Intake via tube as planned: 5 |
| | | Intake via tube in ml: 3 |

Table 9: Annotations 3/3

B Description of stimuli

1. A 20-second video of a man trying to eat food in a kitchen, but ends up pushing it away while frowning.
2. A 20-second video of a man enjoying a sandwich outside.
3. A picture of an elderly woman receiving food through a nasogastric feeding tube.
4. A picture of an elderly woman walking with a walker in a park.
5. A picture of two healthcare professionals using a ceiling hoist to mobilize a man in a hospital bed, with a wheelchair at the end of the bed.
6. A 15-second video of a 100-year old woman running.
7. A picture of a healthcare professional assisting a man using a walker.
8. A picture of two healthcare professionals assisting a man walking with elbow sticks.
9. A picture of a man placing a hearing aid in an ear.
10. A video of a young woman using sign language.
11. A video of an interpreter translating Spanish in a hospital setting.
12. A picture of a man lying in a hospital bed, with another man in non-uniform clothing and a doctor standing besides it.
13. A picture of a happy smiling woman in a hospital gown in a bed.
14. A picture divided in two: To the left a doctor speaking and gesturing with his hands, to the right a man putting his hands pressed against his head and his face and his brow deeply furrowed.

15. A picture divided in two: To the left a doctor speaking and gesturing with his hands, to the right a man with visible tears on his face.
16. A picture of a woman lying in a bed with eyes closed in a dimly lit room.
17. A drawing of a man lying in bed counting sheeps.
18. A 10-second video of a young man walking around restlessly.
19. A drawing of bacteria, with the names of three bacteria known to cause diarrhea.
20. A picture of a diaper.
21. A picture of a person on a toilet.
22. A picture of a urine drainage bag.
23. A 10-second video clip of a woman vomiting in a bag in a restaurant.

Analyzing the Effect of Linguistic Instructions on Paraphrase Generation

Teemu Vahtola¹ Songbo Hu² Mathias Creutz¹
Ivan Vulić² Anna Korhonen² Jörg Tiedemann¹

¹University of Helsinki, Finland

²Language Technology Lab, University of Cambridge, UK

{teemu.vahtola, mathias.creutz, jorg.tiedemann}@helsinki.fi
{sh2091, iv250, alk23}@cam.ac.uk

Abstract

Recent work has demonstrated that large language models can often generate fluent and linguistically correct text, adhering to given instructions. However, to what extent can they execute complex instructions requiring knowledge of fundamental linguistic concepts and elaborate semantic reasoning?

Our study connects an established linguistic theory of paraphrasing with LLM-based practice to analyze which specific types of paraphrases LLMs can accurately produce and where they still struggle. To this end, we investigate a method of analyzing paraphrases generated by LLMs prompted with a comprehensive set of systematic linguistic instructions. We conduct a case study using GPT-4, which has shown strong performance across various language generation tasks, and we believe that other LLMs may face similar challenges in comparable scenarios.

We examine GPT-4 from a linguistic perspective to explore its potential contributions to linguistic research regarding paraphrasing, systematically assessing how accurately the model generates paraphrases that adhere to specified transformation rules. Our results suggest that GPT-4 frequently prioritizes simple lexical or syntactic alternations, often disregarding the transformation guidelines if they overly complicate the primary task.

1 Introduction

Large language models (LLMs) can, without doubt, generate fluent and linguistically correct language with relevance to given prompts (Sottana et al.,

2023). However, to what extent can they follow complex linguistic instructions and execute them in a meaningful way? To this end, we propose a systematic approach for analyzing LLMs in performing explicit, theoretically grounded paraphrase transformations in English, using a validated list of 25 linguistic operations (Bhagat and Hovy, 2013).

It is necessary to have knowledge of fundamental linguistic concepts to follow those specialized instructions. This study provides insight into the capabilities and limitations of LLMs when faced with such a demanding task. Extending our understanding on the connections between linguistically grounded theories of paraphrasing and the practical abilities of LLMs, we hope to improve paraphrasing performance with explicit linguistic operations, with potential applications in text simplification (Nisioi et al., 2017), computer-assisted language learning (Mayhew et al., 2020), machine translation (Callison-Burch et al., 2006; Mehdizadeh Seraj et al., 2015) and automatic summarization (Gupta and Gupta, 2019).

We conduct a case study analyzing paraphrases generated by a representative state-of-the-art LLM, GPT-4 (Achiam et al., 2023), focusing on the abilities of the model to create meaning-preserving and diverse paraphrases using systematic instructions related to the 25 paraphrasing categories of Bhagat and Hovy (2013). Our analysis further looks into the complexity of individual transformations and how GPT-4 copes with them with varying degrees of in-context learning (Brown et al., 2020; Dong et al., 2024). Furthermore, we study how humans perceive the produced paraphrases in terms of semantic similarity and linguistic diversity.

The **contributions** of the paper are the following: (1) Our study connects a descriptive theory of paraphrasing with generative language models and human perception of sentence-level semantic similarity. (2) We conduct a limited case study, in-

| Full Name | Abbreviation |
|--|----------------|
| synonym substitution | synonym |
| antonym substitution | antonym |
| converse substitution | converse |
| change of voice | voice |
| change of person | person |
| pronoun/co-referent substitution | pron./co-ref. |
| repetition/ellipsis | repetition |
| function word variations | func. word |
| actor/action substitution | actor/action |
| verb/’semantic-role noun’ substitution | verb/sem. noun |
| manipulator/device substitution | manip./device |
| general/specific substitution | gen./spec. |
| metaphor substitution | metaphor |
| part/whole substitution | part/whole |
| verb/noun conversion | verb/noun |
| verb/adjective conversion | verb/adj. |
| verb/adverb conversion | verb/adv. |
| noun/adjective conversion | noun/adj. |
| verb-preposition/noun substitution | vp./noun |
| change of tense | tense |
| change of aspect | aspect |
| change of modality | modality |
| semantic implication | sem. impl. |
| approximate numerical equivalences | num. eq. |
| external knowledge | ext. knowl. |

Table 1: This table lists all the paraphrase defining transformations from Bhagat and Hovy (2013), along with their abbreviations as used throughout this paper, particularly in Figure 2.

investigating a systematic approach for analyzing the ability of LLMs to follow complex instructions and how different degrees of complexity influence the result of generated paraphrases. (3) To facilitate further research on controlled paraphrase generation and the variability of human language, we publicly release the set of automatically generated sentence pairs exhibiting diverse transformations, accompanied by their corresponding human annotations, at <https://github.com/Helsinki-NLP/paraphrase-instructions>.

2 Background

Paraphrasing denotes variability in expressed meaning. Vague definitions such as this one are typical ways of framing the concept of paraphrasing in NLP research (Vila et al., 2014). However, previous research in (computational) linguistics has presented various, more fine-grained typologies that outline the linguistic transformations defining paraphrasing.

Through the lens of existing paraphrase theories (Mel’čuk, 2012; Honeck, 1971; Harris, 1957), Bhagat and Hovy (2013) empirically validate paraphrase examples from two corpora: the

Multiple-translation Corpus (Huang et al., 2002) and the Microsoft Research Paraphrase Corpus (MRPC) (Dolan et al., 2004). They outline 25 concrete operations with systematic linguistic instructions of transformations that produce sentences with near-equivalent meaning. The perspective to these operations is mostly lexical, focusing on the specific lexical changes that can be made at the sentence or phrase level to create paraphrases (Bhagat and Hovy, 2013). However, several of the operations trigger changes that would traditionally fall within the domain of syntactic theory. One such operation would be *ellipsis*. We list all the transformations defined in Bhagat and Hovy (2013) in Table 1.

Correctly applying these transformations in automatic paraphrase generation requires the model to process fundamental linguistic concepts and accurately recognize the phrase-level transformations triggered by the defined lexical operations. Furthermore, not every transformation is appropriate for every context. Therefore, the model must thoroughly process the definition and have intricate semantic reasoning abilities to construct sentence pairs that are appropriately suited for the intended transformation. To this end, we analyze the capabilities of LLMs in producing paraphrastic sentence pairs given systematic linguistic instructions. The transformations span from simple local changes, such as synonym substitution (*to build/to construct*) or change of aspect (*studying/studies*), to more complex alterations, such as converse substitution (*buy/sell*).

Along with systematic, descriptive definitions, Bhagat and Hovy (2013) provide 1–3 examples for each paraphrase transformation. Synonym substitution, for example, is defined as follows:¹

Replacing a word/phrase by a synonymous word/phrase, in the appropriate context, results in a paraphrase of the original sentence/phrase. This category covers the special case of genitives, where the clitic ’s is replaced by other genitive indicators like of, of the, and so forth. This category also covers near-synonymy, that is, it allows for changes in evaluation, connotation, and so on, of words or phrases between paraphrases. Example:

1. Google bought YouTube. ↔ Google acquired YouTube.
2. Chris is slim. ↔ Chris is slender. ↔ Chris is skinny.

These definitions followed by a small number of examples can be utilized as such in prompts for

¹For an exhaustive list of the definitions and examples of the paraphrase transformations, we refer the reader to Bhagat and Hovy (2013).

few-shot in-context learning, where an LLM is instructed to generate sentence pairs incorporating the specific transformations. As few-shot learning has been shown to be an effective approach for applying LLMs in various tasks (Brown et al., 2020), we focus on leveraging the framework of Bhagat and Hovy (2013) for evaluating few-shot learning with GPT-4 across a wide range of linguistic operations related to paraphrasing.

In a contemporary work, Meier et al. (2024) analyze various paraphrase types generated by GPT-3.5 by employing more abstract linguistic definitions of paraphrase phenomena as defined by Barrón-Cedeño et al. (2013) and Vila et al. (2014). These phenomena comprise abstract linguistic properties, such as changes based on *morpholexicon*, *structure*, and *semantics*. Each of these classes is further divided into subclasses and types, where one type (e.g., *same-polarity substitution*) can include multiple concrete transformations (e.g., *synonymy*, *general/specific substitution*, or *exact/approximate alternations*) (Barrón-Cedeño et al., 2013). Meier et al. (2024) select 10 of such types for their analysis. Many of the selected types focus on local substitutions, such as *inflectional changes*, *punctuation changes*, and *spelling changes*, while only a few focus on global changes that require intricate contextual understanding. As opposed to this, we use the typology of Bhagat and Hovy (2013), which provides an empirically validated list of concrete linguistic transformations for generating paraphrases, along with their linguistic definitions and examples, covering a wider range of local and contextual transformations. These concrete definitions enable a precise assessment of which specific linguistic features are well-represented by the chosen LLM and which areas the model still lacks sufficient knowledge in.

3 Experimental Details

3.1 Data Generation

We apply GPT-4² (Achiam et al., 2023) via the API to generate potential paraphrase pairs following a comprehensive list of paraphrasing operations (Bhagat and Hovy, 2013). We selected GPT-4 as a representative and powerful LLM after initial experiments with various LLMs suggested that GPT-4 produced the most fluent output, which is essential for accurately analyzing our setting. We

²gpt-4-turbo-2024-04-09 is used.

Template 1: System Prompt

You are a helpful assistant designed to output JSON.

Synonym substitution: Replacing a word/phrase by a synonymous word/phrase, in the appropriate context, results in a paraphrase of the original sentence/phrase. This category covers the special case of genitives, where the clitic 's is replaced by other genitive indicators like of, of the, and so forth. This category also covers near-synonymy that is, it allows for changes in evaluation, connotation, and so on, of words or phrases between paraphrases.

Example:

- (a) Google bought YouTube. \iff Google acquired YouTube.
- (b) Chris is slim. \iff Chris is slender. \iff Chris is skinny.

Template 2: User Prompt for Simple Sentences

Could you give me 10 more examples following the given description? Return the examples as a list of json objects.

Template 3: User Prompt for Complex Sentences

Could you give me 15 more examples following the given description? Generate 5 compound sentences, 5 complex sentences, and 5 compound-complex sentences to showcase a variety of syntactic structures. It is enough to perform the operation in only one of the clauses. Return the examples as a list of json objects.

Figure 1: Prompt templates we use for generating the paraphrases.

use the default values provided by the OpenAI package for all hyper-parameters. Additionally, we configured the response output of the model to JSON mode, following the text generation guidelines recommended by OpenAI.³

³<https://platform.openai.com/docs/guides/text-generation/json-mode>

| Sentence type | Example sentence |
|------------------|--|
| Simple | The company employs 100 workers. |
| Simple | The teacher explained the concept clearly. |
| Complex | Although it was raining, we played football. |
| Compound-complex | She loves running in the morning, and when she returns, she makes breakfast. |
| Compound-complex | She opened a savings account, and she deposited her birthday money, while her parents watched proudly. |

Table 2: A randomly sampled set of five generated sentences along with their corresponding sentence types.

In paraphrase generation, a set of source sentences is typically given, and the task is to generate target sentences with the same meaning. In our experiment, however, we let the model generate both the source and the target sentences given the definition and 1–3 examples. Since not all transformations are possible on just any source sentence, this allows for the model to come up with suitable source/target pairs for each transformation. Moreover, we believe that our approach more effectively encourages the model to engage in deeper semantic reasoning. When provided with a source sentence, the model is already primed towards a certain transformation, potentially making the task simpler. In contrast, when given only a description of a paraphrase operation along with a few examples, the model must first fully identify the relationship between the description and the examples to generate an appropriate source sentence.

We leverage the definitions and examples given in Bhagat and Hovy (2013) as prompts for the LLM, and request it to produce 25 sentence pairs following the definitions of each of the 25 transformations. Our initial experiments suggest that when we only use the definition and the examples as the prompt, the model predominantly generates rather short sentences with simple syntactic structures, which may constrain its ability to execute more complex paraphrasing transformations. Therefore, we explicitly prompt the model to generate *compound*, *complex* and *compound-complex* sentences. Table 2 presents randomly sampled examples of various sentence types.

The prompts are composed of two parts: system prompts and user prompts, as illustrated in Figure 1. For each paraphrase operation described in Bhagat and Hovy (2013), we construct a system prompt following Template 1, adapting the trans-

formation definition and examples as needed. To generate simple sentence pairs, we use Template 2 as the user prompt. For syntactically complex sentence pairs, we employ Template 3. These templates are specifically crafted to guide the model in producing sentence pairs with varying levels of syntactic complexity.

Eventually, we generate 10 simple sentences and 5 each of compound, complex, and compound-complex sentences for every paraphrase transformation.

3.2 Collecting Annotations

We collect manual annotations by four independent annotators to the generated sentences to answer three key questions: **(1)** Does the generated sentence pair follow the given definition of a paraphrase transformation? **(2)** Are the generated sentences paraphrases of each others? **(3)** To what extent are the generated sentences semantically equivalent? Each sentence pair is annotated by all annotators. For evaluating the third question concerning semantic equivalency, we follow previous work involving manually annotating paraphrases (Creutz, 2018; Kanerva et al., 2021), and use the four-point Likert scale with the following scores and associated descriptions: 4: *Full paraphrases*, 3: *Paraphrases in some contexts*, 2: *Semantically similar sentences but not paraphrases*, 1: *Unrelated sentences*.

The annotators are fluent speakers of English, and knowledgeable of fundamental linguistic concepts.⁴ They are provided with the definitions and examples of each paraphrase operation, as well as

⁴In addition to some of the authors, we involve colleagues as annotators, bringing the total number of annotators to four. Each example is annotated by all four annotators to better capture the range of human variability and subjectivity in evaluating paraphrases.

| Annotator | Para. Acc. | Trans. Acc. |
|-----------|------------|-------------|
| 1 | 0.824 | 0.688 |
| 2 | 0.869 | 0.677 |
| 3 | 0.821 | 0.677 |
| 4 | 0.872 | 0.744 |
| Average | 0.847 | 0.696 |

Table 3: Model performance on paraphrase accuracy (Para. Acc.) and transformation accuracy (Trans. Acc.), evaluated by four annotators. Paraphrase Accuracy measures whether the generated sentence pairs qualify as paraphrases. Transformation Accuracy measures whether the sentence pairs adhere to the predefined transformation operation.

the generated sentence pairs. Appendix A shows a screenshot of the customized annotation tool.

4 Results and Discussion

4.1 Paraphrase and Transformation Accuracy

We first focus on evaluating the model’s performance with respect to the aforementioned questions (1) and (2). By *transformation accuracy* we understand the proportion of generated sentence pairs that successfully follow the desired transformation operation (Question 1). By *paraphrase accuracy* we understand the proportion of generated sentence pairs that are true paraphrases (Question 2).

Table 3 presents the obtained paraphrase and transformation accuracies for all the generated sentence pairs, as assessed by our four expert annotators. It can be seen that GPT-4 generally performs well at providing alternative expressions that convey the same meaning (average paraphrase accuracy is 84.7 %). However, it shows clear limitation in accurately following the specified transformations (average transformation accuracy is 69.6 %). Furthermore, the evaluation results indicate that the scores provided by the annotators are consistent and similar. To demonstrate the reliability of our measurement approach, we compute Fleiss’ Kappa for the two binary variables in our dataset: paraphrase accuracy and transformation accuracy. The Fleiss’ Kappa scores were 0.53 for paraphrase accuracy and 0.71 for transformation accuracy. These scores indicate moderate and substantial agreement among annotators, respectively, demonstrating the robustness of our evaluation methodology and the inherent subjectivity in evaluating paraphrases.

Figure 2 presents the paraphrase and transformation accuracies for each individual paraphrase transformation operation, averaged over the different annotators. The figure clearly illustrates that the model achieves high results in paraphrase and transformation accuracies for specific, local changes, such as synonym substitution, antonym substitution, change of voice, and change of aspect. In contrast, the model appears to struggle with transformations that require a more nuanced understanding of context, such as converse substitution, actor/action substitution, or verb/adverb conversion.

Next, we provide an analysis across the various types of paraphrase transformations to better understand where the model succeeds and the kinds of mistakes it makes when it struggles.

4.2 Qualitative Analysis

Figure 3 illustrates the correlation between paraphrase and transformation accuracy. All transformations except one are located either in the top row or the right-most column of Figure 3, meaning that either the transformation or the paraphrasing was performed successfully (accuracy > 75 %). This is an excellent result.

The top right corner represents the most successful transformations, with a high transformation accuracy combined with a high paraphrase accuracy. There are ten such transformations corresponding to 40 % of all 25 types. These are fairly straightforward or local transformations, such as replacing synonyms within sentences (*started* vs. *began*) or substituting a word with its negated antonym (*happy* vs. *not sad*). Approximate numerical equivalence (mapping between units) and external knowledge (the *Louvre* is a museum) are also found here. This outcome is not too surprising given that a number of well-known paraphrase corpora, such as PPDB (Ganitkevitch et al., 2013) and MRPC (Dolan et al., 2004), contain similar examples (cf. Rajana et al., 2017; Bhagat and Hovy, 2013) and the model has most likely been trained on such data. Moreover, knowing that 125 miles corresponds to about 200 kilometers can be memorized from the training data rather than actually being calculated by the model.

There are more transformations in the right-most column (21) than in the top row (13), indicating that the system more accurately generates paraphrases than the desired transformation types.

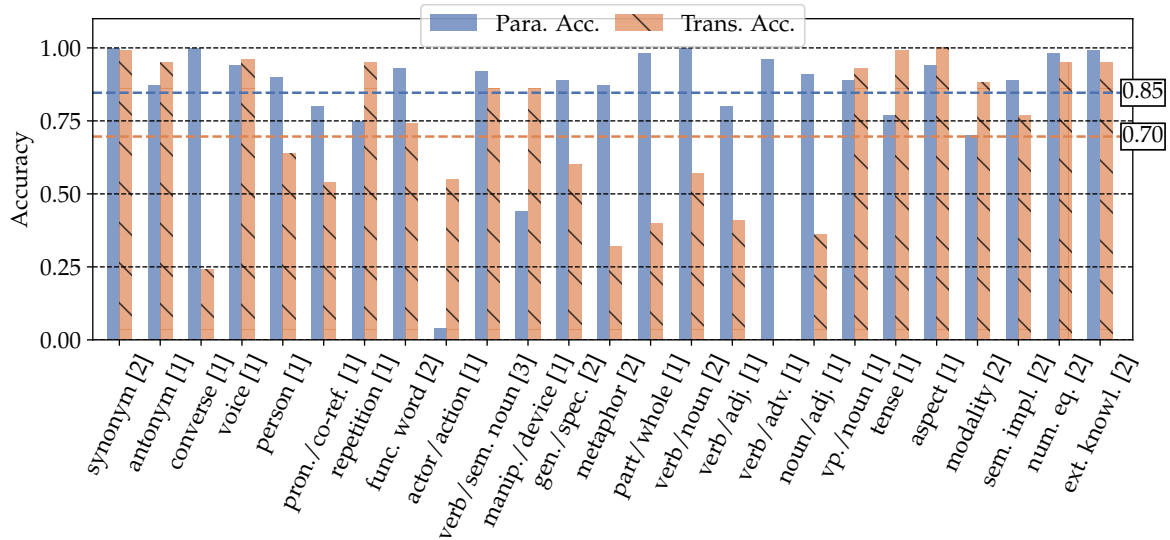


Figure 2: Model performance on Paraphrase Accuracy (Para. Acc.) and Transformation Accuracy (Trans. Acc.). This figure highlights the aggregated mean values for each metric across the 25 transformation operations, indicated by dashed horizontal lines. Abbreviations representing each operation are used, with full names provided in Table 1. All the results are based on annotations by four expert annotators. The number of examples provided by Bhagat and Hovy (2013) for each operation is noted in square brackets. For example, *synonym* [2] indicates two examples for the synonym substitution operation.

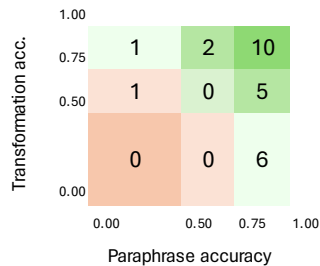


Figure 3: Distribution of the 25 transformations into 3×3 distinct bins depending on paraphrase and transformation accuracy. There are three intervals on the axes, corresponding to accuracies between 0.0 and 0.5, above 0.5 up to 0.75, and between 0.75 and 1.0, respectively.

Failures to capture the desired transformation, while still producing a valid paraphrase, include the following mistakes: (1) using change of voice (*buy/be bought*) instead of converse substitution (*buy/sell*), verb/noun conversion (*to try/make an attempt*) or verb/adjective conversion (*to clean/make clean*), (2) confusion between the categories part/whole (*room/house*) vs. general/specific (*astronomical body/sun*), (3) poor metaphor generation capacity (*“a sea of people”* vs. *“an ocean of people.”*). Apart from the very demanding task of creating metaphors, the failures here are artefacts

of somewhat artificial, grammatical distinctions, such that participle forms of verbs (*interested*) do not qualify as adjectives (*curious*).

Failures to reliably produce paraphrases while still being faithful to the desired transformation (top row, left and center) comprise manipulator/device substitution (*“The photographer (vs. camera) took stunning photos”*) and change of modality (*finds/can find*), which in fact can alter the meaning. Nevertheless these types have been included in the paraphrase taxonomy of Bhagat and Hovy (2013), which may seem odd. While it is possible to produce paraphrases within the limits of the above transformations, it requires strong semantic reasoning abilities from the model. It must first generate a source sentence that is comprised of (potentially limited) concepts that are suitable for such transformations and then create an effective paraphrase as a target sentence.

Additionally, the removal of repetition (ellipsis) is sometimes performed too aggressively and the meaning is not preserved (*“The cat chased the mouse and the dog chased the squirrel.”* vs. *“The cat chased the mouse and the dog did, too.”*). The model may overly prioritize elliptical constructions similar to the example prompt, failing to generalize to different kinds of sentence structures. Specifi-

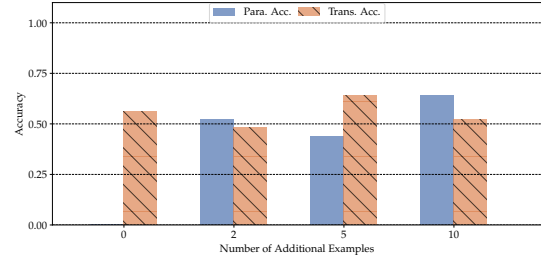
cally in the above example, it fails to recognize that omitting the object in the second clause changes the meaning as the repeated part is the predicate rather than the object.

The poorest result is obtained for actor/action substitution (center left), which mostly generates semantically or grammatically incorrect sentence pairs: “*I love teaching.*” vs. “*I love teacher.*” This operation is particularly challenging, as it demands deep contextual understanding. Merely replacing an actor, such as *teacher* with a corresponding action, such as *teaching*, is not sufficient for preserving the original meaning if the context does not allow it. The example Bhagat and Hovy (2013) provide for actor/action substitution is: “*I dislike rash drivers (vs. driving).*” It is possible that the training data has limited examples of correctly applying this operation, which can result in poor accuracy in recognizing appropriate concepts and contexts.

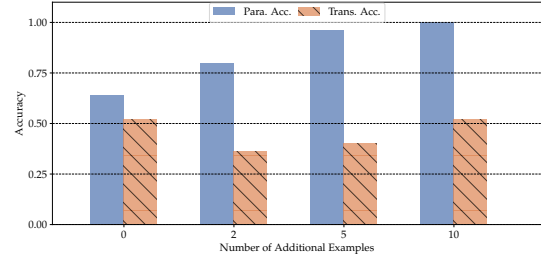
4.3 Semantic Equivalence

Our annotators assessed three criteria (Section 3.2), two of which have been analyzed thoroughly above: transformation accuracy (Question 1) and paraphrase accuracy (Question 2). Question 3 on semantic equivalency remains to be studied. Next, we compare the binary annotations of paraphrase accuracy (Question 2) to the 4-level Likert scale annotations (Question 3). The four-level scale offers a more nuanced view on semantic equivalency than the binary paraphrase classification.

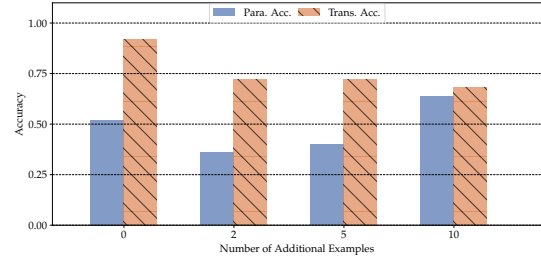
Two out of four annotators had virtually perfect correlation between the binary paraphrase category and Likert scale values 4 and 3 (“full paraphrases” and “paraphrases in some contexts”). The other two annotators did very similarly, but in addition, there was a small number of data points (around 3 % and 6 %) in which Likert scale 3 (“paraphrases in some context”) rendered the “not paraphrases” binary classification. An example where both annotators classified the example as a non-paraphrase but still assigned it a Likert scale score of 3 is: “*The driver (vs. car) accelerated quickly, but the passenger felt nervous.*” Overall, the binary annotations closely align with the detailed results from the 4-level Likert scale. Consequently, we do not conduct further analysis on the relationship between the different annotation granularities but reserve it for future work.



(a) Actor/action Substitution



(b) Verb/adjective Conversion



(c) Manipulator/device Substitution

Figure 4: Model performance for (a) Actor/action Substitution, (b) Verb/adjective Conversion, and (c) Manipulator/device Substitution with increasing number of in-context learning examples. For each example, we append it to the system prompt as shown in Template 1 in Figure 1. Results are based on annotations by one expert annotator.

4.4 Additional In-context Learning

Bhagat and Hovy (2013) do not provide the same number of examples for all of the 25 transformations. In fact, 15 transformations have only 1 example, 9 have 2, and 1 has 3 examples.⁵ As LLMs have been shown to generalize well from few-shot learning (Brown et al., 2020), and as we observe a slight correlation between the number of examples and paraphrase and transformation accuracy⁶, we experiment whether providing additional examples

⁵The example numbers corresponding to each transformation are shown in Figure 2.

⁶We report a mean paraphrase accuracies of 0.81, 0.91, and 0.90, and mean transformation accuracies of 0.66, 0.77, and 0.80 for operations that have 1, 2, and 3 examples, respectively.

improves GPT-4’s performance in the more difficult paraphrase operations (left-most column, and bottom right of Figure 3). The operations we focus on are actor/action substitution, verb/adjective conversion, and manipulator/device substitution, each having 1 provided example in the original prompt.

Figure 4 presents the model accuracies for paraphrasing and the specified transformations for three operations that GPT-4 struggles with. When we add 2, 5, and 10 additional hand-crafted examples to the prompt, we do not see consistent improvement. Additional examples may improve the paraphrasing results, but transformation accuracy does not increase. In fact, higher paraphrase accuracy might even be detrimental to transformation accuracy, because the model prioritizes paraphrasing, if the two criteria seem conflicting. The inconsistency in improving with additional ICL examples suggests that these specific transformations may be challenging to process, possibly due to a lack of training data involving such transformations. Further research is necessary for a deeper understanding of this phenomenon.

5 Related Work

Previous work related to diverse paraphrasing has studied the generation of specific linguistic features, for instance on lexical (e.g., Thompson and Post, 2020) or syntactic level (Iyyer et al., 2018; Chen et al., 2019; Sun et al., 2021, *i.a.*), or controlling for various granularities (Vahtola et al., 2023).

Additionally, previous research has presented various taxonomies of paraphrase types for better understanding of the diverse paraphrase phenomena. Vila et al. (2014) propose a typology of 24 paraphrase types spanning three levels of granularity, while Dutrey et al. (2010) define rephrasing modifications extracted from the revision history of Wikipedia. Less fine-grained categorizations can include for instance differences in specificity or tone (Kanerva et al., 2021). Bhagat and Hovy (2013) propose a list of 25 empirically validated paraphrase transformations with a systematic definition and examples of each transformation.

Detection and generation of diverse paraphrases leveraging a corpus of various paraphrase types (Kovatchev et al., 2018) has been proposed (Wahle et al., 2023). In a concurrent work, Meier et al. (2024) leverage the linguistic phenomena defined in Barrón-Cedeño et al. (2013) to generate specific types of paraphrases. Meier

et al. (2024) also gather human annotations to analyze the accuracy of GPT-3.5 across the different paraphrase types and to evaluate how human annotators rank the generated paraphrases. Their findings are in line with ours, suggesting that LLMs struggle with performing more complex paraphrase transformations. Conversely to the framework of paraphrase operations that we use, the phenomena outlined in Barrón-Cedeño et al. (2013) can often manifest themselves in various surface-form alternations (i.e., one *phenomenon* can include multiple *operations*) as they attempt to capture the general phenomena rather than providing specific mechanisms for paraphrasing. Furthermore, we focus on analyzing the performance of LLMs on various specific paraphrase transformations given their detailed linguistic definitions, and connect the theoretical perspectives of paraphrasing with generative language models and human understanding of semantic similarity.

Another line of related work has focused on benchmarking various pretrained language models, such as BERT (Devlin et al., 2019), across a diverse range of downstream tasks, e.g., GLUE (Wang et al., 2018), SentEval (Conneau and Kiela, 2018), and SICK (Marelli et al., 2014), or a limited range of linguistic phenomena (Marvin and Linzen, 2018; Jumelet and Hupkes, 2018; Ettinger, 2020; Vahtola et al., 2022). Diverging from this line of work, we focus on the capabilities of one state-of-the-art LLM and connect human perception of semantic equivalence to the theory and practice of diverse paraphrasing. In particular, we propose a method and conduct a pilot study to analyze how LLMs manage semantic abstractions in the context of systematically defined paraphrase transformations.

6 Conclusions

In this paper, we design a methodology for testing LLMs to analyze whether they can follow theoretically motivated instructions in the case of paraphrase generation. We utilize explicit linguistic prompts to guide complex transformations and evaluate the results based on human assessment.

Using this framework, we conduct a focused case study on the capabilities of GPT-4 in accurately generating paraphrases. This study is based on 25 paraphrase transformations provided in Bhagat and Hovy (2013), whose definitions of the transformations serve as prompts for few-shot learning. We have customized a web-interface for collecting

manual annotations for the generated sentences in order to assess how accurately the model produces paraphrases that follow the specified transformations.

Our findings indicate that GPT-4 can effectively follow detailed linguistic instructions to generate paraphrastic sentence pairs through simple, local transformations. However, it often prioritizes simple lexical or syntactic substitutions for paraphrasing instead of following specified transformation guidelines. This is especially true when the transformations trigger more complex alternations, indicating limitations in controllability and its ability to process complex linguistic instructions. Furthermore, increasing the number of examples for few-shot in-context learning does not seem to improve the model’s ability to accurately produce paraphrase pairs involving complex operations. This suggests that the model may still lack sufficient proficiency in these linguistic structures. Future work could include a more comprehensive evaluation of how additional few-shot examples, encompassing a broader range of operations, influence performance.

The presented methodology opens many alternative directions for further research. The use of systematic linguistic instructions in text generation tasks is still very much under-explored. Theoretically controlled prompts may help to further understand the abilities of LLMs to generalize and follow explicit rules and guidelines. Such prompts can also be used to compare and benchmark different models about their abstraction capabilities, and the analysis of the results can also be combined with interpretability studies of the network itself in case model weights are openly available.

Limitations

We cover a comprehensive list of transformations, which requires substantial annotations to properly analyze the effect of the instructions. The number of examples for each prompt is still limited in our study but provides a systematic view on linguistically motivated paraphrase generation. Another limitation is the focus on one particular model, GPT-4. Future work could compare the results to other models to deepen our understanding of what and how LLMs learn about human language, even though this is a moving target that is impossible to handle exhaustively. Preliminary studies indicated that GPT-4 is better in handling the complex

instructions we used than other available models. This motivated our choice to look at the limitations of state-of-the-art generative models as GPT-4 abilities in this space currently serve as an upper bound for all the other LLMs. Additional prompt engineering may also be possible to further push the results, and chain-of-thought experiments would also be interesting to study in connection with the task. Finally, we would also like to extend the experiments and annotations in order to expand the dataset and the analyses that can be made on top of the collection.

Acknowledgements

At the time of this research, Teemu Vahtola was supported by the Behind the Words project, funded by the Research Council of Finland. Songbo Hu is supported by the Cambridge International Scholarship. We would like to acknowledge the annotators for their valuable contributions to this work.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Alberto Barrón-Cedeño, Marta Vila, M. Antònia Martí, and Paolo Rosso. 2013. Plagiarism meets paraphrasing: Insights for the next generation in automatic plagiarism detection. *Computational Linguistics*, 39(4):917–947.
- Rahul Bhagat and Eduard Hovy. 2013. Squibs: What is a paraphrase? *Computational Linguistics*, 39(3):463–472.
- 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. Curran Associates, Inc.
- Chris Callison-Burch, Philipp Koehn, and Miles Osborne. 2006. Improved statistical machine translation using paraphrases. In *Proceedings of the Human Language Technology Conference of the NAACL*,

- Main Conference*, pages 17–24, New York City, USA. Association for Computational Linguistics.
- Mingda Chen, Qingming Tang, Sam Wiseman, and Kevin Gimpel. 2019. Controllable paraphrase generation with a syntactic exemplar. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5972–5984, Florence, Italy. Association for Computational Linguistics.
- Alexis Conneau and Douwe Kiela. 2018. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Mathias Creutz. 2018. Open subtitles paraphrase corpus for six languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- 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.
- Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 350–356, Geneva, Switzerland. COLING.
- Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. 2024. A survey on in-context learning.
- Camille Dutrey, Houda Bouamor, Delphine Bernhard, and Aurélien Max. 2010. Local modifications and paraphrases in wikipedia’s revision history. *Procesamiento del lenguaje natural*, 46:51–58.
- Allyson Ettinger. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.
- Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764, Atlanta, Georgia. Association for Computational Linguistics.
- Som Gupta and Sanjai Kumar Gupta. 2019. Abstractive summarization: An overview of the state of the art. *Expert Systems with Applications*, 121:49–65.
- Zellig S. Harris. 1957. Co-occurrence and transformation in linguistic structure. *Language*, 33(3):283–340.
- Richard P Honeck. 1971. A study of paraphrases. *Journal of Verbal Learning and Verbal Behavior*, 10(4):367–381.
- Shudong Huang, David Graff, and George Doddington. 2002. Multiple-translation chinese corpus. Linguistic Data Consortium.
- Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. 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 1875–1885, New Orleans, Louisiana. Association for Computational Linguistics.
- Jaap Jumelet and Dieuwke Hupkes. 2018. Do language models understand anything? on the ability of LSTMs to understand negative polarity items. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 222–231, Brussels, Belgium. Association for Computational Linguistics.
- Jenna Kanerva, Filip Ginter, Li-Hsin Chang, Iiro Rastas, Valtteri Skantsi, Jemina Kilpeläinen, Hanna-Mari Kupari, Jenna Saarni, Maija Sevón, and Otto Tarkka. 2021. Finnish paraphrase corpus. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 288–298, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- Venelin Kovatchev, M. Antònia Martí, and Maria Salamó. 2018. ETPC - a paraphrase identification corpus annotated with extended paraphrase typology and negation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14)*, pages 216–223, Reykjavik, Iceland. European Language Resources Association (ELRA).
- Rebecca Marvin and Tal Linzen. 2018. Targeted syntactic evaluation of language models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Brussels, Belgium. Association for Computational Linguistics.
- Stephen Mayhew, Klinton Bicknell, Chris Brust, Bill McDowell, Will Monroe, and Burr Settles. 2020. Simultaneous translation and paraphrase for language

- education. In *Proceedings of the Fourth Workshop on Neural Generation and Translation*, pages 232–243, Online. Association for Computational Linguistics.
- Ramtin Mehdizadeh Seraj, Maryam Siahbani, and Anoop Sarkar. 2015. Improving statistical machine translation with a multilingual paraphrase database. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1379–1390, Lisbon, Portugal. Association for Computational Linguistics.
- Dominik Meier, Jan Philip Wahle, Terry Ruas, and Bela Gipp. 2024. Towards human understanding of paraphrase types in chatgpt.
- Igor Mel’čuk. 2012. *Semantics: From meaning to text. Volume 1*. John Benjamins.
- Sergiu Nisioi, Sanja Štajner, Simone Paolo Ponzetto, and Liviu P. Dinu. 2017. Exploring neural text simplification models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 85–91, Vancouver, Canada. Association for Computational Linguistics.
- Sneha Rajana, Chris Callison-Burch, Marianna Apidianaki, and Vered Shwartz. 2017. Learning antonyms with paraphrases and a morphology-aware neural network. In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*, pages 12–21, Vancouver, Canada. Association for Computational Linguistics.
- Andrea Sottana, Bin Liang, Kai Zou, and Zheng Yuan. 2023. Evaluation metrics in the era of GPT-4: Reliably evaluating large language models on sequence to sequence tasks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8776–8788, Singapore. Association for Computational Linguistics.
- Jiao Sun, Xuezhe Ma, and Nanyun Peng. 2021. AE-SOP: Paraphrase generation with adaptive syntactic control. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5176–5189, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Brian Thompson and Matt Post. 2020. Paraphrase generation as zero-shot multilingual translation: Disentangling semantic similarity from lexical and syntactic diversity. In *Proceedings of the Fifth Conference on Machine Translation*, pages 561–570, Online. Association for Computational Linguistics.
- Teemu Vahtola, Mathias Creutz, and Jörg Tiedemann. 2022. It is not easy to detect paraphrases: Analysing semantic similarity with antonyms and negation using the new SemAntoNeg benchmark. In *Proceedings of the Fifth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 249–262, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Teemu Vahtola, Mathias Creutz, and Jrg Tiedemann. 2023. Guiding zero-shot paraphrase generation with fine-grained control tokens. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 323–337, Toronto, Canada. Association for Computational Linguistics.
- Marta Vila, M Antònia Martí, Horacio Rodríguez, et al. 2014. Is this a paraphrase? what kind? paraphrase boundaries and typology. *Open Journal of Modern Linguistics*, 4(01):205.
- Jan Philip Wahle, Bela Gipp, and Terry Ruas. 2023. Paraphrase types for generation and detection. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12148–12164, Singapore. Association for Computational Linguistics.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

A Annotation Setup

Figure 5 presents an example of the web-based annotation tool we used for collecting the manual annotations.

Paraphrase Annotation Experiment

> Contact

▼ Instruction

Sentences or phrases that convey the same meaning are called **paraphrases**. For instance, sentences (1) and (2) involve paraphrasing through **synonym substitution**; the verb "seat" is substituted to another verb, "accommodate", and the resulting sentence (2) essentially carries the same meaning with the original sentence (1). **Synonym substitution** can be defined as follows:

synonym substitution

Definition: Replacing a word/phrase by a synonymous word/phrase, in the appropriate context, results in a paraphrase of the original sentence/phrase. This category covers the special case of genitives, where the clitic's is replaced by other genitive indicators like of, the, and so forth. This category also covers near-synonymy, that is, it allows for changes in evaluation, connotation, and so on, of words or phrases between paraphrases.

Example paraphrase pair 1:

(1) The school said that their buses seat 40 students each.

(2) The school said that their buses accommodate 40 students each.

A paraphrasing operation can involve more complex lexical or syntactic transformations. The following example involves **verb/noun conversion**. Here, paraphrasing is performed by changing a verb to its nominalized noun form, accompanied by the addition/deletion of appropriate function words and sentence restructuring. The sentences (3) and (4) are a pair of paraphrases involves **verb/noun conversion**. Paraphrasing is performed by applying the operation defined below (left).

verb/noun conversion

Definition: Replacing a verb by its corresponding nominalized noun form and vice versa, in the appropriate context, results in a paraphrase of the original sentence/phrase. This change may be accompanied by the addition/deletion of appropriate function words and sentence restructuring. This often generates the most strictly meaning-preserving paraphrase.

Example paraphrase pair 2:

(3) The virus spread over two weeks.

(4) Two weeks saw the spreading of a virus.

In this example, the verb "spread" is converted to its nominalized noun "spreading", with the addition/deletion of appropriate function words and sentence restructuring, resulting in a paraphrase pair.

A generated paraphrase can undergo multiple paraphrasing operations. Consider the sentences (5) and (6):

change of voice

Definition: Changing a verb from its active to passive form and vice versa results in a paraphrase of the original sentence/phrase. This change may be accompanied by the addition/deletion of appropriate function words and sentence restructuring. This often generates the most strictly meaning-preserving paraphrase.

Example paraphrase pair 3:

(5) She won a difficult spelling competition.

(6) The challenging spelling competition was won by her.

This example involves both synonym substitution, "difficult" is substituted to its synonym "challenging", and **change of voice**, as the sentence is changed from active to passive voice. In this example, saying that the example follows synonym substitution is correct. So is saying that it follows **change of voice**.

Finally, if the sentences have multiple clauses, it is enough if the desired transformation appears in only one or all of the clauses. Examples (7), (8) and (9) illustrate change of voice in this scenario:

change of voice

Definition: Changing a verb from its active to passive form and vice versa results in a paraphrase of the original sentence/phrase. This change may be accompanied by the addition/deletion of appropriate function words and sentence restructuring. This often generates the most strictly meaning-preserving paraphrase.

Example paraphrase pair 3:

(7) I cooked some food and you ate it.

(8) Some food was cooked by me and you ate it.

(9) Some food was cooked by me and eaten by you.

In this study, we ask you to annotate whether the provided sentence pair accurately reflects the transformation described in the provided definition. We have included 1-3 example pairs demonstrating the paraphrase operation under consideration. Read the definition and the provided examples carefully. Additionally, we ask you to assess whether the provided sentences are paraphrases of each other and to what extent. Please, evaluate the equivalency of the given paraphrase pairs using the following scale:

- **Unrelated:** the sentences have different meanings and are not paraphrases.
- **Non-paraphrases:** the sentences have some semantic overlap but cannot be considered paraphrases.
- **Contextual paraphrases:** the sentences can be paraphrases in some but not in all contexts.
- **Full paraphrases:** the sentences have the same meaning and can be considered paraphrases in all contexts.

In the sections below, you will find examples demonstrating how to effectively use this annotation tool to complete the task.

▼ Examples

synonym substitution example

Step 1: You will be assigned a set of tasks corresponding to each predefined paraphrase operation. For instance, synonym substitution is one such paraphrase operation.

synonym substitution

Definition: Replacing a word/phrase by a synonymous word/phrase, in the appropriate context, results in a paraphrase of the original sentence/phrase. This category covers the special case of genitives, where the clitic's is replaced by other genitive indicators like of, the, and so forth. This category also covers near-synonymy, that is, it allows for changes in evaluation, connotation, and so on, of words or phrases between paraphrases.

Step 2: You will be provided with the definition of each paraphrase operation along with one or more example pairs. It is crucial that you carefully read both the definition and the examples. All definitions and examples will be displayed against a green background to ensure easy visibility.

Example paraphrase pair 1:

Chris is slim.

Chris is slender.

Step 4: After you have read the sentence pair, here is your task. You have three questions for your task. You must answer all three questions here.

synonym substitution task 1

The cat dozed on the rug.

The cat slept on the mat.

Does the pair of sentences on the left follow the synonym substitution operation? ☒ Yes ☐ No

Is the pair of sentences on the left a paraphrase of each other? ☒ Yes ☐ No

To what extent the provided sentences are paraphrases of each other?

☐ Unrelated ☐ Non-paraphrases ☐ Contextual paraphrases ☒ Full paraphrases

synonym substitution task 2

Sarah enjoys painting.

Sarah is painting.

Does the pair of sentences on the left follow the synonym substitution operation? ☒ Yes ☐ No

Is the pair of sentences on the left a paraphrase of each other? ☐ Yes ☒ No

To what extent the provided sentences are paraphrases of each other?

☐ Unrelated ☒ Non-paraphrases ☐ Contextual paraphrases ☐ Full paraphrases

Figure 5: A screenshot of our web-based annotation tool.

766

SweClinEval: A Benchmark for Swedish Clinical Natural Language Processing

Thomas Vakili, Martin Hansson, and Aron Henriksson

Department of Computer and Systems Sciences

Stockholm University, Kista, Sweden

{thomas.vakili, martin.hansson, aronhen}@dsv.su.se

Abstract

The lack of benchmarks in certain domains and for certain languages makes it difficult to track progress regarding the state-of-the-art of NLP in those areas, potentially impeding progress in important, specialized domains. Here, we introduce the first Swedish benchmark for clinical NLP: *SweClinEval*. The first iteration of the benchmark consists of six clinical NLP tasks, encompassing both document-level classification and named entity recognition tasks, with real clinical data. We evaluate nine different encoder models, both Swedish and multilingual. The results show that domain-adapted models outperform generic models on sequence-level classification tasks, while certain larger generic models outperform the clinical models on named entity recognition tasks. We describe how the benchmark can be managed despite limited possibilities to share sensitive clinical data, and discuss plans for extending the benchmark in future iterations.

1 Introduction

The field of natural language processing (NLP) has seen several important breakthroughs in the past decade. Currently, the field is dominated by pre-trained transformers models (Vaswani et al., 2017) that can be used to solve a wide and – ideally – diverse set of tasks. The capabilities of these models have to a large degree been tracked through the use of *benchmarks*, significantly helping to drive progress in the area. These evaluation suites test how the models perform on different pre-defined tasks and allow for comparisons between models and approaches.

While there are many benchmarks available, there are also many potential uses for NLP that

they do not cover. Frequently, evaluations rely on English data (Joshi et al., 2020; Søgaaard, 2022). However, a model performing well on an English benchmark in no way guarantees similar performance if the language changes. Additionally, benchmarks such as GLUE (Wang et al., 2018) tend to focus on tasks formulated for general-domain data. With increasing calls for NLP to be applied to specific domains, such as the clinical domain, there is a pressing need for benchmarks that address these areas.

The clinical domain, in particular, suffers from a lack of datasets for evaluating NLP systems. One critical reason for this is the inherently sensitive nature of clinical data. There are multiple studies (Carlini et al., 2021; Nasr et al., 2023) demonstrating the potential risks of using sensitive data for machine learning – let alone sharing data in their raw form. That said, there are some widely used resources for clinical NLP. Prominent examples include the various versions of MIMIC (Johnson et al., 2022) and the i2b2 datasets (Murphy et al., 2010). Crucially, these datasets predominantly evaluate NLP systems on data in English or other higher-resourced languages.

In this paper, we introduce the first Swedish benchmark based on real clinical NLP data: *SweClinEval*. This benchmark consists of datasets built from electronic health records from the Health Bank (Dalianis et al., 2015) and includes a wide range of clinical tasks. These tasks include three different document-level sequence classification tasks and three token-level named entity recognition (NER) tasks. This introduction of *SweClinEval* includes nine different models, and future additions will be added to the benchmarks online leaderboard¹.

The evaluations presented in this paper show that many models targeting Swedish data per-

¹The leaderboard of *SweClinEval* is available at: <https://sweclineval.dsv.su.se>

form strongly on our benchmark. However, the performances vary, and several interesting trends emerge from our results. These results highlight the importance of continuing to focus on domain-specific evaluations for languages other than English. Our results demonstrate the current state of Swedish clinical NLP, and the benchmark serves as an important tool for monitoring progress in this important NLP domain.

2 Related Research

The NLP community has seen impressive advances in the past few years with the advent of LLMs. Several new model architectures have been proposed since Vaswani et al. (2017) described the transformer, and new models are released at a rapid pace. These LLMs aim to be general-purpose models, with task-specific applications requiring only smaller adjustments in the form of fine-tuning or prompt engineering. In response to this new paradigm, there has been an increasing focus on creating benchmarks that capture the nuanced difference in performance in the growing plethora of models.

2.1 General-Domain Benchmarks

Benchmarks come with different objectives and designs. A prominent example is the GLUE (Wang et al., 2018) family of benchmarks. The original *General Language Understanding Evaluation* (GLUE) benchmark aimed to, as the name suggests, capture a wide range of capabilities that act as proxies for natural language understanding. As models have become more powerful, the NLP community has responded with more varied and difficult benchmarks. These include the SuperGLUE (Wang et al., 2019) benchmark that introduces more difficult tasks, and the XGLUE benchmark (Liang et al., 2020) that also examines the multilingual capabilities of models.

2.2 Swedish Benchmarks

The vast majority of papers at NLP conferences focus on English data (Søgaard, 2022), to the detriment of smaller and less well-resourced languages. The introduction of multilingual benchmarks such as XGLUE is in part a response to this dominance of English-only datasets.

Another development is the creation of language-specific benchmarks. For Swedish, this trend has materialized in the form of benchmarks

such as the Superlim² (Berdicevskis et al., 2023) and OverLim³ benchmarks. These benchmarks mirror the structure of the GLUE family of benchmarks, but use datasets that specifically use Swedish data.

An important benchmark, especially for the purposes of this paper, is the ScandEval (Nielsen, 2023) benchmark. This benchmark is multilingual but focuses mainly on the Scandinavian language family. LLMs for these languages have been found to benefit from training on shared datasets. The ScandEval benchmark was also used to determine which models to benchmark, as detailed in Section 3.2.

2.3 Clinical Benchmarks

The most commonly used benchmarks aim to measure general-purpose capabilities in a general-domain setting. However, many important applications of NLP are domain-specific. In this paper, we focus on NLP for clinical data, which has several domain-specific features. Due to the setting in which they are produced, clinical data are often riddled with domain-specific acronyms and terminology that can be harder for general-domain models to process (Dalianis, 2018). Furthermore, clinical datasets are difficult to share due to the inherently sensitive nature of the data.

Nevertheless, there have been efforts to create benchmarks that measure the clinical or biomedical capabilities of LLMs. BLURB (Gu et al., 2021) is a benchmark in the vein of GLUE and includes a wide range of clinical tasks. This benchmark highlighted the shortcomings of general-domain models and the benefits of using LLMs specific to the clinical domain. In contrast, the later Dr. Bench (Gao et al., 2023) benchmark shows that general-domain models can indeed out-compete domain-specific models on certain tasks. These diverging conclusions exemplify the need for diverse domain-specific benchmarks to monitor the progress of LLMs in the clinical domain.

A recent benchmark highly relevant for Swedish biomedical NLP is the *Swedish Medical Benchmark* introduced by Moëll and Farestam (2024). This benchmark is comprised of a selection of four datasets with multiple-choice questions. These datasets were collected from public

²Superlim is Swedish for super glue, a reference to the SuperGLUE benchmark.

³<https://huggingface.co/datasets/KBLab/overlim>

sources and probe LLMs for biomedical knowledge. A benefit of using publicly available data is that the data can be shared. On the other hand, such data are not representative of the types of clinical data and tasks encountered when creating, for example, a system interfacing with patient records.

The main contribution of this paper is the introduction of the SweClinEval benchmark. This benchmark is not only focused on the clinical domain, but is the first benchmark that monitors the state of Swedish clinical NLP using real electronic patient records for realistic clinical tasks.

3 Methods and Materials

Creating this first rendition of SweClinEval involved collecting resources for evaluation and deciding how to conduct the evaluations. This section describes the datasets used for the benchmark and the models that were tested, and how they were chosen. The design of the evaluations and the metrics used for comparing models are also described.

3.1 Datasets

The benchmark consists of six datasets that are part of the Health Bank (Dalianis et al., 2015) infrastructure⁴. The Health Bank consists of over 2 million Swedish electronic health records written between 2006 and 2014 from a range of different clinical units in Sweden. The datasets have been collected for more than a decade, either through manual annotation or by mining information from the Health Bank data. Three of the datasets are document-level classification tasks, and the other three are token-level NER tasks.

ICD-10 The Stockholm EPR Gastro ICD-10 Corpus (Remmer et al., 2021) is a document-level classification task where discharge summaries related to gastrointestinal patients are assigned high-level diagnosis code blocks. These 10 different code blocks encode information about what type of diagnosis was assigned to the patient. The task is a multi-label classification task, meaning that each document can be associated with more than one code block.

ADE The Stockholm EPR ADE ICD-10 Corpus (Vakili et al., 2022) is another document-level classification task that determines whether or not a discharge summary describes a patient suffering from an adverse drug event. This is a binary classification problem.

Factuality The Stockholm EPR Diagnosis Factuality Corpus (Velupillai, 2011; Velupillai et al., 2011) is the third document-level classification task. This manually annotated corpus assigns a *factuality* level to the diagnoses of each clinical note. These different levels describe the confidence with which a diagnosis was decided. The six different classes are: *Certainly Negative*, *Probably Negative*, *Possibly Negative*, *Possibly Positive*, *Probably Positive*, and *Certainly Positive*.

Factuality NER This version of the Stockholm EPR Diagnosis Factuality Corpus is a token-level NER task. The task involves assigning the same six labels to tokens in each document that indicate a diagnosis. The task is to both detect these diagnoses and assign them a factuality level. This version also includes an *Other* tag for clinically relevant information that is not indicating factuality.

Clinical Entity NER The Stockholm EPR Clinical Entity Corpus (Skeppstedt et al., 2014) is a manually annotated NER corpus that describes a task in which the model needs to identify clinically relevant terms. These are divided into four classes: *Diagnosis*, *Findings*, *Body Parts*, and *Drugs*. The model needs to detect tokens associated with these classes and assign them the correct labels.

PHI NER The final corpus used in the benchmark is the Stockholm EPR PHI Corpus (Dalianis and Velupillai, 2010). This corpus consists of patient records and has been manually annotated for named entities describing personally identifiable protected health information (PHI). Each instance of PHI is assigned one of nine classes: *First Name*, *Last Name*, *Age*, *Phone Number*, *Partial Date*, *Full Date*, *Location*, *Health Care Unit*, and *Organization*.

Additional statistics about the six datasets are listed in Table 1. None of the datasets have been

⁴This research has been approved by the Swedish Ethical Review Authority under permission no. 2019-05679.

adapted for use with prompt-style autoregressive language models. This limitation is reflected in the model selection for this paper and adapting the datasets for broader use is left to future iterations of SweClinEval.

3.2 Models

Nine different models were included for the experiments in this paper and are listed in Table 2. Two of these – SweDeClin-BERT and SweClin-BERT – were specifically created for use in Swedish clinical NLP and have previously shown strong performance on the datasets in SweClinEval (Vakili et al., 2024). Additionally, seven general-domain models known to perform well for Swedish data were included. These seven models were selected based on their performance in the ScandEval (Nielsen, 2023) benchmark.

The majority of the models are based on the BERT/RobERTa architecture (Devlin et al., 2019; Liu et al., 2019). The RemBERT (Chung et al., 2020) and Multilingual E5 Large (Wang et al., 2024) models are based on their own transformer architectures. These two models also exhibit the greatest language diversity in their training data. The training data for the *RobERTa Large* and *BERT Large* models from AI Sweden are also multilingual. These were trained using *The Nordic Pile* corpus (Öhman et al., 2023) which consists mainly of Scandinavian and English data.

Crucially, all nine models are encoder models. This is a limitation imposed by the nature of the datasets, as described in the previous section. It is possible to restructure datasets so that they can be used autoregressively. However, such a conversion would be non-trivial and is left for future research.

3.3 Evaluation Procedure

All nine models were trained and evaluated using the six datasets. To ensure a fair estimate of each model’s performance, the evaluations were done using 10-fold cross-validation. This allowed us to calculate the average performance alongside the standard deviation, enabling a more fair comparison. The comparisons were based on the F_1 scores of each cross-validation.

For each fold in the cross-validation, models were trained for a maximum of three epochs. Early stopping was enabled, and the best-performing checkpoint was used to predict the test set in each fold. The F_1 scores used for the comparisons were based on the average score from

each fold and the standard deviation. For the NER tasks, these were the token-level micro F_1 scores. The *PHI NER* task uses the IOB scheme to mark where an entity begins and ends, and this distinction was included in the evaluation. The document-level sequence classification tasks instead rely on F_1 scores weighted for the support of each class in the test set.

4 Results

Nine models were evaluated using 10-fold cross-validation for six different datasets, resulting in 540 evaluations. The average F_1 scores and their deviations are listed in Table 3.

For the sequence-level classification tasks, the highest average F_1 scores are consistently obtained using the domain-adapted models. The same is not true for the token-level NER tasks. For these tasks, the highest F_1 scores were obtained by the general-domain *RobERTa Large* model from AI Sweden. However, the domain-adapted *SweDeClin-BERT* model has the second-highest average F_1 scores for the *Factuality NER* and *Clinical Entity NER* tasks.

The different average F_1 scores vary substantially between the best- and worst-performing models. Nevertheless, the standard deviations are large. This means that many of the averages are within a standard deviation of a competing model. This necessarily limits the analysis into which models are *best*, since randomness has a strong influence on the variability in the F_1 scores.

In addition to the predictive performance, Table 4 also lists the processing time of each model when performing inference. Unsurprisingly, the smaller models are faster to run. These figures are based on the HuggingFace implementations of each model running on an *Nvidia RTX A5000* GPU. Although the exact inference time will depend on the hardware available, the number indicate the relative cost of running these model in a production environment.

5 Discussion

A few trends emerge from the results in the previous section. There are also some limitations and pointers to future work that are important to discuss. However, we begin by discussing the findings from our results.

As previously mentioned, the highest average F_1 scores in the sequence classification tasks are

| Task | Type | Classes | Documents | Tokens |
|---------------------|----------------|---------|-----------|---------|
| ICD-10 | Classification | 10 | 6,062 | 930,550 |
| ADE | Classification | 2 | 21,725 | 931,778 |
| Factuality | Classification | 6 | 3,710 | 102,223 |
| Factuality NER | NER | 7 | 3,822 | 286,205 |
| Clinical Entity NER | NER | 4 | 3,120 | 178,672 |
| PHI NER | NER | 9 | 29,560 | 282,820 |

Table 1: Six different datasets were used in the benchmark evaluation. Three of these are NER tasks and three are sequence classification tasks. This table lists the datasets alongside their size, the number and classes, and the types of classification they target.

| Model | Parameters | Paper |
|-------------------------|------------|---------------------------|
| SweDeClin-BERT | 125 M | (Vakili et al., 2022) |
| SweClin-BERT | 125 M | (Lamproudis et al., 2021) |
| KB-BERT Base | 125 M | (Malmsten et al., 2020) |
| AI Nordics BERT Large | 335 M | N/A ⁵ |
| AI Sweden RoBERTa Large | 355 M | N/A ⁶ |
| AI Sweden BERT Large | 369 M | N/A ⁷ |
| KB-BERT Large | 370 M | N/A ⁸ |
| Multilingual E5 Large | 560 M | (Wang et al., 2024) |
| RemBERT | 576 M | (Chung et al., 2020) |

Table 2: In this initial edition of the SweClinEval benchmark, nine different models were evaluated. All models are encoder models, and they are listed here in order of parameter count. When available, the paper that introduced the model is listed. SweDeClin-BERT and SweClin-BERT are the only models created specifically for Swedish clinical NLP.

achieved by the domain-adapted models. This indicates that, at least for these tasks, domain adaptation results in better performance on clinical NLP tasks. On the other hand, this finding is not as clear when examining the NER tasks. While the domain-adapted models perform competitively, the best-performing model on all three NER tasks is AI Sweden’s *RoBERTa Large* model.

Crucially, the models differ greatly in size. The smaller models are around three times smaller than the medium-sized models, and more than four times smaller than the largest models. The comparatively strong performance of the domain-adapted models, which are both small, is more im-

pressive when seen from this perspective. Domain adaptation seems to allow smaller models to compete with larger counterparts. Naturally, this leads to the question of whether this finding holds true for larger models, too. The two clinical models are initialized from *KB-BERT Base*, and an interesting direction for future work could be examining if initializing from larger models produces analogous results. The *RoBERTa Large* model from AI Sweden would be an interesting candidate, given its strong performance on the NER tasks. In any case, the benefits from domain adaptation align with many previous studies (Gu et al., 2021; Lamproudis et al., 2021).

Perhaps somewhat surprisingly, parameter count itself does not seem to be a determining factor in what models are the strongest. This is not only the case when comparing domain-adapted and general-domain models. For example, *KB-BERT Base* and *KB-BERT Large* were both trained by the same organization, and are from the same model family. The main difference between the

⁵<https://huggingface.co/AI-Nordics/bert-large-swedish-cased>

⁶<https://huggingface.co/AI-Sweden-Models/roberta-large-1160k>

⁷<https://huggingface.co/AI-Sweden-Models/bert-large-nordic-pile-1M-steps>

⁸<https://huggingface.co/KBLab/megatron-bert-large-swedish-cased-165k>

| Model | Size | ICD-10 | Factuality | ADE |
|-------------------------|------|--------------------|--------------------|--------------------|
| | | Classification | Classification | Classification |
| SweDeClin-BERT | S | <u>0.832±0.011</u> | 0.735±0.018 | 0.203±0.022 |
| SweClin-BERT | S | 0.836±0.014 | <u>0.731±0.021</u> | <u>0.196±0.014</u> |
| KB-BERT Base | S | 0.801±0.015 | 0.671±0.017 | 0.185±0.012 |
| AI Nordics BERT Large | M | 0.811±0.012 | 0.657±0.025 | 0.192±0.013 |
| AI Sweden RoBERTa Large | M | 0.816±0.018 | 0.594±0.126 | 0.159±0.028 |
| AI Sweden BERT Large | M | 0.816±0.012 | 0.654±0.032 | 0.167±0.057 |
| KB-BERT Large | M | 0.801±0.013 | 0.683±0.019 | 0.190±0.011 |
| Multilingual E5 Large | L | 0.824±0.013 | 0.525±0.074 | 0.192±0.015 |
| RemBERT | L | 0.823±0.010 | 0.379±0.059 | 0.149±0.050 |

| Model | Size | Factuality | Clinical Entity | PHI |
|-------------------------|------|--------------------|--------------------|--------------------|
| | | NER | NER | NER |
| SweDeClin-BERT | S | <u>0.623±0.024</u> | <u>0.766±0.034</u> | 0.945±0.012 |
| SweClin-BERT | S | 0.610±0.018 | 0.754±0.038 | 0.938±0.014 |
| KB-BERT Base | S | 0.600±0.025 | 0.743±0.039 | 0.941±0.025 |
| AI Nordics BERT Large | M | 0.612±0.026 | 0.721±0.039 | <u>0.948±0.010</u> |
| AI Sweden RoBERTa Large | M | 0.641±0.011 | 0.779±0.036 | 0.965±0.009 |
| AI Sweden BERT Large | M | 0.513±0.185 | 0.738±0.038 | 0.854±0.285 |
| KB-BERT Large | M | 0.552±0.025 | 0.697±0.046 | 0.936±0.012 |
| Multilingual E5 Large | L | 0.603±0.019 | 0.511±0.339 | 0.608±0.037 |
| RemBERT | L | 0.417±0.026 | 0.600±0.075 | 0.947±0.011 |

Table 3: Nine encoder models were evaluated for sequence classification using six different clinical tasks. Three of the tasks were sequence classification tasks, and three were token-level NER tasks. The performance is summarized using F_1 with standard deviations. The highest F_1 of each task is bolded, and the second highest is underlined. Models are ordered according to ascending parameter count as listed in Table 2 and categorized as *Small*, *Medium*, or *Large* models.

| Model | Sequence | NER |
|-------------------------|----------|---------|
| SweDeClin-BERT | 2.86 ms | 2.85 ms |
| SweClin-BERT | 2.86 ms | 2.84 ms |
| KB-BERT Base | 2.88 ms | 2.87 ms |
| AI Nordics BERT Large | 5.60 ms | 5.56 ms |
| AI Sweden RoBERTa Large | 6.91 ms | 6.05 ms |
| AI Sweden BERT Large | 5.60 ms | 5.56 ms |
| KB-BERT Large | 8.76 ms | 8.67 ms |
| Multilingual E5 Large | 6.08 ms | 6.03 ms |
| RemBERT | 9.38 ms | 9.36 ms |

Table 4: The different models used in the benchmark use different architectures and are of different sizes. This table lists the time of each model for inference on one sample, both for sequence classification and NER.

models is that the larger model consists of more parameters and was trained using a much larger corpus. Nevertheless, *KB-BERT Base* actually outperforms its larger counterpart in some cases.

While the large standard deviations call for cautious interpretations of the results, it is at least clear the larger model is not outperforming its smaller competitor.

On the other hand, parameter count clearly influences the inference speed of the models, as indicated in Table 4. While this is not surprising, it is worth mentioning. Other benchmarks, such as the GLUE benchmark, do not always present this information. However, inference speed can be important in practice, especially when differences in performance are small. Smaller and faster models require less expensive hardware, which can be important in cases where it is not possible to use cloud providers to run the models. This is frequently the case for clinical uses, due to the sensitivity of clinical data.

6 Conclusions

In this paper, we present SweClinEval – the first Swedish benchmark for clinical NLP. We evaluate

a wide range of encoder-style LLMs for six different Swedish clinical NLP tasks. This effort represents the first such evaluation to be conducted, and forms a basis for future monitoring of the advances in Swedish clinical NLP.

The results of this first evaluation indicate several interesting trends. The benchmark results suggest that domain adaptation is an effective strategy for improving the performance of LLMs in the clinical domain, at least for small LLMs. Future research should examine whether this also holds for larger models. Furthermore, the evaluations also show that parameter count alone is not enough to perform strongly in the tasks included in our benchmark.

The aim of this paper is to enable monitoring of the progress within Swedish clinical NLP. Due to privacy constraints, the data cannot be shared. We strongly encourage others interested in Swedish clinical NLP to contact us for inclusion in the benchmark. This pragmatic approach to benchmarking enables us to monitor the progress that is being made, which SweClinEval makes possible.

6.1 Limitations

A limitation of the current version of the benchmark is that it only supports encoder models. This is unfortunate, as there is a strong trend towards using autoregressive models both in fine-tuning and few-shot settings. Future versions of the benchmark would benefit from including versions of the datasets that allow non-encoder models to be evaluated. This is not trivial but, as demonstrated by the ScandEval benchmark, it is possible and is an aim for future iterations of the benchmark. Furthermore, we aim to extend the benchmark with more datasets for tasks such as summarization and question-answering.

A more significant limitation of SweClinEval is that currently, only parts of the data can be shared. This restriction is due to privacy regulations surrounding the inherently sensitive clinical data from which the datasets were created. However, two of the datasets – the *Stockholm EPR PHI Corpus* and the *Stockholm EPR ICD-10 Corpus* – are available in automatically de-identified form for academic users. As the regulatory environment around secondary use of private information changes, it may be possible to share the data more freely in the future. For now, our view is that SweClinEval is a pragmatic solution that allows the

Swedish NLP community to monitor the progress in Swedish clinical NLP.

References

- Aleksandrs Berdicevskis, Gerlof Bouma, Robin Kurtz, Felix Morger, Joey Öhman, Yvonne Adesam, Lars Borin, Dana Dannélls, Markus Forsberg, Tim Isbister, Anna Lindahl, Martin Malmsten, Faton Rekathati, Magnus Sahlgren, Elena Volodina, Love Börjeson, Simon Hengchen, and Nina Tahmasebi. 2023. <https://doi.org/10.18653/v1/2023.emnlp-main.506> Superlim: A Swedish language understanding evaluation benchmark. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8137–8153, Singapore. Association for Computational Linguistics.
- Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. 2021. Extracting Training Data from Large Language Models. In *Proceedings of the 30th USENIX Security Symposium*, pages 2633–2650.
- Hyung Won Chung, Thibault Fèvry, Henry Tsai, Melvin Johnson, and Sebastian Ruder. 2020. <http://arxiv.org/abs/2010.12821> Rethinking embedding coupling in pre-trained language models.
- H. Dalianis, A. Henriksson, M. Kvist, S. Velupillai, and R. Weegar. 2015. HEALTH BANK - A workbench for data science applications in healthcare. In *CEUR Workshop Proceedings*. CEUR-WS.
- Hercules Dalianis. 2018. <https://doi.org/10.1007/978-3-319-78503-5> *Clinical Text Mining*. Springer International Publishing, Cham.
- Hercules Dalianis and Sumithra Velupillai. 2010. <https://doi.org/10.1186/2041-1480-1-6> De-identifying Swedish clinical text - refinement of a gold standard and experiments with Conditional random fields. *Journal of Biomedical Semantics*, 1(1):6.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. <https://doi.org/10.18653/v1/N19-1423> 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.
- YanJun Gao, Dmitriy Dligach, Timothy Miller, John Caskey, Brihat Sharma, Matthew M. Churpek, and Majid Afshar. 2023. <https://doi.org/10.1016/j.jbi.2023.104286> Dr.bench: Diagnostic reasoning benchmark for clinical natural

- language processing. *J. of Biomedical Informatics*, 138(C).
- Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. <https://doi.org/10.1145/3458754> Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *ACM Transactions on Computing for Healthcare*, 3(1):2:1–2:23.
- Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. 2022. <https://doi.org/10.13026/7VCR-E114> MIMIC-IV.
- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The State and Fate of Linguistic Diversity and Inclusion in the NLP World. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.
- Anastasios Lamproudis, Aron Henriksson, and Hercules Dalianis. 2021. Developing a Clinical Language Model for Swedish: Continued Pretraining of Generic BERT with In-Domain Data. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 790–797, Held Online. INCOMA Ltd.
- Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Ruofei Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Daniel Campos, Rangan Majumder, and Ming Zhou. 2020. <https://doi.org/10.18653/v1/2020.emnlp-main.484> XGLUE: A New Benchmark Dataset for Cross-lingual Pre-training, Understanding and Generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6008–6018, Online. 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. *arXiv:1907.11692 [cs]*. ArXiv: 1907.11692.
- Martin Malmsten, Love Börjeson, and Chris Hafenden. 2020. Playing with Words at the National Library of Sweden – Making a Swedish BERT. *arXiv:2007.01658 [cs]*. ArXiv: 2007.01658.
- Birger Moëll and Fabian Farestam. 2024. https://sltc2024.github.io/abstracts/moell_farestam.pdf Swedish Medical Benchmark, an evaluation framework for LLMs in the Swedish medical domain.
- Shawn N. Murphy, Griffin Weber, Michael Mendis, Vivian Gainer, Henry C. Chueh, Susanne Churchill, and Isaac Kohane. 2010. <https://doi.org/10.1136/jamia.2009.000893> Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2). *Journal of the American Medical Informatics Association: JAMIA*, 17(2):124–130.
- Milad Nasr, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A. Feder Cooper, Daphne Ippolito, Christopher A. Choquette-Choo, Eric Wallace, Florian Tramèr, and Katherine Lee. 2023. <https://doi.org/10.48550/arXiv.2311.17035> Scalable Extraction of Training Data from (Production) Language Models. ArXiv:2311.17035 [cs].
- Dan Nielsen. 2023. ScandEval: A benchmark for Scandinavian natural language processing. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 185–201, Tórshavn, Faroe Islands. University of Tartu Library.
- Sonja Remmer, Anastasios Lamproudis, and Hercules Dalianis. 2021. Multi-label Diagnosis Classification of Swedish Discharge Summaries – ICD-10 Code Assignment Using KB-BERT. In *Proceedings of RANLP 2021: Recent Advances in Natural Language Processing, RANLP 2021, 1-3 Sept 2021, Varna, Bulgaria*, pages 1158–1166.
- Maria Skeppstedt, Maria Kvist, Gunnar H Nilsson, and Hercules Dalianis. 2014. Automatic recognition of disorders, findings, pharmaceuticals and body structures from clinical text: An annotation and machine learning study. *Journal of biomedical informatics*, 49:148–158. Publisher: Elsevier.
- Anders Søgaard. 2022. Should We Ban English NLP for a Year? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5254–5260, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Thomas Vakili, Aron Henriksson, and Hercules Dalianis. 2024. End-to-end pseudonymization of fine-tuned clinical BERT models. *BMC Medical Informatics and Decision Making*, 24:162.
- Thomas Vakili, Anastasios Lamproudis, Aron Henriksson, and Hercules Dalianis. 2022. Downstream Task Performance of BERT Models Pre-Trained Using Automatically De-Identified Clinical Data. In *Proceedings of the 13th Conference on Language Resources and Evaluation*, pages 4245–4252. European Language Resources Association.
- 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.

- Sumithra Velupillai. 2011. Automatic classification of factuality levels: A case study on Swedish diagnoses and the impact of local context. In *Fourth International Symposium on Languages in Biology and Medicine, LBM 2011*.
- Sumithra Velupillai, Hercules Dalianis, and Maria Kvist. 2011. Factuality levels of diagnoses in Swedish clinical text. *Studies in Health Technology and Informatics*, 169:559–563.
- 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.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. <https://doi.org/10.18653/v1/W18-5446> GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.
- Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. <http://arxiv.org/abs/2402.05672> Multilingual e5 text embeddings: A technical report.
- Joey Öhman, Severine Verlinden, Ariel Ekgren, Amaru Cuba Gyllensten, Tim Isbister, Evangelia Gogoulou, Fredrik Carlsson, and Magnus Sahlgren. 2023. <https://doi.org/10.48550/arXiv.2303.17183> The Nordic Pile: A 1.2TB Nordic Dataset for Language Modeling. ArXiv:2303.17183.

Dialectal treebanks and their relation with the standard variety: The case of East Cretan and Standard Modern Greek

Socrates Vakirtzian¹, Vivian Stamou², Yannis Kazos^{2,3}, Stella Markantonatou^{2,4}

¹Department of Informatics and Telecommunications, NKUA

²Archimedes, Athena R.C.

³National Technical University of Athens, NTUA

⁴Institute of Language and Speech Processing, Athena R.C.

socratesvak@hotmail.com, [vivianstamou, kazosj, stiliani.markantonatou]@gmail.com

Abstract

We report on the development of the first treebank and parser for Eastern Cretan in the framework of Universal Dependencies (UD). Eastern Cretan is a living but under-resourced dialect of Modern Greek. We have worked on the transcription of oral material and relied on active annotation and knowledge transfer from GUD, a treebank of Standard Modern Greek. Along with its other phonological and morphosyntactic differences from Standard Modern Greek, Eastern Cretan (and other varieties of Modern Greek) makes heavy use of euphonics and voicing that have not been included in the UD annotation guidelines so far. We have provided annotation guidelines for East Cretan euphonics and voicing and included them in the models. Knowledge transfer from the treebank of Standard Modern Greek to the dialectal models helped to initiate annotation via an active annotation procedure.

1 Introduction

The leaps in NLP in recent years have brought considerable efficiency to language analysis tools. This rapid progress has reduced the cost of the oral material-to-linguistically annotated text pipeline and facilitated knowledge transfer from well resourced languages to less resourced ones. At the same time it is challenging because the resulting representation of the less resourced languages may be biased by the massive evidence collected for the richly resourced ones (Bird, 2020). In the face of the increasingly rapid digitization characterising our era, it is a matter of survival for under-resourced languages to gain an independent digital presence that respects their individual nature so that they can be integrated into modern technologies and methods of study.

Considering dialects, the available linguistic data are not only scarce but are also often characterized by a significant lack of consistency in their orthographic representation. This is due to the primarily oral nature of these language varieties. Since our goal was to create language models capable of understanding the current linguistic reality, it was essential to rely on contemporary speech data.

The Eastern Cretan treebank¹ is the first morphosyntactically annotated treebank of a living Modern Greek dialect. Annotation complies to the Universal Dependencies - Version 2 (UD.V2) guidelines (de Marneffe et al., 2021). For Standard Modern Greek (SMG) there are two UD.V2 treebanks, GDT and GUD, with GUD being the most recent one². GUD contains 1,807 sentences (25,493 tokens) randomly selected from fiction texts. We trained models on the Eastern Cretan treebank only, and on the Eastern Cretan treebank plus the GUD (henceforth Eastern Cretan+GUD), to see whether and to what extent SMG can contribute to the development of Eastern Cretan language models.

In Section 2, the basic linguistic differences of the Eastern Cretan dialect from SMG are briefly presented. In Section 3, we present the linguistic resources we used and in Section 4, we provide details about the compilation of the treebank and the handling of specific morphological and syntactic phenomena. In Section 5, we discuss the annotation method, and in Section 6 and 7, we present and comment on the models we developed. In the last three sections we present the limitations of our approach and the conclusions we reached.

¹https://github.com/UniversalDependencies/UD_Greek-Cretan

²https://github.com/UniversalDependencies/UD_Greek-GUD

2 The Eastern Cretan dialect and its relation with SMG

Cretan is a language variety of Modern Greek (MG) primarily spoken on the island of Crete and by the Cretan diaspora. This includes communities of Cretan descent who relocated to Hamidieh in Syria and Western Asia Minor after the 1923 population exchange between Greece and Turkey. The preservation and development of the dialect have been influenced by Crete's long-term isolation from the mainland and the island's domination by non-Greek-speaking powers such as the Arabs, Venetians, and Turks for more than nine centuries. Cretan is divided into two main dialect groups—western and eastern—based on phonological, morphological and lexical characteristics. The two groups share a lot of features that characterise the Cretan dialects. The division aligns with the island's administrative boundaries between the prefectures of Rethymno and Heraklion.

The phenomenon of the gradual decline of MG dialects in the face of SMG is observed. Beyond the social and economic reasons for the depopulation of rural areas, which are the natural speaking environments for these language varieties, efforts to preserve and reproduce them have not yet taken on a systematic character. Specifically, the dialects have not been systematized regarding their orthographic representation and are not taught.

Unlike most other MG dialects, Cretan is not endangered and remains widely used as the primary mode of communication in many parts of the island. However, as all MG dialects, it is under-resourced, in particular with regard to resources that would support its presence in the contemporary technological landscape.

Below we will mention some of the distinctive features that the Cretan dialect retains, according to the studies by Kontosopoulos (1969, 2008).

Phonological level

1. Palatalization and affrication of /k/, /g/, /x/, /ɣ/ before the phonemes /e/, /i/. The corresponding cretan allophones in the aforementioned phonetic environment are respectively: [tʃ], [dʒ], [ç], [ʒ]
2. Fricativation of /t/ to /θ/ and /d/ to /ð/ before semivocalic phonemes:

- [ta 'ma. tʃa] → [ta 'ma. θʃa]
- [ku.ve.'dʒa.zo] → [ku.ve.'ðʒa.zo]

3. Realization of the clusters <μπ>, <ντ>, <γχ> as voiced plosive phonemes [b], [d], [g] without the nasal element in any position.
4. Development of the euphonic sounds [e], [n], and [j] to avoid hiatus in cases of word coarticulation (see also Section 4.3):
 - <τον βάνω>, [ton 'va. no] → <τονε βάνω>, [tone 'va. no], 'I put him'
 - <ούτε όμπιασε>, ['u.te 'o.bja.se] → <ούτε νόμπιασε>, ['u.te 'no.bja.se], 'nor did it swell'
 - <η αφορμή>, [i a.for.'mi] → <η γιαφορμή> [i ja.for.'mi], 'the occasion'
5. Elision of the final /n/ in the genitive plural:
 - [ton spit.'ʒon] → [to spiθ.'ʒo]
6. Stress on the fourth syllable from the end as opposed to SMG where the so-called 'law of three syllables' demands that no word carries a stress beyond the third syllable from the end.
 - [ef. 'ta.ksa.me.ne]
 - [e.'fi.ɣa.me.ne]
7. Development of prothetic /a/ or /o/.
 - [a.mo.na.'xos]
 - [oɣ.'li.ɣo.ra]

Morphological level

1. Use of different article forms than SMG.
 - <τση>, *the*.GEN.FEM.SG, [tsi] instead of SMG <της>, [tis]
 - <τσι>, *the*.ACC.F.PL, [tsi] instead of SMG <τις>, [tis]
 - <τσου>, *the*.ACC.MASC.PL, [tsi] instead of SMG <τους>, [tus]
2. Inflection suffix <-ομε> instead of SMG <-ουμε> in the first person plural of verbs in active voice:
 - <έχομε>, [e.xo.me], instead of SMG <έχουμε>, [e.xu.me], 'we have'
 - <κάνομε>, [ka.no.me], instead of SMG <κάνουμε>, [ka.nu.me], 'we do'

3. Several masculine nouns in <-ος> are used as neuter nouns:

- <το λαός>, [to la.'os].neuter, instead of SMG <ο λαός>, [o la.'os].masc, 'the people'
- <το πλούτος>, [to 'plu.tos].neuter, instead of SMG <ο πλούτος>, [o 'plu.tos].masc, 'the wealth'

4. Extension of forms of demonstrative pronouns:

- <τουτοσές>, [tu.to.'ses], instead of SMG <τούτος>, [tu.tos], 'this.NOM.MASC.SING'
- <εχειοσές>, [e.cio.'ses], instead of SMG <εχέινος>, [e.'ci.nos], 'that.NOM.MASC.SING'

5. Verbs ending in <-εύω> instead of <-εύω>:

- <χορεύω>, [xo.'re.vɣo] instead of SMG <χορεύω>, [xo.'re.vo], 'I dance'

6. In both SMG and Cretan, the future tense is expressed periphrastically. In contrast to SMG, which employs one auxiliary element, Cretan uses two: the subordinating conjunction <να> and the verb <θέλει>. The verb <θέλει> can appear in two forms: either in its indeclinable form, which is considered the infinitive form of <θέλω> ('I want'), or in finite form, but only for the singular (Chairetakis, 2020), e.g.,

- infinitive form: <να πας θέλει>, [na 'pas 'θe.ɫi], 'You will go'
- finite form: <να πας θες> [na 'pas 'θes], 'You will go'
- instead of SMG <θα πας>, [θa 'pas], 'You will go'

7. The use of <ξ>, [ks] as a perfective aspect marker:

- <τραγούδηξα>, [tra.'ɣu.ði.ksa] instead of SMG <τραγούδησα>, [tra.'ɣu.ði.sa], 'I sang'

Lexicological level In the Cretan dialect, a wealth of words is attested that are not found in SMG. Most of these words are loanwords from Turkish and Venetian. The influence of each of these languages on the Cretan dialect spans four

centuries, with Turkish linguistic influences being comparatively stronger due to the fact that the Turkish conquest was more recent. These loanwords are frequently used to name objects and processes related to the material culture of the people.

- <θιαμπόλι>, [θɕa.'bo.ɫi], 'cretan flute' <italic <fiabuolo>
- <ντελικανής> [de.ɫi.ka.'ɲis], 'the young man' <turkic <delikanli>

Some Cretan word forms are used in SMG with a different meaning.

- <κουράδι>, [ku.'ra.ði] 'flock of sheep' instead of SMG 'faeces'
- <ξανοίγω>, [ksa.'ni.ɣo], 'to see' instead of SMG 'fade out (for a colour)'

Finally, the Cretan dialect also attests to stereotypical expressions not found in SMG.

- <μια ολιό>, [mja o.'ɫa], Lit. one sip, 'a little'
- <δίδω των αμμαθιώ μου>, ['ði.ðo ton a.ma.'θɕo mu], Lit. I give to my eyes, 'I flee upset'

Syntactic level The weak pronouns that are functioning as objects are placed after the verb in contrast with the SMG that places them before the verb:

- <ρωτώ σε>, [ro.'to se] instead of SMG <σε ρωτώ>, [se ro.'to], 'I ask you'

Many verbs take objects in genitive case; in SMG the same verbs take objects in the accusative case:

- <ζηλεύω σου>, [zi.'le.vɣo su] instead of SMG <σε ζηλεύω>, [se zi.'le.vo], 'I envy you'

3 Resources

For the compilation of this corpus, we collected 32 tapes containing material from radio broadcasts in digital format, with permission from the Audiovisual Department of the Vikelaia Municipal Library of Heraklion, Crete. The broadcasts were

recorded and aired by Radio Mires, in the Mes-sara region of Heraklion, during the period 1998-2001, totaling 958 minutes and 47 seconds. The recordings primarily consist of narratives by one speaker, Ioannis Anagnostakis, who is responsible for their composition. The material belongs to the Eastern Cretan dialect group. In terms of textual genre, the linguistic content of the broadcasts consists of folklore narratives. Out of the total volume of material collected, we utilized nine tapes. Criteria for material selection were digital clarity of speech and the representative sampling among the entire three-year period of radio recordings.

For the transcription of the recorded speech to text, the Whisper large-v2 model was utilized. At the time this process was carried out (April 2023), Whisper large-v2 returned the best results to small trials with the Cretan data. The transcriptions were edited by a linguist who is native speaker of the Eastern Cretan dialect. Given that the Cretan dialect is primarily an oral language variety, there is no standardized orthography. The general trend in the orthographic representation of Cretan is conformity with that of Standard Modern Greek. We followed that trend, in an effort to strike a balance among facilitating knowledge transfer from GUD, representing the linguistic characteristics of the dialect in the orthography and aligning with the dominant orthographic trends adopted by the dialect's native speakers. The handling of the distinctive phonological phenomena of the Cretan language variety, such as the frequent insertion of euphonic sounds and the occurrences of voicing, will be discussed below.

4 The treebank

The annotation of East Cretan has relied on the UD annotation guidelines for GUD.³ Only deviations and new constructs and forms have been documented in the guidelines for the East Cretan treebank that are listed as comments of the GUD guidelines.

4.1 Morphology

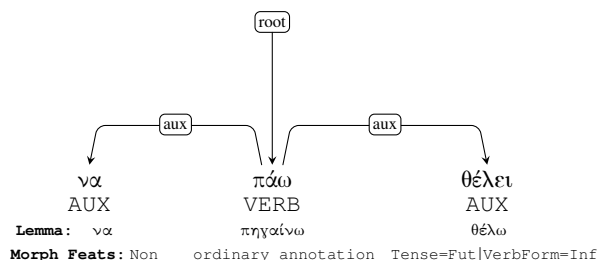
1. For the case of nouns and adjectives, which form diminutives and augmentatives, it was decided to list the basic word as the lemma, mean-

³https://github.com/UniversalDependencies/docs/tree/pages-source/_el

ing the word without the diminutive or augmentative suffix, e.g., <μπεγλιρ-άκι>, 'the little horse' has been assigned the lemma 'μπεγλιρ' that does not contain the diminutive suffix <-άκι>.

2. As mentioned in Section 2, Morphological level 6, the (Eastern) Cretan dialect uses a distinctive periphrastic structure for the future tense, which is not found in SMG. We annotated these structures as follows:

- <να πάω θέλει>, [na 'pa.o 'θe.ði], 'I will go'



3. The perfect tense is expressed, in addition to the usual SMG way, with the following structure: auxiliary verb έχω 'have' + passive participle (Chairetakakis, 2020):

- το 'χει λεομένο
(1) it.ACC has said.PARTICIPLE.ACC
'He/She has said it'

4. All words of the Cretan dialect that appear slightly different from their SMG counterparts were assigned a lemma form that bears the dialectal linguistic characteristics:

- <βρίχνω>, ['vri.xno] instead of SMG <βρίσκω>, ['vri.sko], 'I find'

4.2 Syntax

Because of the oral nature of the collected linguistic material, we encountered many elliptical sentences in the corpus. Copulas were often omitted as well as verb heads: in the example below the subject φτωχός is promoted as the head of the sentence.

- ο φτωχός μια φουρνιά κουτσούβελα
(2) the poor.man a bunch kids
'the poor man had a bunch of kids'

According to the UD guidelines, the non-promoted dependents (here: <φουρνιά>) are con-

nected with the promoted one using the special relation “orphan”.

4.3 Voicing and Euphonics

Both voicing and euphonics are phenomena due to the phonetic environment but with no effect on the syntax and meaning of an utterance. In the Cretan treebank they are annotated separately.

Voicing in MG is a phonological phenomenon where, given the sequence of two words, the initial unvoiced consonant (/ts/, /t/, /p/, /k/) of the second word is voiced, e.g., /tsi/→/dzi/, /t/→/d/, /p/→/b/, /k/→/g/.

In contrast, euphonics are sounds that are added with the phonological procedure of epenthesis, in order to avoid the hiatus produced by vowel sequences, e.g., /'u.te 'om.bja.se/ → /'u.te 'n om.bja.se/ or sequences of consonants, e.g., /'an 'θe.ʎi/ > /'an e 'θe.ʎi/. In all cases, the result of the epenthesis are two open syllables of the type consonant+vowel.

Below, we first discuss the phenomena briefly and then we make a proposal for their representation in the Eastern Cretan UD treebank.

4.3.1 Euphonics in the East Cretan UD treebank

Euphonics are vowels or consonants that occur within a word or between words (3, 4) or at the end of a word (5). In Cretan (and Modern Greek in general) their function is to create open syllables and eliminate hiatuses. For instance, in Eastern Cretan, the ‘γι’ euphonic is used within phonological words as a hiatus breaker, so the condition for its occurrence is the particular hiatus and the existence of a (phonological) word (Kappa, 2014).

- | | | |
|-----|---------------------|---------------------------------|
| | οι | γι-άλλοι |
| (3) | <i>the.NOM.M.PL</i> | <i>EUPH-other.NOM.M.PL</i> |
| | ‘the others’ | |
| | ούτε | ν-όμπιασε |
| (4) | <i>nor.CCONJ</i> | <i>EUPH-swell.PERF.3SG.PAST</i> |
| | ‘not did it swell’ | |
| | χάιν’ | τον-ε |
| (5) | <i>do.3SG.IMP</i> | <i>I.PRON.ACC.M.3SG-EUPH</i> |
| | ‘do it’ | |

The textual encoding of euphonics is an issue. In SMG orthography, the euphonic ‘e’ is attached to the preceding word (5). We had to define additional guidelines for Cretan. We did not encode them as orthographic words because they are sin-

gle sounds and have no morphosyntactic impact on the utterance. In all cases, we have attached euphonics to the word that precedes or follows them, on the condition that open syllables are created:

- παιδιών-ε, *child.PL.GEN-EUPH*
- τον-ε, *I.PRON.ACC.M.3SG-EUPH*
- αν-ε, *if-EUPH*
- γι-άλλοι, *EUPH-other.NOM.M.PL.*

The euphonic ‘γι’ (3) is encoded with two characters because the Greek alphabet does not have a dedicated character for the sound [j]. We could probably use non-Greek characters for them, for instance in (3) we could use ‘j’. As explained in Section 3, we retain the Greek alphabet, which is also used by the speakers of the dialect.

4.3.2 Voicing in the orthography of SMG

SMG orthography uses the following conventions for encoding voicing; these conventions are adopted by most authors who write in other Greek varieties:

1. A ‘-ν’ /n/ is added to the end of an article with the features *CASE=ACC|GENDER=MASC|FEM* when it is followed by another word whose first consonant is voiced in this context but unvoiced in other contexts.

- | | | |
|-----|-----------------------|----------------------------|
| | την πατρίδα | /’ti ba.’tri.ða/ |
| (6) | <i>the-ACC.FEM.SG</i> | <i>homeland-ACC.FEM.SG</i> |
| | ‘the homeland’ | |

2. In all other [word1 word2] sequences where word2 appears with a voiced first consonant (while occurrences of the word with a non-voiced first consonant are attested in other contexts of the same dialect) and word1 is not independently found with a final ‘-ν’, voicing is represented on word2. In the example below the word ‘μύτη’ is in the nominative case that does not accept a final -ν with this type of nouns.

- | | | |
|-----|-----------------------|------------------------|
| | η μύτη τζη | /i ’mi.ti dzi/ |
| (7) | <i>the-NOM.FEM.SG</i> | <i>nose-NOM.FEM.SG</i> |
| | <i>her-GEN.FEM.SG</i> | |
| | ‘her nose’ | |

The Greek alphabet has no single letter corresponding to the sounds /dz/, /d/ and /b/ so Modern Greek orthography represents them with two characters (τζ, ντ and μπ respectively).

4.3.3 Annotation of euphonics and voicing in the Eastern Cretan treebank

We use the MSeg|MGloss representation and the label *euphonic* for annotating euphonics in the Cretan treebank. With the MSeg annotation schema we are able to isolate the euphonic segments from the rest of the word and handle each part as a separate token.

γιάλλοι DET
MSeg=γι-άλλοι|MGloss=euphonic-others

We cannot resort to the MSeg|MGloss representation in order to annotate voicing because the results of voicing cannot be separated from the rest of the word, e.g., in the form of an affix. For instance, ‘τζη’ (/dzi/) cannot be divided as ‘τζ-η’ because ‘-η’ (/i/) is not a word with the same morphosyntactic features as ‘τζη’ (recall that voicing has no syntactic or semantic effect). Instead, we define a feature that differentiates the unvoiced form from the voiced one. This is a new MISC feature of the Cretan treebank called *Voicing* with values *Voiced* and *Unvoiced*.

τζη PRON ... Case=Gen ...
Voicing=Voiced

Voicing characterises all MG dialects, including SMG, in the environment of the Accusative case and contributes to the distinction between Accusative and Nominative case. We do not annotate this type of expected voicing. However, sometimes the voiced version of a word is also used in environments where no voicing is expected, suggesting that the voiced version is lexicalised and co-exists and competes with the unvoiced one, e.g., (dialect of the island of Lemnos) ‘η μπα-τρίδα’ (/i ba.ʈri.ða/) coexists with ‘η πατρίδα’ (/i pa.ʈri.ða/), both in the nominative case, singular number. The question is which lemma should be assigned to each of the two versions. We assign the unvoiced version of the lemma to both versions; in addition, the voiced form is assigned the feature-value pair *Voicing=Voiced*. Our choice of the unvoiced version contributes to the consistency of the annotation and to knowledge transfer from SMG to the dialects because SMG usually has the unvoiced version of the lemma, if it has this lemma at all.

In the example *Ψυχοπόνεσέ ντονε το πα-παδόκι*, Lit. felt.sorry him the altar.boy, ‘The altar boy felt sorry for him’ both unexpected voicing and euphonics are used because the verb form

‘Ψυχοπόνεσε’ never appears with a final -ν:

ντονε Voicing=Voiced|MSeg=ντον-ε
|MGloss=him-euphonic

5 Active annotation

To annotate the Cretan treebank we used active annotation (Vlachos, 2006) implemented in 6 iterative cycles. The first set of 40 unlabelled Cretan samples was annotated with a model trained on GUD, which represents SMG. In each cycle, the annotator edited 40 samples from the output, split in 30 for the training set and 10 for the development set, added them to the existing training and development sets and the model was retrained on the revised data. For the test set, 30 manually annotated samples were used. All samples were randomly selected, with the only criterion being that each sample contained more than five tokens to avoid sentences with minimal linguistic information.

| | | 1st | 2nd | 3rd | 4th | 5th | 6th |
|-----------|-------|-----|-----|------|------|------|------|
| Sentences | Train | 30 | 60 | 90 | 120 | 150 | 180 |
| | Dev | 10 | 20 | 30 | 40 | 50 | 60 |
| | Test | 30 | 30 | 30 | 30 | 30 | 30 |
| Tokens | Train | 448 | 903 | 1395 | 1880 | 2398 | 2976 |
| | Dev | 175 | 348 | 504 | 728 | 939 | 1129 |
| | Test | 523 | 523 | 523 | 523 | 523 | 523 |

Table 1: East Cretan sentences and tokens per round.

During this first attempt to develop a UD treebank of Cretan, the annotation guidelines were developed as research progressed. Any revisions to the annotation guidelines were implemented across the entire training, development and test sets.

6 Including euphonics and voicing in the models

To introduce euphonics in the model, we process the input CoNNLU representations of sentences by transferring information from the MSeg annotation (column 10 of the CoNNLU format) on the LEMMA, UPOS and XPOS columns and training the model on the modified treebank⁴. The original word’s UPOS and DEPREL tags are inherited by the piece of the token that remains after the euphonic is removed (the ‘original word’) and the

⁴The script for the transformation of the input CONLLU files can be found at <https://anonymous.4open.science/r/euphonics-7F98>

euphonic is represented as a separate token with the new XPOS/UPOS tag EUPH that depends on the original word with the new dependency relation ‘euph’. The XPOS tag EUPH and the dependency ‘euph’ have been defined for the purposes of the Eastern Cretan treebank and are used in ongoing work on other varieties of Modern Greek, including SMG. The tag EUPH was introduced to the UPOS column to satisfy a requirement of the processing tool.

We did not use the UPOS X because euphonics can hardly be called words, at least in the sense of self-standing linguistic entities that combine a form with some type of semantic contribution. But even if euphonics were considered a type of word, again the X UPOS would not be a choice because euphonics are clearly parts of the language varieties we study and play a well defined role. These two facts contrast with the UD annotation guidelines about UPOS X: “(UPOS) X is discouraged for words that clearly belong to the language, even if they are idiosyncratic in form or distribution and thus do not neatly fit into other syntactic categories.” Neither did we use the UPOS PART(icle). UD define particles as “function words that must be associated with another word or phrase to impart meaning and that do not satisfy definitions of other universal parts of speech”. Euphonics do not impart any meaning at all. Finally, we did not consider them clitics as suggested by one of our reviewers, because clitics do not define a POS of their own and we have argued that euphonics cannot be assigned any of the POS available in UD.

The output of the model that knows about euphonics cannot be used in the active annotation cycle because its form differs from the form of the UD treebank. This output contains a modified XPOS column (which may not be a problem), no information on the MISC column about voicing and euphonics while the UPOS column is modified with an extra tag. We have not applied active annotation on voicing and euphonics but for future needs, since the phenomena occur in many MG dialects, we will have to post-edit the model’s output and make it comply with the form of the annotation of the input.

A complete example is included below featuring the word <ντονε> that contains the voiced masculine, singular, accusative form of the personal pronoun <εγώ> ‘I’ with the euphonic ‘ε’ /e/ attached to it. Similarly, the feature-value pair

“Voicing=Voiced” is added to the list of morphological features.

```
2 ντονε εγώ PRON Case=Acc...|Gender=Masc|Number=Sing|Person=3|PronType=Prs
1 obj _ Voicing=Voiced|MSeg=ντονε-ε|MGloss=him-euphonic
2-3 ντονε _ _ _ _ _ Voicing=Voiced|MSeg=ντονε-ε|MGloss=him-euphonic
2 τον εγώ PRON _
Case=Acc|Gender=Masc|Number=Sing|Person=3|PronType=Prs|Voicing=Voiced _ 1 obj _
3 ε ε _ EUPH _ 2 euph _ _
```

7 Models

For the experiments we used the open source Stanza package (Qi et al., 2020). The embeddings for all experiments were generated by combining the GUD treebank with the Cretan corpus. We used two different settings for the treebanks: GUD plus the Eastern Cretan data (henceforth GUD+Cretan treebank) that increased at each round by 40 samples (30 in the training set and 10 in the development set) and, the Eastern Cretan samples only that increased exactly in parallel with the GUD+Cretan treebank. In both settings, we finetuned the Greek BERT model (Koutsikakis et al., 2020) for the tasks of PoS tagging and dependency parsing.

| Metric | R1 | R2 | R3 | R4 | R5 | R6 |
|---------|-------|-------|-------|-------|-------|-------|
| UPOS | 80.12 | 83.57 | 85.80 | 88.64 | 87.83 | 89.25 |
| XPOS | 79.31 | 78.09 | 80.12 | 82.56 | 82.35 | 83.37 |
| UFeats | 55.38 | 63.49 | 72.82 | 77.08 | 76.47 | 78.70 |
| AllTags | 48.68 | 53.75 | 59.84 | 65.92 | 65.92 | 68.15 |
| Lemmas | 66.53 | 73.02 | 77.28 | 80.12 | 81.74 | 81.34 |
| UAS | 65.31 | 73.02 | 75.25 | 78.09 | 75.25 | 78.50 |
| LAS | 45.84 | 58.22 | 63.29 | 65.92 | 65.52 | 67.75 |
| CLAS | 32.59 | 46.54 | 51.47 | 55.76 | 55.22 | 59.33 |
| MLAS | 10.37 | 20.00 | 30.51 | 36.06 | 33.58 | 40.67 |
| BLEX | 14.81 | 29.23 | 34.19 | 40.15 | 40.67 | 43.28 |
| ELAS | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| EULAS | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |

Table 2: Accuracy scores across rounds: East Cretan treebank. R=Round.

8 Discussion

The results are depicted in Table 2 and 3, and were obtained using the pre-tokenized text option provided by Stanza. Figure 1 shows that the model trained on the GUD+Eastern Cretan treebank consistently outperforms the model trained on the Eastern Cretan treebank across all rounds and tasks. Therefore, GUD was an excellent resource for knowledge transfer from SMG to Eastern Cretan models. This result must have been

| Metric | R1 | R2 | R3 | R4 | R5 | R6 |
|---------|-------|-------|-------|-------|-------|-------|
| UPOS | 89.25 | 92.29 | 92.49 | 92.09 | 92.90 | 92.90 |
| XPOS | 89.25 | 89.45 | 89.66 | 89.45 | 88.84 | 89.45 |
| UFeats | 83.77 | 85.40 | 84.99 | 87.22 | 85.60 | 85.60 |
| AllTags | 76.27 | 78.50 | 77.28 | 78.30 | 77.28 | 77.48 |
| Lemmas | 83.98 | 87.42 | 87.83 | 87.42 | 89.05 | 88.44 |
| UAS | 84.58 | 83.98 | 85.40 | 87.02 | 87.02 | 85.40 |
| LAS | 73.83 | 74.85 | 77.08 | 76.88 | 78.50 | 78.30 |
| CLAS | 66.54 | 68.56 | 70.30 | 70.57 | 71.64 | 72.76 |
| MLAS | 51.88 | 55.30 | 57.14 | 56.60 | 55.97 | 57.09 |
| BLEX | 53.76 | 57.58 | 60.90 | 58.87 | 61.19 | 61.57 |
| ELAS | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| EULAS | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |

Table 3: Accuracy scores across rounds: GUD+Cretan treebank. R=Round.

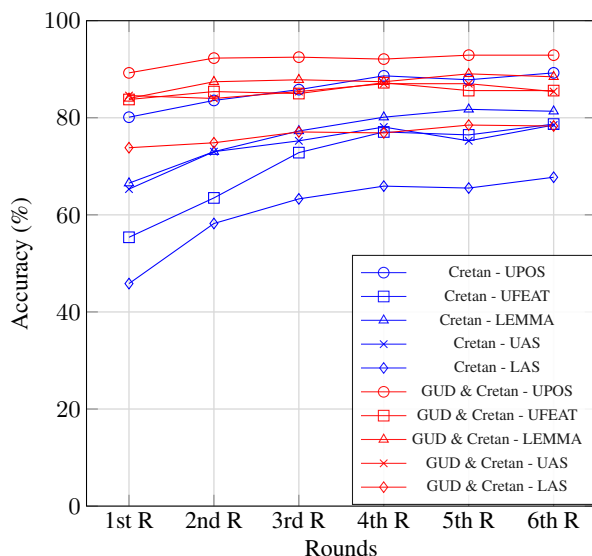


Figure 1: Accuracy scores for the GUD+Cretan & Cretan datasets.

corroborated by the fact that the texts of both language varieties are written with the same orthographic conventions.

After the 4th cycle the GUD+Eastern Cretan models tend to decrease or stabilize across all accuracy measures, while the Cretan-only models still improve. This suggests that after the 4th cycle information from GUD added noise. Therefore, 4 or 5 cycles with GUD were enough for successful knowledge transfer for this variety of Greek and the set up we used (40 new samples at each cycle).

In a 7th training round, we applied on the Eastern Cretan treebank the transformation that introduces euphonics and voicing in the models (see Section 4.3). The results are shown on Table 4. The East Cretan treebank returns still improving results. The test set contained 10 instances of these phenomena and the training and development sets 67 instances. The model achieved a 100% Recall

and Precision, probably because the forms of voicing and euphonics are very distinctive.

| Metric | Accuracy |
|---------|----------|
| UPOS | 89.45 |
| XPOS | 85.27 |
| UFeats | 78.00 |
| AllTags | 68.36 |
| Lemmas | 82.36 |
| UAS | 78.73 |
| LAS | 69.27 |
| CLAS | 59.80 |
| MLAS | 39.86 |
| BLEX | 44.59 |

Table 4: Accuracy scores for the 7th Round that includes EUPHONICS-VOICING. East Cretan treebank.

9 Conclusion

We have developed the first UD treebank of Eastern Cretan, which is a living, non standardised variety of Modern Greek. We have attempted to model phenomena new to UD guidelines such as voicing and euphonics; these phenomena abide in the dialects of Modern Greek. The successful active annotation procedure and the knowledge transfer from the GUD treebank of SMG to the models of Eastern Cretan suggests that a similar pipeline can facilitate the modelling of other varieties of MG, starting from Western Cretan. We hope that this treebank will support future efforts to provide additional digital material from more native speakers, the textual legacy of East Cretan as well as other, linguistically challenging, dialects of Modern Greek.

10 Limitations

A weak point of our approach is that we have relied on data from one speaker only. However, this was the first time that the full pipeline from oral data to annotated UD treebanks was studied for a Greek dialect (here we report on the work after the Speech-to-Text step). We are currently collecting data from more speakers from the same area (the Heraklion prefecture) and aim to enrich the Cretan UD treebank soon. The orthography we used to transcribe the East Cretan oral material is identical to the orthography used for SMG and has probably facilitated knowledge transfer from the treebank of SMG; however, as it has been mentioned,

this is the orthography preferred by many speakers of Cretan (and many other dialects of MG). Future work may try to exploit the existing textual legacy of the Cretan dialect that occasionally adopts an orthography partially different from the orthography of SMG. The exploitation of non-standardised textual legacy, especially for under-resourced language varieties, for model development is a well-known problem (Plank, 2016). These said, we would like to add that we relied a lot on the GUD guidelines in order to develop the Eastern Cretan UD guidelines and, while doing so, we did not have to suppress or alter information particular to this dialect; this may be an indication of the proximity of these two varieties of Modern Greek and of the relatively little bias that SMG exerted on the models of Eastern Cretan.

Acknowledgments

This work has been partially supported by project MIS 5154714 of the National Recovery and Resilience Plan Greece 2.0 funded by the European Union under the NextGenerationEU Program.

References

- Steven Bird. 2020. Decolonising speech and language technology. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3504–3519, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- George Chairetakis. 2020. *The morphology of the Cretan dialect: Inflection and derivation*. Phd thesis, University of Patras.
- Ioanna Kappa. 2014. Epenthetic consonants in the western cretan dialect. In G. Kotzoglou et al., editors, *Selected Papers of the 11th International Conference on Greek Linguistics*, pages 674–688. University of the Aegean, Rhodes.
- Nikolaos G. Kontosopoulos. 1969. *Linguistic and geographic investigations of the Cretan Dialect*, [Γλωσσογεωγραφικά διερευνήσεις εις την Κρητικήν διάλεκτον], 1st ed. edition. Graphic Arts Efstathios Papoulas, Athens.
- Nikolaos G. Kontosopoulos. 2008. *Dialects and idiolects of Modern Greek*, 5th ed. edition. Grigoris, Athens.
- John Koutsikakis, Ilias Chalkidis, Prodromos Malakiotis, and Ion Androutsopoulos. 2020. Greek-bert: The greeks visiting sesame street. In *11th Hellenic Conference on Artificial Intelligence*, SETN 2020, page 110–117, New York, NY, USA. Association for Computing Machinery.
- Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.
- Barbara Plank. 2016. What to do about non-standard (or non-canonical) language in nlp.
- Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.
- Andreas Vlachos. 2006. Active annotation. In *Proceedings of the Workshop on Adaptive Text Extraction and Mining (ATEM 2006)*.

Danoliteracy of Generative Large Language Models

Søren Vejlggaard Holm^{1,2}, Lars Kai Hansen¹, Martin Carsten Nielsen²

¹Technical University of Denmark, Anker Engelunds Vej 1, 2800 Kongens Lyngby, Denmark,

²Alvenir, Applebys Plads 7, 1411 København K, Denmark

Correspondence: swiho@dtu.dk

Abstract

The language technology moonshot moment of Generative Large Language Models (GLLMs) was not limited to English: These models brought a surge of technological applications, investments, and hype to low-resource languages as well. However, the capabilities of these models in languages such as Danish were, until recently, difficult to verify beyond qualitative demonstrations due to a lack of applicable evaluation corpora. We present a GLLM benchmark to evaluate *Danoliteracy*, a measure of Danish language and cultural competency across eight diverse scenarios such as Danish citizenship tests and abstractive social media question answering. This limited-size benchmark was found to produce a robust ranking that correlates to human feedback at $\rho \sim 0.8$ with GPT-4 and Claude Opus models achieving the highest rankings. Analyzing these model results across scenarios, we find one strong underlying factor explaining 95% of scenario performance variance for GLLMs in Danish, suggesting a g factor of model consistency in language adaptation.

1 Introduction

Benchmarks shape technologies. By acting as normative guidelines for technology applications, benchmarks imply directions of research and development that ultimately impact users (Liang et al., 2022). GLLMs specifically have emerged as a technology with near-universal impact, including lower-resource languages such as Danish (Olsen, 2023). If the challenging task of general GLLM evaluation is not extended to low-resource languages, practitioners start from scratch for each model use case, inhibiting practical adoption or possibly resulting in risky, undertested implementations.

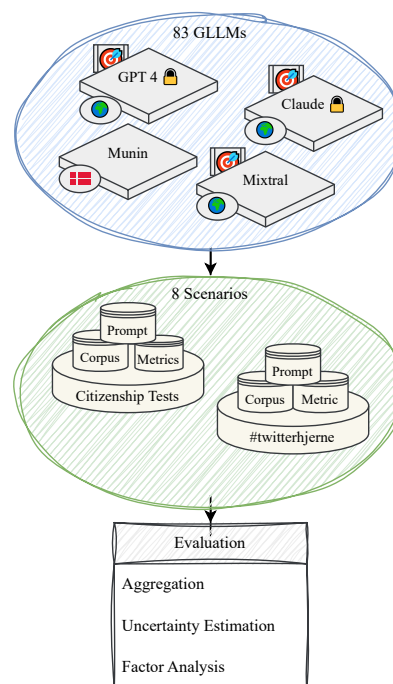


Figure 1: The overall evaluation setup: A collection of GLLMs, including closed-source (lock symbol) instruct-tuned (bulls-eye) and multilingual (globe) ones, were evaluated in Danish across diverse use-case scenarios.

We take up the challenge of creating a GLLM evaluation benchmark for Danish, a North Germanic language spoken by 6 million people, primarily in the Nordic country of Denmark. As depicted in Figure 1, our approach is to create a compilation of small-scale, diverse evaluation scenarios combined into one general benchmark to reveal model Danoliteracy. By Danoliteracy we refer to the level of GLLM real-world knowledge (RWK), natural language understanding (NLU), and natural language generation (NLG) in Danish.

This paper presents the resulting *Danoliterate Benchmark*, describing evaluation methods, datasets and model results. We analyze these results with the goals of validating and exploring evaluation methodology. An important part of this analysis is to investigate the feasibility of such evaluation: Does this small-scale, language-specific approach achieve a significant ranking of GLLMs? Even if a non-spurious leaderboard can be discerned from the result, it is not enough to validate the benchmark which might actually show something orthogonal to Danoliteracy. Thus, as a benchmark validation tool, we additionally present a user survey, collecting the preferences of Danish speakers when interacting with hidden pairs of GLLMs in an arena A/B test setup.

The availability of a suite of meaningful benchmark results allows us to investigate GLLM behavior: Initially, we explore which specific models are most Danoliterate and how different types of GLLMs compare. Beyond that, we are particularly interested in capability consistency across tasks: If a GLLM performs strongly in one Danish use-case scenario, does this performance generalize to other Danish scenarios across different domains and objectives?

We hope so. If the answer is no, practitioners are without general results to trust, requiring a full model re-evaluation for each downstream use. However, if capability consistency is present, we should be able to find a single underlying axis that correlates with performance across diverse scenarios. Such a general dimension of Danoliteracy can be compared to the g factor of general human intelligence (Spearman, 1904). If one significant, main factor is found, it implies a level of stability that can help guide the expectations of practitioners across GLLM implementations in varying and even novel Danish tasks.

The contributions presented in this paper can be summarized as follows:

- An open-source benchmark for GLLMs in low-resource languages with an evaluation framework and a live leaderboard site.
- The release of a set of novel evaluation datasets for Danish.
- Evidence that GPT-4 and Claude Opus models are currently uniquely capable in Danish, outperforming other closed models which in turn overcome open-weights models.

- Evidence suggesting the existence of a Danoliteracy g factor in GLLMs supported by preliminary results from our open-source human feedback study.

2 Related Work

2.1 GLLM Evaluation

The hard task of evaluating free-generation, multitask models has been attempted in many ways. Liang et al. define an empirical approach for revealing model behaviour: Evaluate each model on a compilation of many scenarios and use-cases of interest, spanning different languages, domains and task categories – ideally across multiple performance dimensions in addition to raw model capability such as efficiency, bias, and robustness (Liang et al., 2022).

This scenario compilation approach has been applied in many ways to GLLMs: The HELM Lite benchmark presents evaluations of GLLMs on question answering (QA) and translation tasks (Liang et al., 2023). Influential benchmarks include the Huggingface OpenLLM Leaderboard (Beeching et al., 2023) and other implementations of the knowledge-based scenarios MMLU (Hendrycks et al., 2021) and HellaSwag (Zellers et al., 2019).

These benchmarks mainly use comparison or similarity algorithms to parse model answers e.g. for finding a chosen option for multiple-choice QA. Other approaches include applying other GLLMs to grade generations (Zheng et al., 2023) (OpenAI, 2023) or using human feedback (Chiang et al., 2024).

2.2 Low-resource NLP Evaluation

Most broadly reported GLLM evaluations are only or primarily performed on examples in English. Approaches to evaluate lesser-resourced languages include both attempts to compile massively multilingual benchmarks either by automatic translation (Lai et al., 2023) or dataset curation (Ahuja et al., 2023).

Other approaches focus on one language exclusively in attempts to evaluate GLLM language performance beyond surface-level lexical or syntactical literacy. Using this method, practitioners can align scenario domain, cultural content, and real-world facts with the setting of the language, though a lack of relevant data can be problematic (Liu et al., 2023).

Specifically in Danish, the comprehensive Scan-dEval benchmark, which packages scenarios across eight languages divided into NLU and NLG leaderboards, implements evaluation on GLLMs in Danish on eight NLG scenarios with some overlap in dataset sources with this work (Nielsen, 2023).

2.3 GLLM g Factor

The idea that GLLM performance is strongly correlated across tasks has been noted previously by for example Ilić who carried out factor analysis on the Open LLM Leaderboard and GLUE, (Wang et al., 2018) obtaining results similar to ours (Ilić, 2023).

3 Methods

3.1 Datasets

The eight scenario datasets are divided into three broad categories: Scenarios testing RWK, scenarios requiring models to perform free NLG and those that imply solving classical NLU tasks.

Real-world Knowledge

1. **Citizenship Test** is a novel dataset of 605 multiple-choice questions acquired from governmental tests that require applicants for Danish citizenship to demonstrate familiarity with national societal structure, culture, and history (siri.dk, 2023).
2. **HyggeSwag** is a novel manual translation¹ of 125 HellaSwag (Zellers et al., 2019) ActivityNet (Caba Heilbron et al., 2015) questions testing commonsense natural language inference as a multiple-choice task to pick the only completion consistent with real-world logic.
3. **Gym 2000** is a small, novel extraction of 50 literature comprehension multiple-choice questions from the Danish Centre for Reading Research (CRR) aimed at high-schoolers (Arnbak and Elbro, 2000).

Free NLG

- 4 **#twitterhjerne** is a novel abstractive question-answering dataset containing 78 anonymized question tweets from the Danish hashtag of that name, translated to *Twitter Brain*, where users ask the social media hive mind for help, input or recommendations. For each question

¹The text was translated by the authors with each translation being validated by another author completing the inference task.

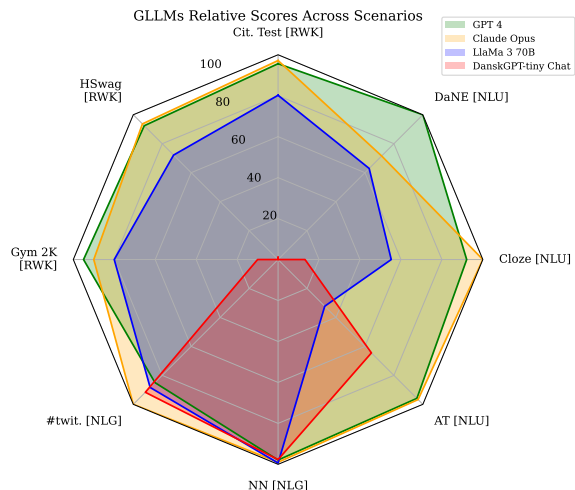


Figure 2: Selected model normalized results across the eight scenarios divided into three categories as described in Section 3.1. Claude Opus is overtaken by GPT-4 on the NER task but wins on an NLG task. LLaMa 3 70B, the SOTA open-weights model, lags behind on NLU and knowledge-based tasks. A Danish-specialized model with only 1.1B parameters, DanskGPT-tiny Chat, benchmarks well in NLG but fails on knowledge and understanding.

tweet, 2-9 reference answer tweets were extracted making it possible to use the $\text{score}_{\text{olo}}$ metric (1).

- 5 **Nordjylland News** is an existing news summarization dataset (Kinch, 2023) from which a subset of 300 short news articles with corresponding summaries were used.

NLU Tasks

- 6 **Cloze Self Test** is another small, novel extraction from CRR materials (Jensen et al., 2015), this one containing 33 cloze-style questions evaluated as multiple-choice selection.
- 7 **DaNE** is an existing canonical Danish NER dataset with four entity categories (Hvingelby et al., 2020) from which a subset of 256 examples were used.
- 8 **Angry Tweets** is an existing sentiment classification dataset (Brogaard Pauli et al., 2021, Sec. 4) with three sentiment categories from which 256 examples were used for multiple-choice prompts.

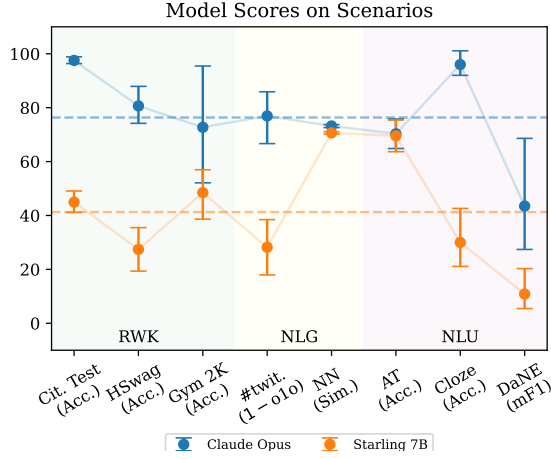


Figure 3: The non-normalized metric scores across evaluation scenarios for two models that were judged highly according to human feedback. Uncertainties are 95% confidence intervals according to the bootstrapping procedure and the micro-average is displayed for each model.

All datasets are released on the Huggingface Datasets Hub with dataset cards² except for the two small datasets extracted from CRR which require practitioners to re-run data collection for personal use. More details on dataset licensing and collection as well as data examples can be found in Appendix B.

3.2 Evaluation

Each evaluation scenario consisted of a dataset, a prompt template, and a chosen metric.

Most of the available datasets allowed primarily for testing discriminative RWK and NLU of the GLLM by requiring it to select between multiple-choice answers. For these multiple-choice scenarios, frequency of generating the correct option number was reported as model accuracy.

Two metrics were used for NLG. First, summarization was implemented using a similarity score between model summary and a reference summary $s(\mathcal{T}_{\mathcal{M}}, \mathcal{T}_{\text{ref}})$. Secondly, we implemented abstractive question answering tasks for the specific type of dataset D where each question has not just one correct answer but a corresponding set of reference, human-generated answers. This was done by scoring GLLMs using the frequency with which generated answers were the odd-one-out, defined by the lowest total similarity to all possible answers

²Datasets can be found on danoliterate.compute.dtu.dk/Scenarios

$T = \{\mathcal{T}_{\mathcal{M}}\} \cup \{\mathcal{T}_{\text{ref}, i}\}_{i=1..k}$ as shown in Eq. 1. For similarity scores s , the BERT score algorithm (Zhang* et al., 2020) based on the DFM Encoder Large (Enevoldsen et al., 2022) was used.

$$\text{score}_{\text{o1o}} = \mathbb{P}_D \left[\mathcal{T}_{\mathcal{M}} = \underset{t_1 \in T}{\text{argmin}} \sum_{t_2 \in T} s(t_1, t_2) \right] \quad (1)$$

Finally, few-shot named entity recognition (NER) was implemented for GLLMs using 3-shot prompting and the GPT-NER multiple queries idea (Wang et al., 2023). Here, word-level entity class predictions were aligned and the standard NER micro-average F1 scores were calculated using the SeqEval framework (Nakayama, 2018).

Scenarios were operationalized by prompting GLLMs in the scenario language, Danish, and structuring prompts with headers marked with the # character as in Markdown. In order to use the same prompts for instruct-tuned and base GLLMs, prompts started with the instruction and ended with a text leading towards an answer in the continuation as shown in Figure 4. Prompting and metric implementation details are covered in Appendix A.

```

1 Write a one-sentence summary of the
  text.
2 # TEXT
3 Lorem ipsum dolor sit amet ...
4 # SUMMARY
5 A summary of the text could be:

```

Figure 4: The general prompting approach translated to English.

3.3 Models

Both local prediction of open-weights models and API access to externally hosted GLLMs were implemented. 54 autoregressive, decoder language models trained for general text generation were included. Models were tested if we saw any reason to suspect a degree of Danoliteracy, thus including multilingual models with possibly small amounts of Danish training data as well as other Mainland Scandinavian monolingual models but excluding strictly English-only models. Both base and instruct-tuned models were evaluated. All model generation was performed with greedy decoding and with a maximum number of generated model tokens of 256. OpenAI’s o1 model was allowed to generate internal tokens freely. The models run

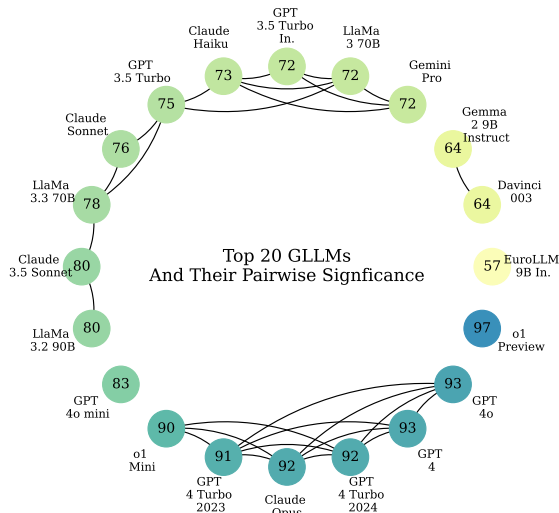


Figure 5: Model Danoliteracy Index across all scenarios for top performers. Two model nodes are connected iff the bootstrapping procedure could not reveal significant benchmark performance difference at $\alpha = 0.05$. Together with the special o1 model, Claude Opus and the GPT 4 family models are consistent winners.

locally ranged in sizes from 124M parameters to 13B parameters, resulting in a total project GPU use of ~ 100 hours on a single Nvidia H100.

3.4 The Danoliterate Framework

A modular, open-source evaluation framework was implemented in Python, using Huggingface Transformers (Wolf et al., 2019) and Datasets (Lhoest et al., 2021) as central tools as well as Hydra (Yadan, 2019) and a Weights and Biases-integration (Biewald, 2020) for structuring experimentation. This framework, danoliterate, is released on GitHub³ under the MIT License.

Furthermore, an interactive site displaying the leaderboard as well as other benchmark results and examples was produced using the Streamlit framework. See Figure 6 for a screenshot of this frontend and Appendix A.4 for versions of software dependencies.

3.5 Human Feedback

For a subset of 18 instruct-tuned models, we have set up a parallel study to collect human judgment on model performance. Volunteers were presented with a anonymized pair of models and were asked

³github.com/sorenmulli/danoliterate

| | 🔍 | 📄 | 📊 | 🏆 | Citizenship Test |
|-------------------------|---|---|------|----|------------------|
| OpenAI Davinci 003 | ☑ | ☑ | | 65 | 69±2 |
| Mixtral (@ Groq) | ☑ | ☐ | 46.7 | 52 | 61±2 |
| SOLAR 10.7B Instruct | ☑ | ☐ | 10.7 | 51 | 66±2 |
| Qwen1.5 7B chat | ☑ | ☐ | 7.7 | 50 | 55±2 |
| Starling-LM-7B-beta | ☑ | ☐ | 7.2 | 43 | 45±2 |
| Heidrun Mistral 7B Chat | ☑ | ☐ | 7.2 | 42 | 54±2 |
| LLaMa 2 (@ Groq) | ☑ | ☐ | 69.0 | 41 | 54±2 |

Figure 6: A screenshot from the leaderboard frontend allowing users to explore how model results change with different metric choices as well as inspecting model output examples and reading further details on evaluation scenarios.

to report their preferred model. This was done based on side-by-side model answers on at least three prompts selected by the volunteer from a pool of 100 prompt examples. Prompt selection was chosen independently of the Danoliterate Benchmark by creating one Danish prompt for each of 100 popular generative AI use-cases according to Zao-Sanders (Zao-Sanders, 2024). The study is ongoing: At the time of writing, 477 responses were analyzed. More details on data collection and analysis can be seen in Appendix C.

4 Results

4.1 Benchmark Feasibility

Benchmarks must have a sufficiently clear signal to be useful. The ranking in the final leaderboard should be determined by meaningful model differences and influenced minimally by sampling noise.

To quantify benchmark noise, we implemented blocked bootstrapping, resampling all examples with replacement and aggregating all $N = 8$ scenario scores for each of the $M = 83$ models. For each $K = 10,000$ bootstrap samples, the $M \times N$ model scenario results were aggregated into one overall Danoliteracy Index for each model d_M . This index was computed by considering one scenario at a time, assigning to the winner index 100 and to the lowest-scoring model index 0 with a linear scaling between the two. The micro-average across the N scenarios is reported as the resulting Danoliteracy Index of the sample.

The median index is presented as the main leaderboard of this report⁴ with top 20 shown in

⁴The full $M \times N$ version can be seen at the live leaderboard site: danoliterate.compute.dtu.dk/Leaderboard.

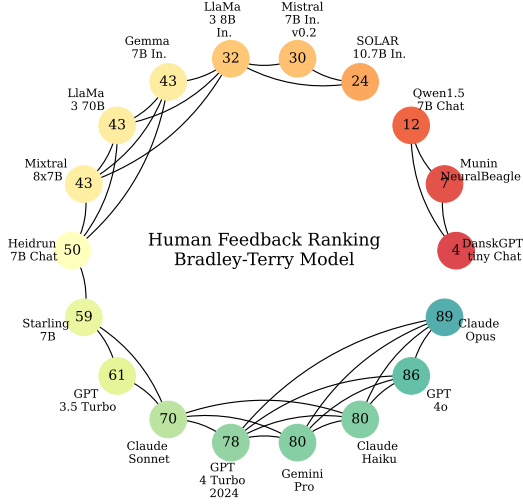


Figure 7: Model Danish capabilities based on human feedback. Values are normalized Bradley-Terry coefficients where two models are connected if the coefficients are not significantly different in the ranking model described in Appendix C.3.

Figure 5. For these, pairwise model index comparisons were performed using the bootstrap samples, correcting p values to control the false discovery rate across $\frac{1}{2}M \times (M - 1)$ comparisons (Benjamini and Hochberg, 1995), presenting significant differences at $\alpha = 5\%$.

The results show groups of similarly performing models whose Danoliteracy cannot be distinguished. This increases the lower you go: Many mediocre models, especially non instruct-tuned models, get $d_M \sim 20$: As an example, this small-sample size, curated benchmark cannot reveal a difference between the base models LLaMa 2 7B, $d_{L2} = 20$, and LLaMa 3 8B, $d_{L3} = 23$.

However, robust separation is visible for some models, providing basis for statements like ”The different GPT-4 models benchmark at the same level but clearly perform better than GPT-3 models” or ”The bigger the Claude 3, the better the performance – but even the cheap Haiku version performs at GPT-3.5 level” or ”Small, Danish-specialized models like Heidrun can perform at LLaMa 2 70B level but LLaMa 3 has moved the SOTA for open-weights models in Danish”. This signal allows us to learn more about reasons for model performance which we explore in the next section.

First, we turn to the important question of validity: We see a robust benchmark signal resulting in

a significant ranking but must question the meaning of the signal. One superficial indication of a meaningful signal is that, as expected, the ranking correlates significantly with model parameter counts⁵ $\rho \sim 0.6$. However, more importantly: We find that it does correlate with the preliminary results of our Danish human judgement survey.

Ranking human judgement using the Bradley-Terry model as in (Chiang et al., 2024, Sec. 4), we achieve a ranking shown in Figure 7. We observe meaningful differences compared to the Danoliteracy Index: For example, Claude models are more competitive against GPT 4 and the title as best included open-weights model is taken by Nexusflow Starling (Zhu et al., 2023) from LLaMa 3 70B. Crucially, however, the general ranking is similar, resulting in a correlation⁶ of $\rho \sim 0.8$ with the Danoliteracy Index for these 18 judged models. We note this as a high value. As a comparison, the Danoliteracy index has a weaker correlation with English benchmarks like HELM Lite and the Open LLM Leaderboard⁷, $\rho \sim 0.5$.

Thus, the results from our monolingual scenario compilation approach differ from those from English benchmarks while importantly, showing high correspondence to judgments made by Danish speakers.

4.2 Model Outcomes

The leading models are familiar, proprietary top products. Though LLaMa 3 reaches GPT 3.5 level, Figure 8 shows that most models capable in Danish do not have openly available weights. Furthermore, these top performers are generally also large and instruct-tuned: Quantitatively, models get about ~ 0.5 further Danoliteracy Index points per additional billion parameters and around ~ 15 from instruct-tuning⁸.

The substantial requirements of dataset and model scale as well as creation of instruct datasets might explain why nationally anchored organizations have not been able to come up with Danish-first models competing with the multilingual behemoths. Such multilingual models have the advan-

⁵Here, only 38 open models with known parameter counts were considered. $\alpha = 5\%$ confidence interval: $[0.37; 0.78]$.

⁶ $\alpha = 5\%$ confidence interval: $[0.6; 0.9]$

⁷This is based on 12 models that overlap between this benchmark and HELM Lite ($\alpha = 5\%$ confidence interval: $[-0.2; 0.8]$) and the 15 that overlap with the Open LLM Leaderboard ($\alpha = 5\%$ confidence interval: $[0.; 0.8]$).

⁸From fitting a naïve linear model on the results including only models with known parameter counts, see Appendix D.1

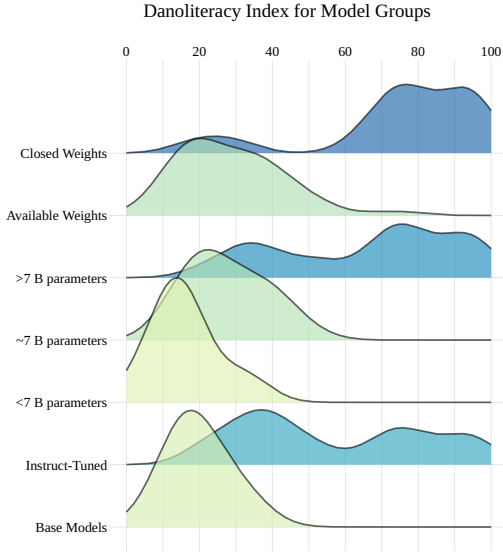


Figure 8: Model Danoliteracy Index for groups of models. For the models tested in Danish, closed weights dominate open-weights in a remarkably clear way. Bigger models are better and on this benchmark, instruct-tuning is necessary to achieve high benchmark scores.

tage of linguistic and factual knowledge enhancement across training languages which also benefit them in the monolingual setting.

4.3 Capability Dimensionality

The previous analysis primarily considered the aggregated benchmark results across scenarios. What is going on at the scenario level? While different model capability profiles can be seen, exemplified in Figure 2 and Figure 3, the main first impression is that model performance at one benchmark scenario strongly predicts performance at other scenarios: One principal component explains 75% of the model result variance across the eight scenarios.

This finding leads us to the conclusion that a "general factor of Danoliteracy" exists. We investigate this further using Exploratory Factor Analysis (EFA) on the $M \times N$ scenario result matrix, analyzing the underlying result dimensionality: How many factors are needed to explain the variance induced by model results over the N scenarios?

This analysis, further detailed in Appendix D.2, shows a sharp drop in factor eigenvalue when moving from one to two factors as shown in Figure 9.

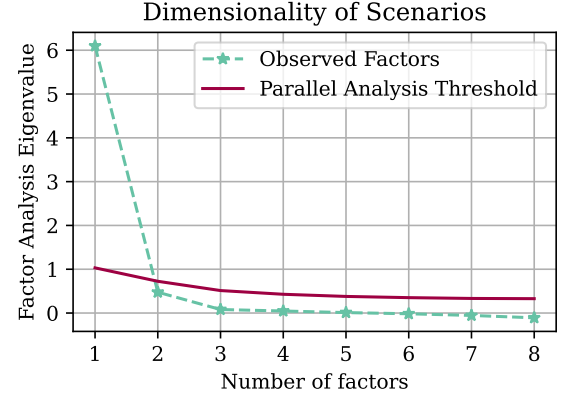


Figure 9: Factor Analysis on model results across eight scenarios reveal one underlying dimension of Danoliteracy deemed significant by Horn’s Parallel Analysis.

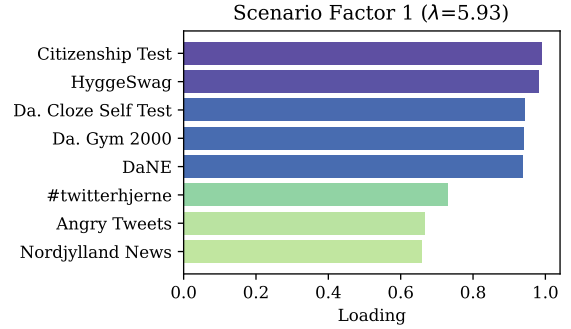


Figure 10: The scenarios most contributing to the underlying signal of Danoliteracy: The factual multiple-choice evaluation scenario containing Danish citizenship tests most strongly explains benchmark performance.

According to both Horn’s Parallel Analysis (Horn, 1965) and the Kaiser Criterion requiring relevant factors to have $\lambda > 1$ (Kaiser, 1960), the resulting number of significant factors is 1. Loadings for this factor is shown in Figure 10 suggesting that the RWK scenarios of HyggeSwag and, importantly, the Danish culturally aligned Citizenship Test scenario explain the largest part of the dynamics of the model results.

GLLM capability being consistent across different tasks is not just suggested by this benchmark: Carrying out the same analysis for Scandinavian and English scenario compilations shows an underlying benchmark dimensionality much lower than scenario count, with significant factor count close to 1 as seen in Table 1.

| Benchmark | D | σ_{F1} | σ_{F2} | F_K | F_{PA} |
|----------------|-----------------|---------------|---------------|-------|----------|
| Danoliterate | 83×8 | 93% | 7% | 1 | 1 |
| ScandEval Da. | 199×8 | 88% | 12% | 1 | 2 |
| ScandEval Full | 199×24 | 77% | 16% | 2 | 2 |
| HELM Lite | 90×10 | 78% | 19% | 2 | 2 |
| OpenLLM | 2859×6 | 97% | 3% | 1 | 2 |

Table 1: How much variance did the first and second factors in EFA explain for the Danoliterate Benchmark as well as the ScandEval benchmark, both full and Danish subset (Nielsen, 2023), English benchmarks HELM Lite (Liang et al., 2022) and OpenLLM (Beeching et al., 2023) as of January 2025. Leaderboard dimensionality, model count \times scenario count, is presented along with suggested significant factor count by the Kaiser criterion and Parallel Analysis: All benchmarks have an important first component.

5 Conclusions

Based on the ability to robustly discover model groupings at different Danish capability levels and correlate these rankings with human feedback, we conclude that a scenario compilation approach can meaningfully reveal GLLM capabilities. We show that, in Danish, open-weights GLLMs currently lag behind large, closed, multilingual, instruct-tuned models, such as GPT-4 and Claude Opus.

For our evaluation setup, we observe one underlying factor in model capability across the diverse test scenarios. This observation is supported by similar structures in other Danish, English and multilingual scenario compilations which we consider a positive result for low-resource evaluation: By using curated and language-specific scenarios, the general landscape of GLLM capabilities for a given low-resource language can be meaningfully inferred even if resources limit the scale.

6 Concerns of the Ethical Impacts

This work releases a benchmark and leaderboard with the hope of a positive outcome of increased understanding of potentials and limitations of GLLMs in Danish. However, we note some risks in the use of such leaderboards.

The results presented here only focus on model capability but, on the leaderboard site, versions of other important dimensions for model applicability are presented; such as model efficiency, model likelihood calibration and model generated output toxicity. However, these are presented with a disclaimer as preliminary results and our work on

other crucial dimensions such as GLLM performance fairness across gender and nationalities or robustness to input noise have not been released due to limitations to current datasets to robustly carry out these analyses.

There is an increased risk of bias, fairness and toxicity violations in low-resource languages to which models are less tuned. Problematically, when the evaluation situation is also low-resource, these risks might be undiscovered for practitioners that only focus on a model capability. Further work is crucially needed but for now, the leaderboard site displays a disclaimer against blindly trusting that high benchmark numbers mean predictable downstream performance or applying GLLMs with unchecked assumptions about robustness, fairness, bias, and toxicity.

7 Limitations

The study only focuses on one language, Danish, with limited comparisons to other language results.

The presented benchmark consists of eight specific scenarios: Although we find high correlation between scenario results, all our statements about model performance on Danish in general are evidently biased by the scenario selection. A similar statement can be made about prompt and metric design decisions though these seem robust in ablation studies in Appendix A.

We stress the importance of the uncertainty quantification for this benchmark where all scenarios are small-scale, $n < 1,000$: The bootstrap analysis revealed some model result differences, such as Mixtral 8x7B ($d_M = 54$) and Qwen1.5 7B Chat ($d_M = 50$), are not significant at the desired level and might be spurious. Other differences such as that between GPT-4 and Claude Opus might be obscured by the important Citizenship Test scenario, where these models achieve close to 100% accuracy (Figure 3), being saturated by SOTA GLLMs. Though most other scenarios still show far-from-perfect accuracy, more difficult scenarios are needed to accommodate future developments.

As all evaluation data is publicly available, unintentional or malignant dataset contamination is possible. This issue requires attention but might, in the short-term, be less of a risk for low-resource language evaluation with smaller and less widely published corpora.

Acknowledgments

We would like to thank all the anonymous reviewers for the insightful and helpful comments. This work was supported by the Pioneer Centre for AI, DNRF grant number P1.

References

- Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, Kalika Bali, and Sunayana Sitaram. 2023. MEGA: Multilingual evaluation of generative AI. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4232–4267, Singapore. Association for Computational Linguistics.
- Elisabeth Arnbak and Carsten Elbro. 2000. Læsetekster for gymnasium, hf mv. Uddannelsesstyrelsens Internetpublikationer. Adgang 2000.
- Maurice S. Bartlett. 1951. The effect of standardization on a χ^2 approximation in factor analysis. *Biometrika*, 38(3/4):337–344.
- Edward Beeching, Clémentine Fourier, Nathan Habib, Sheon Han, Nathan Lambert, Nazneen Rajani, Omar Sanseviero, Lewis Tunstall, and Thomas Wolf. 2023. Open llm leaderboard. https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard.
- Yoav Benjamini and Yosef Hochberg. 1995. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300.
- Lukas Biewald. 2020. Experiment tracking with weights and biases. Software available from wandb.com.
- Amalie Brogaard Pauli, Maria Barrett, Ophélie Lacroix, and Rasmus Hvingelby. 2021. DaNLP: An open-source toolkit for danish natural language processing. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa 2021)*.
- Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. 2015. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–970.
- 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.
- Kenneth Enevoldsen et al. 2022. dfm-encoder-large-v1. A Transformer encoder model, part of the BERT family, intended for Danish natural language tasks.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *Proceedings of the 9th International Conference on Learning Representations (ICLR)*.
- Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python. Software available from spacy.io.
- John L. Horn. 1965. A rationale and test for the number of factors in factor analysis. *Psychometrika*, 30(2):179–185.
- Rasmus Hvingelby, Amalie Brogaard Pauli, Maria Barrett, Christina Rosted, Lasse Malm Lidegaard, and Anders Søgaard. 2020. DaNE: A named entity resource for Danish. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4597–4604, Marseille, France. European Language Resources Association.
- David Ilić. 2023. Unveiling the general intelligence factor in language models: A psychometric approach.
- Katrine Lyskov Jensen, Anna Steenberg Gellert, and Carsten Elbro. 2015. Rapport om udvikling og afprøvning af selvtest af læsning – en selvtest af voksnæs læsefærdigheder på nettet.
- Henry F. Kaiser. 1960. The application of electronic computers to factor analysis. *Educational and Psychological Measurement*, 20(1):141–151.
- Henry F. Kaiser. 1970. A second generation little jiffy. *Psychometrika*, 35:401–415.
- Oliver Kinch. 2023. Nordjylland news summarization. Hugging Face Datasets.
- Viet Lai, Chien Nguyen, Nghia Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 318–327, Singapore. Association for Computational Linguistics.
- Vladimir I. Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707–710.
- Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Šaško, Gunjan Chhablani, Bhavitvya Malik, Simon Brandeis, Teven Le Scao, Victor Sanh, Canwen Xu, Nicolas

- Patry, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Clément Delangue, Théo Matussière, Lysandre Debut, Stas Bekman, Pierric Cistac, Thibault Goehringer, Victor Mustar, François Lagunas, Alexander Rush, and Thomas Wolf. 2021. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D. Manning, Christopher Ré, Diana Acosta-Navas, Drew A. Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue Wang, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. 2022. Holistic evaluation of language models.
- Percy Liang, Yifan Mai, Josselin Somerville, Farzaan Kaiyom, Tony Lee, and Rishi Bommasani. 2023. HELM Lite: Lightweight and broad capabilities evaluation.
- 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.
- Peng Liu, Lemei Zhang, Terje Nissen Farup, Even W. Lauvrak, Jon Espen Ingvaldsen, Simen Eide, Jon Atle Gulla, and Zhirong Yang. 2023. Nlebench+norglm: A comprehensive empirical analysis and benchmark dataset for generative language models in norwegian.
- Samuel R. Mathias. 2024. Horns: Horn’s parallel analysis in python. <https://github.com/sammosummo/Horns>.
- Hiroki Nakayama. 2018. sequeval: A python framework for sequence labeling evaluation. Software available from <https://github.com/chakki-works/sequeval>.
- Dan Nielsen. 2023. ScandEval: A benchmark for Scandinavian natural language processing. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 185–201, Tórshavn, Faroe Islands. University of Tartu Library.
- Jesper Olsen. 2023. Hvor taler du flot dansk! selv skaberne er forbløffede over chatbottens sprogøre.
- OpenAI. 2023. evals. <https://github.com/openai/evals>. Accessed: 2023-11-22.
- siri.dk. 2023. Danskundervisning og prøver for udlændinge.
- Charles Spearman. 1904. "General Intelligence," Objectively Determined and Measured. *The American Journal of Psychology*, 15(2):201–292.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.
- Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023. Gpt-ner: Named entity recognition via large language models.
- 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. 2019. Transformers: State-of-the-art natural language processing. <https://github.com/huggingface/transformers>. Hugging Face, Brooklyn, USA.
- Omry Yadan. 2019. Hydra - a framework for elegantly configuring complex applications. Github.
- Marc Zao-Sanders. 2024. How people are really using genai. *Harvard Business Review*. Technology and analytics.
- Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. 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*.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhonghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena.
- Banghua Zhu, Evan Frick, Tianhao Wu, Hanlin Zhu, Karthik Ganesan, Wei-Lin Chiang, Jian Zhang, and Jiantao Jiao. 2023. Starling-7b: Improving llm helpfulness and harmlessness with rlai.

A Evaluation Methodology

A.1 Prompting

An example of a prompt following the structure shown in Figure 4 is the Citizenship Test example shown in Figure 11.

```

1 Svar kun med tallet for den rigtige
  mulighed.
2 # SPØRGSMÅL
3 Hvilket af følgende lande har flest
  indvandrere og efterkommere i
  Danmark oprindelse i?
4 # SVARMULIGHEDER
5 1. Pakistan
6 2. Iran
7 3. Tyrkiet
8 # SVAR
9 Svaret er mulighed nummer

```

Figure 11: Prompting of an example from the Citizenship Test scenario. Translated, this question prompt reads: *Answer only with the number corresponding to the correct answer. # QUESTION From which of the following countries do the highest number of immigrants and descendants in Denmark have their roots? # OPTIONS 1. Pakistan 2. Iran 3. Turkey # ANSWER The answer is option number.*

Alternatives Some prompt alternatives to the approach shown in Figures 4 and 11 were run for a subset of GLLMs: For the Citizenship Test scenario, Table 2, results were similar when not presenting options to the models, instead parsing their output choice by selecting the option with highest similarity to their generation. The same table suggests that changing the Citizenship Test scenario to a simpler prompt without markdown headers lowered results minimally..

For the summarization task, Nordjylland News, an alternative prompt with longer and more detailed instructions had no effect on model results shown in Table 3.

Translating all instruction text in the prompt format while keeping data content in Danish maintained or improved non-Danish model results, Table 4 on the Gym 2000 scenario.

A.2 GLLM Output Parsing

Multiple-choice A numbered option was considered selected if it was the only number generated by the model. In the case of multiple generated option numbers, the most frequent number was

| | Std. | Simple Q | No opt. |
|---------------------|------------|------------|------------|
| Gemini Pro | 85 ± 1 | 78 ± 1 | 79 ± 1 |
| GPT 3.5 Turbo | 82 ± 1 | 77 ± 1 | 82 ± 1 |
| Mistral 7B Instruct | 47 ± 2 | 50 ± 2 | 49 ± 2 |
| Mistral 7B | 45 ± 2 | 44 ± 2 | 41 ± 2 |
| Dano. Mistral 7B | 43 ± 2 | 45 ± 2 | 59 ± 2 |
| LlaMa 2 7B | 39 ± 2 | 42 ± 2 | 36 ± 2 |
| Dano. LlaMa 2 7B | 37 ± 2 | 40 ± 2 | |
| Dummy Baseline | 36 ± 2 | 36 ± 2 | 36 ± 2 |

Table 2: Alternative prompting and scoring approaches to the Citizenship Test run for a subset of models including a baseline outputting a fixed, random string and Danish-tuned versions of base GLLMs. Std. is the prompt version presented in the benchmark, Figure 11, Simple Q removes the first instruction and the markdown headers, simply presenting the question, the options and the final text. No opt. asks the question openly without multiple-choice options, choosing argmax similarity score, as in Section 3.2, as model choice. Presented with 95% Wald confidence interval.

| | Std. | Detailed |
|----------------------------|------------|------------|
| Gemini Pro | 74 ± 2 | 74 ± 2 |
| GPT 3.5 Turbo | 73 ± 2 | 74 ± 2 |
| Mistral 7B Instruct (v0.2) | 70 ± 2 | 71 ± 2 |
| Mistral 7B | 62 ± 3 | 58 ± 3 |
| Dano. Mistral 7B | 57 ± 3 | 59 ± 3 |
| Dano. LlaMa 2 7B | 52 ± 3 | 58 ± 3 |
| LlaMa 2 7B | 54 ± 3 | 56 ± 3 |
| Dummy Baseline | 43 ± 3 | 43 ± 3 |

Table 3: Impact of Nordjylland News alternative prompting.

chosen. Maximum generation likelihood-based selection was also implemented and is available for open-weights models on the frontend leaderboard but is not presented here.

GPT-NER Following (Wang et al., 2023), for each example, the GLLM was prompted four times, once for each entity category in the DaNE dataset (Hvingelby et al., 2020). The model was instructed to mark all words belonging to this entity category with @. These were parsed to one, multi-class prediction, handling overlap by selecting the generation with highest likelihood for models exposing probabilities. To mitigate small errors resulting in catastrophic results, model output annotated words were aligned using Levenshtein matching to the input example word list (Levenshtein, 1966).

| | Std. | English |
|----------------------------|------------|------------|
| Gemini Pro | 61 ± 8 | 64 ± 8 |
| GPT 3.5 Turbo | 45 ± 9 | 52 ± 9 |
| Danoliterate Mistral 7B | 48 ± 9 | 36 ± 8 |
| Mistral 7B | 39 ± 8 | 45 ± 9 |
| Mistral 7B Instruct (v0.2) | 36 ± 8 | 42 ± 9 |
| LlaMa 2 7B | 33 ± 8 | 33 ± 8 |
| Danoliterate LlaMa 2 7B | 27 ± 7 | 30 ± 7 |
| Dummy Baseline | 21 ± 6 | 21 ± 6 |

Table 4: How the Gym 2000 results change if the prompt instructions are in English. The instructed models handle this strongly while the Danoliterate Mistral model fails to perform under English prompting.

A.3 Metrics

Differences in results for the similarity-based metrics used for #twitterhjerne and Nordjylland News summarization are presented in Tables 5 and 6. Model rank is minimally changed.

| | Odd-one-out | Avg. sim. | Min. sim. | Max. sim. |
|----------------------|---------------|------------|------------|------------|
| GPT 3.5 Turbo | 29 ± 5 | 64 ± 5 | 61 ± 5 | 66 ± 5 |
| Gemini Pro | 31 ± 5 | 63 ± 5 | 61 ± 5 | 66 ± 5 |
| GPT 4 | 35 ± 5 | 63 ± 5 | 61 ± 5 | 66 ± 5 |
| SOLAR 10.7B Instruct | 50 ± 6 | 62 ± 5 | 60 ± 5 | 65 ± 5 |
| LlaMa 2 13B Chat | 60 ± 5 | 61 ± 5 | 58 ± 5 | 63 ± 5 |
| Mistral 7B Instruct | 64 ± 5 | 62 ± 5 | 59 ± 5 | 64 ± 5 |
| Dano. Mistral 7B | 96 ± 1 | 54 ± 6 | 52 ± 6 | 57 ± 6 |
| Dano. LlaMa 2 7B | 97 ± 1 | 53 ± 6 | 50 ± 6 | 55 ± 6 |
| OpenAI Davinci 002 | 99 ± 0.3 | 52 ± 6 | 49 ± 6 | 54 ± 6 |
| LlaMa 2 7B | 100 ± 0.3 | 51 ± 6 | 49 ± 6 | 53 ± 6 |
| Mistral 7B | 99 ± 0.3 | 50 ± 6 | 48 ± 6 | 52 ± 6 |
| Dummy Baseline | 100 ± 0.3 | 44 ± 6 | 43 ± 6 | 46 ± 6 |

Table 5: Standard version of #twitterhjerne using the odd-one-out metric (1) compared to a simpler metric just reporting average similarity score.

A.4 Software Dependencies

The relevant Python packages and their versions are presented in Table 7. Python version 3.11.8 was used.

| | BERT similarity | ROUGE-1 | ROUGE-L |
|----------------------|-----------------|------------|-------------|
| Gemini Pro | 74 ± 2 | 35 ± 3 | 28 ± 2 |
| GPT 3.5 Turbo | 73 ± 2 | 32 ± 2 | 25 ± 2 |
| GPT 4 | 73 ± 2 | 32 ± 2 | 23 ± 2 |
| SOLAR 10.7B Instruct | 71 ± 2 | 28 ± 2 | 20 ± 2 |
| Mistral 7B Instruct | 70 ± 2 | 25 ± 2 | 18 ± 2 |
| LlaMa 2 13B Chat | 69 ± 2 | 17 ± 2 | 12 ± 1 |
| Mistral 7B | 62 ± 3 | 16 ± 1 | 12 ± 1 |
| Dano. Mistral 7B | 57 ± 3 | 11 ± 1 | 8 ± 1 |
| OpenAI Davinci 002 | 55 ± 3 | 9 ± 1 | 7 ± 1 |
| Dummy Baseline | 43 ± 3 | 11 ± 1 | 8 ± 1 |
| LlaMa 2 7B | 54 ± 3 | 6 ± 1 | 5 ± 1 |
| Dano. LlaMa 2 7B | 52 ± 3 | 5 ± 1 | 4 ± 0.5 |

Table 6: Nordjylland News summarization results presented with an alternative lemma-based similarity score. The score is computed by lemmatizing text using the SpaCy framework (Honnibal et al., 2020) and then computing the ROUGE score (Lin, 2004)

| Library | Version |
|-------------------------|---------|
| google-cloud-aiplatform | 1.38.1 |
| openai | 0.28.1 |
| anthropic | 0.21.3 |
| groq | 0.4.2 |
| pandas | 1.5.3 |
| datasets | 2.14.5 |
| transformers | 4.36.1 |
| torch | 2.1.1 |
| evaluate | 0.4.0 |
| rouge_score | 0.1.2 |
| bert_score | 0.3.13 |
| huggingface_hub | 0.19.4 |
| hydra-core | 1.3.2 |

Table 7: Evaluation framework Python dependencies and used versions.

B Evaluation Corpora Details

B.1 Data Permissions

- Citizenship Test:** All rights reserved "Styrelsen for International Rekruttering og Integration". Written permission was given for the data to be re-released as an appendix to Academic work.
- HyggeSwag:** MIT.
- Gym 2000:** Unreleased. Written permission was given by CRR for Academic use but not for re-releasing the dataset.
- #twitterhjerne:** CC-BY-4.0.
- Nordjylland News:** CC-0-1.0.

6. **Cloze Self Test:** Unreleased. Written permission was given by CRR for Academic use but not for re-releasing the dataset.

7. **DaNE:** CC-BY-Sa-4.0-

8. **Angry Tweets:** CC-BY-4.0.

B.2 Data Content

All novel datasets were manually inspected for offensive content. Some crime-related and sexual themes were found in Nordjylland News examples but deemed unproblematic. The #twitterhjerne dataset was manually anonymized, removing all examples with personally identifiable content.

B.3 Examples

Below, one prompted example per evaluation corpus is presented.

1. Citizenship Test: See Figure 4.

2. HyggeSwag

```
1 Svar kun med tallet for den
  rigtige fortsættelse af
  sætningen
2 # SÆTNING
3 En gruppe venner sidder på
  slæder på toppen af bakken.
  De to venner
4 # SVARMULIGHEDER
5 1. er udstyr kørende ned ad
  bakken med en udstyrsrem på.
6 2. presser deres rygge op mod en
  klippe.
7 3. skubber en slæde med et reb,
  da hele bakken er dækket
  med sne.
8 4. skubbes ned ad bakken, og de
  glider til bunden.
9 # SVAR
10 Den rigtige fortsættelse er
  mulighed nummer
```

3. #twitterhjerne

```
1 Skriv et kort tweet på dansk,
  der besvarer nedenstående
  spørgsmål. Svar kun med
  tweetet.
2 # TWEET MED SPØRGSMÅL
3 Sønnen vil gerne lave
  #pebernødder. De par gange
  jeg har prøvet det, blev de
  kun OK. Er der nogen, der
  kan anbefale en opskrift?
  #twitterhjerne
4 # TWEET MED SVAR
5 Et svar kunne være:
```

4. Gym 2000

```
1 "Selv før jeg lærte Max Kelada
  at kende, var jeg
  indstillet på ikke at kunne
  lide ham. Krigen var lige
  blevet afsluttet, og
  passagertrafikken på de
  store oceandampere var
  livlig. Det var meget
  vanskeligt at få plads, og
  man måtte finde sig i at
  tage, hvad skibsagenterne
  tilbød én. Man kunne ikke
  vente at få en kahyt for
  sig selv, og jeg var
  temmelig taknemmelig over
  at få en, hvor der kun var
  to køjer. Men da jeg
  erfarede navnet på min
  medpassager, sank mit
  humør. Det betød lukkede
  køjer, så det ikke ville
  være muligt at få den
  mindste smule frisk luft om
  natten. Det var ubehageligt
  nok at dele kahyt i fjorten
  dage med hvem som helst
  (jeg rejste fra San
  Francisco til Yokohama),
  men jeg ville have været
  mindre bekymret ved tanken,
  hvis min medpassagers navn
  havde været Smith eller
  Brown."
```

```
2
3 Svar kun med tallet for den
  rigtige mulighed
4 # SPØRGSMÅL
5 Hvorfor var det svært at få en
  kahyt for sig selv?
6 # SVARMULIGHEDER
7 1. Det var moderne at tage på
  krydstogt.
8 2. Det var midt i ferieperioden
9 3. Mange mennesker flyttede til
  USA
10 4. Krigen var lige forbi.
11 # SVAR
12 Svaret er mulighed nummer
```

5. Cloze Self Test

```
1 "Henrik bladede frem til siderne
  med boligannoncer. Deres
  lejlighed var <MASK> for
  lille til dem, så nu ledte
  de efter noget større. De
  ville gerne flytte lidt
  tættere på kysten. De ledte
  efter en lille gård, hvor
  der var plads til at holde
  et par heste ."
2 Erstat det maskerede ord i
  ovenstående tekst (markeret
  med '<MASK>') med et af
  følgende ord: indrettet,
  solgt, annonceret, blevet.
  Svar *kun* med det rigtige
  ord:
```

6. Nordjylland News

```
1 Skriv et kort dansk resumé på én
  enkelt sætning af følgende
  tekst.
```

```

2 # TEKST
3 Manden kom kørende på Sønder
  Havnevej ved kiosken på
  havnen i Aalbæk, da han
  påkørte flere afmærkninger
  på stedet og fortsatte
  direkte ind i den bygning,
  hvor kiosken holder til.
  Der skete i forbindelse med
  påkørslen skade på
  bygningen. Uden for sad en
  mand, og han blev i lav
  fart påkørt af bilen ført
  af 53-årig mand. Den
  uheldige kiosk-gæst blev
  kørt til sygehuset med
  lettere skader.
  Nordjyllands Politi
  oplyser, at den 53-årige
  blev anholdt og sigtet for
  at køre i spirituspåvirket
  tilstand. Han er efter endt
  afhøring løsladt igen.
4 # RESUMÉ
5 Et resumé på en sætning er:

```

7. Angry Tweets

```

1 Vurdér, om sentimentet i
  følgende tweet er
  'positiv', 'neutral' eller
  'negativ'. Svar kun med et
  enkelt ord.
2 # TWEET
3 @USER Klæk det æg!
4 # SENTIMENT:
5 Sentimentet var

```

8. DaNE (prompting for location)

```

1 Fuldfør annotering af sidste
  eksempel i opgaven.
2 Her er en lingvists arbejde med
  at annotere entiteter af
  typen 'lokation'.
3 # TEKST
4 Det blev naboens store, sorte
  hund også, "siger
  Københavns politidirektør,
  Poul Eefsen,
  galgenhumoristisk til B.T.
  efter et stort smykkekup i
  hans Holte-villa og en
  række tilsvarende kup i
  området.
5 # ANNOTERING
6 Det blev naboens store , sorte
  hund også , " siger
  @@København##
  politidirektør , Poul
  Eefsen , galgenhumoristisk
  til B.T. efter et stort
  smykkekup i hans
  Holte-villa og en række
  tilsvarende kup i området .
7 # TEKST
8 Diskussionen om forklaringen på
  det "japanske økonomiske
  mirakel" har især drejet
  sig om, hvorvidt man kunne
  nøjes med økonomiske
  faktorer i sin forklaring,
  eller om det også er
  nødvendigt at inddrage

```

```

særlige kulturelle og
historiske forhold for at
finde en rimelig forklaring.
9 # ANNOTERING
10 Diskussionen om forklaringen på
  det " japanske økonomiske
  mirakel " har især drejet
  sig om , hvorvidt man kunne
  nøjes med økonomiske
  faktorer i sin forklaring ,
  eller om det også er
  nødvendigt at inddrage
  særlige kulturelle og
  historiske forhold for at
  finde en rimelig forklaring
  .
11 # TEKST
12 De lyssky fremmede elementer af
  enhver art, der har sneget
  sig til landet, er fjenden.
13 # ANNOTERING
14 De lyssky fremmede elementer af
  enhver art , der har sneget
  sig til landet , er fjenden
  .
15
16 # TEKST
17 "Vi tar'en tysker frem, vi
  tar'en tysker tilbage, vi
  tar'en tysker frem, åååårrr
  så ryster vi ham lidt!"
18 # ANNOTERING

```

C Survey

C.1 Survey Design and Instructions

For 100 GLLM use-cases divided into six categories (Zao-Sanders, 2024), we translated use-cases and categories into Danish and crafted an example prompt in Danish corresponding to that theme. We saved model answers from 18 models and used them in the survey to allow interactivity without requiring infrastructure for true dynamic model responses.

The survey front-end allowed volunteers to pick between the 100 prompts separated into categories, seeing model outputs from "Model A" and "Model B" side-by-side, streamed with a delay of 0.1 seconds between each word to simulate model generation. The volunteer was then instructed to try out at least a total of three prompts before answering. The answer consists of a question of preference, with optional additional Likert scales for each model and a text field for more details. The user instruction was in Danish meant *Two models have now been secretly selected for you: Model A and Model B. Test them out by choosing a prompt under a category that interests you. Look at the models' responses and get an impression of both A and B. Now choose a new prompt and please provide your*

Hemmeligt par af sprogmodeller #1 ud af 4

Nu er der i hemmelighed valgt to modeller for dig: A og B.

Afprøv dem ved at vælge en prompt under en kategori, der interesserer dig. Se modellernes svar og få et indtryk af både A og B. Vælg nu en ny prompt og giv endelig din vurdering efter mindst 3 prompts.

1. Vælg prompts

Udforsk de seks kategorier og vælg en prompt, der interesserer dig.

Læring og undervisning Personlig støtte Teknisk problemløsning og programmering

Prompt 1: Følg medicinske symptomer

Prompt 2: Rådgivning om iværksætteri

Prompt 3: Forretningsmæssig rådgivning

Prompt 4: Hjælp med at fokusere

Prompt 5: Terapi

3. Giv din vurdering

Du har set 3 forskellige eksempler for disse modeller.

Hvilken model foretrak du?

☒ A ☐ B

☐ Ved ikke

Hvad synes du om A?

☐ 😊 ☐ 😐 ☐ 😞 ☐ Ved ikke

Hvad synes du om B?

☐ 😊 ☐ 😐 ☐ 😞 ☐ Ved ikke

Evt. uddybende tekst:

Tilbage Næste

Afslut modellernes (læs dine svar)

2. Se modellernes svar

Jeg føler mig ensom på efter at være startet i 1.g. Det føles, som om alle bliver bedste venner omkring mig, men jeg har det svært med så mange nye mennesker. Svar gerne kort: Er det normalt, hvad jeg føler?

Model A

Ja, det er normalt! Det kan være overvældende at starte i en ny skole eller klasse, og det kan tage tid at bygge relationer

Model B

Ja, det er normalt at føle ensomhed i begyndelsen af et nyt studieprogram, hvor man møder mange nye mennesker. Det kan

Figure 12: A screenshot of the survey UI presented to users. Under header 1, users select prompts of their interest. Header 2 contains model answers side-by-side and in header 3, volunteers fill in their preference and can get model identities revealed and move on to another pair.

assessment after at least 3 prompts.⁹. See Figure 12 for an overview of the A/B test user interface.

C.2 Volunteers

The survey is openly available online, inviting users to voluntarily try out the A/B tests, filling out their preferences. The survey was promoted on social media networks and newsletters. Most of the promotions were made on channels for AI enthusiasts or professionals. Volunteers were made aware that the data would contribute to studies into GLLM evaluation in Danish.

Volunteers could optionally fill in demographic details before carrying out A/B test which is shown in Figure 13 suggesting a bias towards young, male AI professionals.

⁹Da.: Nu er der i hemmelighed valgt to modeller for dig: Model A og Model B. Afprøv dem ved at vælge en prompt under en kategori, der interesserer dig. Se modellernes svar og få et indtryk af både A og B. Vælg nu en ny prompt og giv endelig din vurdering efter mindst 3 prompts.

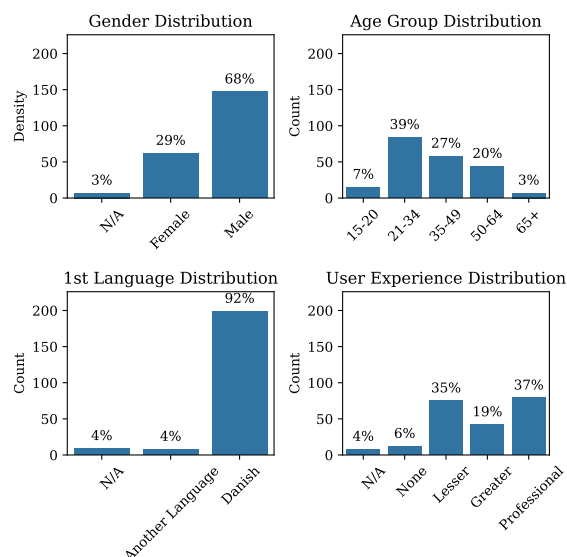


Figure 13: Demographic details filled out by survey volunteers.

Work is ongoing to increase scale and diversity of respondents.

C.3 Ranking Model

All 18 models included in the survey were sampled uniformly and the user model preference was used for ranking. An initial version ranking is the model win frequency presented in Table 8. As a model for the human preferences, we follow Chiang et al. to employ the Bradley-Terry model in a non-parametric fashion, using the sandwich robust standard errors (Chiang et al., 2024, Sec. 4, Sec. 5, Appendix B). The approach produces a linear model coefficient per model with estimated standard errors. These can be used for a paired Wilk's test to present significance of differences at $\alpha = 0.05$ level.

D Analysis Methodology Details

D.1 Model Outcomes Linear Model

The model

$$d_m = \beta_0 + \beta_1 p_m + \beta_2 \mathbb{I}(\mathcal{M} \in \text{instruct}) + \varepsilon, \quad (2)$$

where $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ and p_m is the number of model parameters in billions, was fitted with results shown in Table 9.

D.2 Factor Analysis

For the EFA on the 83×8 scenario results, the Bartlett Sphericity (Bartlett, 1951) p value is

| | Lower | Estimate | Upper |
|------------------------|-------|----------|-------|
| Claude Opus | 87% | 94% | 100% |
| Claude Sonnet | 65% | 76% | 89% |
| GPT 4o | 65% | 76% | 88% |
| Gemini Pro | 59% | 71% | 85% |
| Claude Haiku | 59% | 71% | 83% |
| GPT 4 Turbo 2024-04-09 | 55% | 67% | 80% |
| Starling 7B | 53% | 65% | 81% |
| GPT 3.5 Turbo | 48% | 61% | 72% |
| Heidrun 7B Chat | 35% | 48% | 61% |
| Gemma 7B In. | 29% | 44% | 60% |
| Mixtral 8x7B | 24% | 40% | 56% |
| LlaMa 3 70B | 24% | 40% | 53% |
| LlaMa 3 8B In. | 19% | 32% | 44% |
| SOLAR 10.7B In. | 17% | 29% | 41% |
| Munin NeuralBeagle | 13% | 26% | 39% |
| Qwen1.5 7B Chat | 11% | 22% | 33% |
| Mistral 7B In. v0.2 | 9% | 20% | 30% |
| DanskGPT-tiny Chat | 6% | 18% | 29% |

Table 8: How frequently each model wins their A/B tests with uncertainty estimation to a 95% confidence interval from bootstrapping blocked per volunteer.

| Parameter | Value | Std. error | t-value |
|-----------------|-------|------------|---------|
| $\hat{\beta}_0$ | 17.2 | 8 | |
| $\hat{\beta}_1$ | 15 | 3 | 5 |
| $\hat{\beta}_2$ | 0.4 | 0.01 | 4 |
| $\hat{\sigma}$ | 9 | | |

Table 9: Linear model fitted with $\hat{R}^2 = 0.6$ to the Danoliteracy Index for 38 open-weights models with known parameter counts.

$< 2 \cdot 10^{-16}$ and the Kaiser-Meyer-Olkin Test (Kaiser, 1970) yields a variance proportion of 90%, both suggesting that the data is usable for EFA. Fitting an EFA using the Scikit-learn Factor Analysis model yields $\lambda_1 = 5.9$, $\lambda_2 = 0.3$. Explained factor variance is calculated as eigenvalue proportion of summed eigenvalues, and the analysis is repeated for scenario results acquired from the open API at scandeval.com/danish-nlg/, at crfm.stanford.edu/helm/lite/v1.3.0/#/leaderboard, and using the OpenLLM Leaderboard Scraper GitHub project¹⁰. The datasets updated to most recent versions on January 13th, 2025.

All datasets were subjected Horn’s Parallel Analysis (Horn, 1965) simulating 1000 datasets of same shape but without correlation structure: This was implemented using the Python package `horns` (Mathias, 2024).

¹⁰github.com/Weyaxi/scrape-open-llm-leaderboard

NorEventGen: generative event extraction from Norwegian news

Huiling You¹, Samia Touileb², Erik Velldal¹, and Lilja Øvrelid¹

¹University of Oslo

²University of Bergen

{huiliny, erikve, liljao}@ifi.uio.no
samia.touileb@uib.no

Abstract

In this work, we approach event extraction from Norwegian news text using a generation-based approach, which formulates the task as text-to-structure generation. We present experiments assessing the effect of different modeling configurations and provide an analysis of the model predictions and typical system errors. Finally, we apply our system to a large corpus of raw news texts and analyze the resulting distribution of event structures in a fairly representative snap-shot of the Norwegian news landscape.

1 Introduction

Event extraction is a central information extraction task that is aimed at extracting structured representations of real-world event information provided in unstructured texts, commonly expressed in terms of an event trigger and its arguments in the text. While modeling approaches to this task have traditionally been based on sequence-labeling at the token level (Ji and Grishman, 2008; Du and Cardie, 2020; Lin et al., 2020), more recent approaches have allowed for a structure decoding that is less constrained by the exact input string. In particular, the widespread adoption of pre-trained language models based on encoder-decoder architectures have allowed for the formulation of this task as text-to-structure generation (Lu et al., 2021; Wang et al., 2023).

Current event extraction systems typically focus on English, with noteworthy exceptions for other large languages like Chinese and Arabic. This focus is largely due to the availability of manually annotated datasets in these languages (Doddington et al., 2004; Song et al., 2015). The newly released Norwegian event detection dataset EDEN (Touileb et al., 2024) contains manual annotation of news

texts from newspapers as well as transcribed news broadcasts and enable large-scale event extraction from Norwegian news sources.

In this paper, we present the NorEventGen system for Norwegian event extraction, which builds on recent developments in the formulation of event extraction as text-to-event structure generation, mapping sentences into linearized event structures. While developing this system using the recently released EDEN dataset, we also evaluate a number of modeling choices related in particular to the format of the input data and the task formulation. Specifically, we analyze the choice of pre-trained Norwegian language model, the localization of event labels using translation and the reliance on explicit trigger word identification for event argument extraction. We provide a detailed analysis of the generated event structures and examine typical errors of our system. Finally, we apply our system to a large collection of news texts from a range of different sources and provide a preliminary analysis of the extracted event structures.

The paper is structured as follows. The next section presents related work, before section 3 presents a system description for our approach. We further describe experimental set-ups in section 4, and discuss the results in section 5. Section 6 presents a use case for our system on a large Norwegian news corpus, before we summarize our finding and contributions section 7.

2 Related work

2.1 Event detection

Event extraction has commonly been approached as a supervised classification task approached through sequence labeling. Classification-based methods typically perform event extraction via several more specific subtasks (trigger detection and classification, argument detection and classification), and either solve these separately with a pipeline-based ap-

proach (Ji and Grishman, 2008; Li et al., 2013; Liu et al., 2020; Du and Cardie, 2020; Li et al., 2020) or infer these subtasks jointly at the token-level (Yang and Mitchell, 2016; Nguyen et al., 2016; Liu et al., 2018; Wadden et al., 2019; Lin et al., 2020). Moving beyond sequence-labeling, event extraction has also been approached as structured prediction into graph structures (You et al., 2022).

More recently, however, approaches that solve the event extraction task as a generation task have received more attention, mapping a text into a linearized event structure or even a natural language representation of the event. For a recent survey of generative approaches to event extraction, see (Simon et al., 2024). Of particular relevance to our work, however, is the Text2Event system of Lu et al. (2021) which pioneered the text-to-structure approach to event extraction, jointly modeling event detection and argument extraction using a T5 encoder-decoder model (Raffel et al., 2020): Given an input sentence, the model generates a structured representation of an event in the form of an S-expression (i.e., an associative dictionary of labels and values), constrained decoding is enforced to restrict the output vocabulary to valid tokens at each step. The latter is shown to be particularly helpful for small training sets. Their ablation study also includes curriculum learning and shows that using natural language tokens for argument roles is preferable to arbitrary tokens.

In an effort to further generalize the text-to-structure approach, Lu et al. (2022) introduce UIE – unified information extraction. UIE formalises a unified “structural extraction language” for encoding different information elements for different IR tasks, and includes IE-specific pre-training that removes the need for constrained decoding. Inspired by instruction tuning, Wang et al. (2023) further build on this to propose InstructUIE, where different IE tasks are reformulated into the task of natural language generation with instructions that include a description of the output format.

2.2 Event datasets

There are several manually annotated datasets for event extraction for English and a few other resource-high languages, such as Arabic and Chinese. The Automatic Content Extraction (ACE) program (Doddington et al., 2004) was an early effort in this space that resulted in several richly annotated datasets including entities, relations, and

events for English, Arabic, and Chinese. The English ACE dataset has been widely used for development of event extraction systems and annotates 8 distinct event types (e.g. `Life`, `Conflict`, `Transaction`), along with 33 subtypes (e.g. `Conflict.Attack`) and 22 event-specific subtypes that adorn specific event trigger words in the text along with their event arguments (e.g. `Attacker`, `Agent`, and `Recipient`).

The ERE (Song et al., 2015) dataset, also referred to as Light ERE comprises the same event types and subtypes as ACE. Compared to ACE, ERE adopts a more simplified scheme by merging tags (Aguilar et al., 2014). ERE also comes in a version with richer annotations, dubbed Rich ERE (Song et al., 2015), which is aimed at enabling document-level event co-reference and extends on the ACE event ontology by incorporating 9 event types and 38 event arguments (You et al., 2023).

The MAssive eVENt detection dataset (MAVEN) (Wang et al., 2020), was introduced to cover more general event types, compared to ACE and ERE. It comprised 4,480 Wikipedia documents, containing 168 event types covering 118,732 event mentions. This dataset is only annotated for event types, which are derived from FrameNet (Baker et al., 1998). In MAVEN, first candidate event triggers were semi-automatically identified, followed by an automatic labeling phase, before human annotators provided the final annotations.

3 NorEventGen: text to event records

Our system is built upon Text2Event (Lu et al., 2021), with inspiration from InstructUIE (Wang et al., 2023), as described in Section 2.1 above. Our system differs from Text2Event by applying no constraints on generation and from InstructUIE by using the input sequence only without instructions. This means approaching event extraction as a text-to-structure problem. Given the input sequence $x = x_1, \dots, x_{|x|}$, NorEventGen directly generates the event records in a linearized, structured format with a pretrained Norwegian encoder-decoder model.

3.1 Structured event records

Event records are represented in a structure similar to a linearized parse tree, where multiple event records are just sub-trees. As shown in Figure 1, an event record is structured as

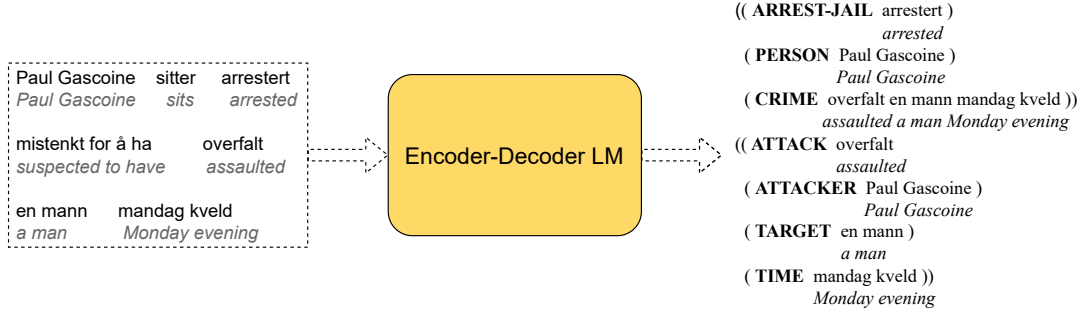


Figure 1: The architecture of NorEventGen. The model takes raw text as input and generates event records in a structured format. In this example, there are two events: (i) an ARREST-JAIL with its trigger “arrestert” and arguments Person and Crime and (ii) an ATTACK event with its trigger “overfalt” and associated Attacker, Target and Time arguments.

((Event_type trigger (arg_role₁ arg₁) (arg_role₂ arg₂)), and there are two events (ARREST-JAIL, ATTACK) from the example sentence. For a sentence that does not describe any event, empty event records are simply “()”. To differentiate from text snippets and labels for event records, during implementation, the structure indicators “()” are replaced with special tokens <extra_id.0> and <extra_id.1>, which are trained together with the model. Event records can easily be retrieved via reading structured event records as trees.

3.2 Text to structure framework

With the above mentioned structured representations, NorEventGen generates structured event records via a transformer-based encoder-decoder T5 model (Raffel et al., 2020). For an input sequence $x = x_1, \dots, x_{|x|}$, NorEventGen outputs structured event records $y = y_1, \dots, y_{|y|}$. First, the raw text sequence x is processed by the encoder into hidden states \mathbf{H} :

$$\mathbf{H} = \text{Encoder}(x_1, \dots, x_{|x|}) \quad (1)$$

With encoded input tokens, the decoder predicts the output structure token-by-token in an autoregressive manner. At each generation step i , the i -th token y_i of the output and the decoder hidden state \mathbf{h}_i^d are generated as following:

$$y_i, \mathbf{h}_i^d = \text{Decoder}([\mathbf{H}; \mathbf{h}_1^d, \dots, \mathbf{h}_{i-1}^d]) \quad (2)$$

Decoder(\cdot) predicts the conditional probability $p(y_i | y < i, x)$ for the token y_i . Prediction terminates once the end symbol (<eos>) is generated.

| Split | #Sents | #Tokens | #Events | #Arguments |
|-------|--------|---------|---------|------------|
| Train | 20,968 | 326,145 | 4,584 | 7,416 |
| Dev | 1,919 | 35,668 | 387 | 626 |
| Test | 3,365 | 57,413 | 834 | 1,257 |

Table 1: Statistics of the Norwegian EDEN dataset.

Compared with some previous studies which treat labels (event ontology) as specific symbols or enforce various constraints during the decoding process, our text-to-structure framework treats labels as natural language tokens and employs greedy decoding during the generation stage. By verbalizing and generating the labels, the model learns event schema knowledge during training.

4 Experiments

In the following, we present the details of our experimental setup, and the specific experiments conducted as evaluation of our model.

4.1 Experimental setup

EDEN The recently released Event DETection for Norwegian (EDEN) dataset (Touileb et al., 2024) generally adopts the ACE annotation schema and further adapts it to the annotation of news data and transcribed news broadcasts in Norwegian. The event ontology of EDEN defines 34 event types and 28 event argument roles. In total, it contains data from 630 documents containing over 500k tokens and almost 6,000 unique events. Detailed statistics can be found in Table 1.

Pre-trained LMs As mentioned above, we will be using the T5 architecture (Raffel et al., 2020) for

the underlying base model. We experiment with two different versions pre-trained for Norwegian, named North-T5¹ and NorT5 (Samuel et al., 2023). Both come in several sizes, and we here use North-T5 base (220 million parameters) and large (770M), and NorT5 base (228M) and large (808M). The main difference is that while the NorT5 models were trained from scratch for Norwegian, the North-T5 models are based on the multilingual mT5 (Xue et al., 2021) (including the tokenizer) with further fine-tuning for Norwegian.

Evaluation Event extraction is evaluated on two key elements: 1) an *event trigger* is correctly predicted if the event type and trigger word(s) match a reference trigger; 2) an *event argument* is correctly predicted if its role type, event type, and argument word(s) match a reference argument. We report F measure (F1) for the following four metrics: Trg-I (trigger identification), Trg-C (trigger classification), Arg-I (argument identification), and Arg-C (argument classification). Since our system directly generates event records, the offset of the generated tokens in the input sequence is unknown; when evaluating trigger and argument identification, we therefore require an exact match towards a substring of the input text.

System comparison We compare our NorEventGen with JSEEGraph (You et al., 2023), a semantic-graph-parsing approach with previously reported results for the EDEN dataset. JSEEGraph differs fundamentally from our NorEventGen, since it is essentially an extract-and-classify approach.

Implementation detail All the reported models were trained on a single node of Nvidia RTX3090 GPU. We adopt AdamW (Loshchilov and Hutter, 2019) to optimize model weights with the learning rate of $6e-6$. We train all the models with batch size of 16 for 25 epochs. All the hyper-parameters are tuned on the development set of EDEN.

4.2 Experiments on label translation

Most event ontologies are formulated in English, including that of EDEN, which adopts the ACE annotation schema in English for the annotation of Norwegian texts. As such, the serialized event structures contain a mixture of Norwegian and English (see Figure 1). When monolingual models

¹For access and more information about the North-family of models, please see; <https://huggingface.co/north>

| Model | Trans | Trg-I | Trg-C | Arg-I | Arg-C | PLM |
|-------------|-------|-------------|-------------|-------------|-------------|----------------|
| JSEEGraph | — | 69.1 | 68.0 | 52.4 | 51.5 | XLMR-large |
| NorEventGen | — | 61.8 | 47.4 | 48.5 | 47.4 | NorT5-base |
| | ✓ | 69.0 | 66.0 | 55.4 | 52.7 | |
| | — | 63.1 | 61.1 | 51.8 | 50.1 | NorT5-large |
| | ✓ | 69.4 | 66.8 | 56.8 | 54.9 | |
| | — | 61.3 | 57.9 | 44.4 | 42.0 | North-T5-base |
| | ✓ | 61.7 | 58.1 | 45.2 | 42.9 | |
| | — | 66.7 | 64.2 | 54.7 | 52.6 | North-T5-large |
| | ✓ | 67.6 | 65.7 | 56.0 | 54.3 | |

Table 2: Experimental results on EDEN (F1-score, %). Trg-I and Trg-C correspond to event trigger identification and classification; Arg-I and Arg-C correspond to event argument identification and classification. Trans indicates whether the labels are translated into Norwegian.

are used on non-English datasets, this language mix might affect model performance. To examine the influence of English labels on Norwegian event generation, we translate the ontology (event types and argument roles) into Norwegian, so that both labels and texts are in Norwegian. By comparing the results on original and translated datasets, we can evaluate to what extent the event structure language influences the results.

4.3 Experiments on trigger essentiality in structured event generation

As mentioned above, event extraction has traditionally been approached as a token-based classification task, which explicitly anchors the event structures to tokens in the input. This means that the classification of the event type is explicitly related to the event trigger word. For the current approach, this relation is less constrained, and it is therefore possible to evaluate the extent to which event extraction performance relies on the generation of the event triggers. Although the task of event extraction includes both event detection and argument extraction, the evaluation of arguments is exclusive of the trigger words, and is only affected by event type prediction. With our NorEventGen framework, it is convenient to re-structure the output by excluding the trigger text generation, by simply updating the structured event record to `((Event_type (arg_role1 arg1) (arg_role2 arg2)))`. Together with the change of task formulation, we introduce “Evt-C” (event type classification) as the metric to evaluate event type prediction; an event type is correctly predicted if it matches a gold event type. The evaluation metric for event arguments remains the same.

| PLM | Trans | Top 5 difficult event types |
|----------------|--------|---|
| NorT5-base | ✓
— | END-ORG, TRIAL-HEARING, START-POSITION, START-ORG, ELECT
END-ORG, START-ORG, CHARGE-INDICT, CONVICT, TRIAL-HEARING |
| NorT5-large | ✓
— | START-ORG, TRIAL-HEARING, END-ORG, BE-BORN, TRANSFER-MONEY
START-ORG, TRIAL-HEARING, CONVICT, CHARGE-INDICT, END-ORG |
| North-T5-base | ✓
— | START-ORG, BE-BORN, END-ORG, TRIAL-HEARING, CHARGE-INDICT
INJURE, END-ORG, TRIAL-HEARING, PHONE-WRITE, START-ORG |
| North-T5-large | ✓
— | END-ORG, BE-BORN, TRIAL-HEARING, CONVICT, START-ORG
END-ORG, START-ORG, TRIAL-HEARING, CONVICT, BE-BORN |

Table 3: Top 5 difficult event types for our models to predict, measured by F1 scores of Trg-C (event trigger classification). Trans indicates whether the labels are translated into Norwegian.

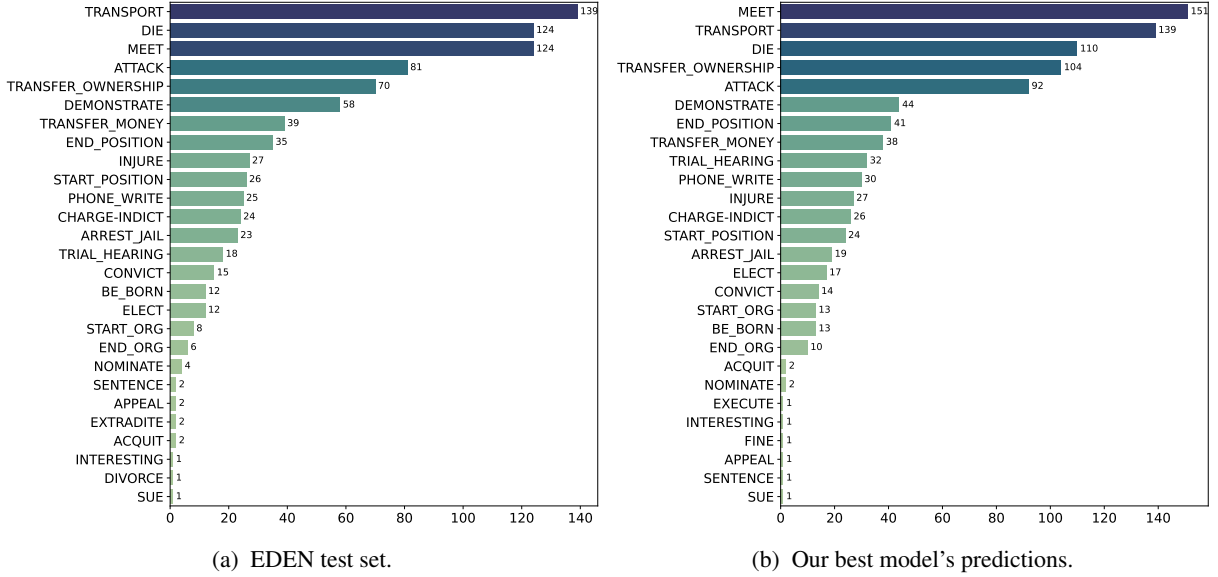


Figure 2: Event type distributions in EDEN test set versus our best model's predictions.

| PLM | Evt-C | Arg-I | Arg-C |
|----------------|-------|-------|-------|
| NorT5-base | 61.3 | 30.4 | 28.5 |
| NorT5-large | 63.6 | 21.5 | 20.6 |
| North-T5-base | 54.7 | 28.0 | 25.3 |
| North-T5-large | 59.0 | 22.2 | 21.2 |

Table 4: Experimental results with trigger text extraction excluded (F1-score, %). “Evt-C” refers to event type classification; Arg-I and Arg-C correspond to event argument identification and classification

5 Results and discussion

We here present the results of NorEventGen on Norwegian event extraction with generative modeling. We first present the overall performance for different model configurations, before discussing the role of label translation and trigger generation, as described above. We then provide a more in-depth

analysis of the generated event structures with a specific focus on invalid generations and present an error analysis for the best performing system.

5.1 Overall performance

As shown in Table 2, our results align quite closely with those of JSEEGraph. Compared with previous work, our system shows better performance on event argument extraction; our best-performing system presents an improvement of around 4 percentage points on both argument identification and classification F1 scores. However, on trigger extraction, only large models are on par with previous work.

In terms of the choice of pretrained LMs, NorT5 generates better results than North-T5 across different model sizes, which is especially true for base models. For model size, moving from a base model to a large model, we find that the results improve

| PLM | Trans | Event type | | | Trigger | | | Argument role | | | Argument | | |
|----------------|-------|------------|-------|-------|----------|-------|-------|---------------|-------|-------|----------|-------|-------|
| | | #Invalid | #Gold | #Pred | #Invalid | #Gold | #Pred | #Invalid | #Gold | #Pred | #Invalid | #Gold | #Pred |
| NorT5-base | — | 0 | 881 | 614 | 2 | 881 | 614 | 1 | 1,524 | 1010 | 7 | 1,524 | 1,010 |
| | ✓ | 1 | 881 | 803 | 8 | 881 | 883 | 1 | 1,524 | 1,374 | 16 | 1,524 | 1,374 |
| NorT5-large | — | 0 | 881 | 656 | 0 | 881 | 656 | 1 | 1,524 | 1,030 | 4 | 1,524 | 1,030 |
| | ✓ | 2 | 881 | 956 | 10 | 881 | 956 | 1 | 1,524 | 1,595 | 10 | 1,524 | 1,595 |
| North-T5-base | — | 0 | 881 | 856 | 1 | 881 | 856 | 0 | 1,524 | 1,527 | 4 | 1,524 | 1,527 |
| | ✓ | 1 | 881 | 939 | 3 | 881 | 939 | 0 | 1,524 | 1,649 | 5 | 1,524 | 1,649 |
| North-T5-large | — | 0 | 881 | 1,065 | 5 | 881 | 1,065 | 0 | 1,524 | 1,835 | 10 | 1,524 | 1,835 |
| | ✓ | 0 | 881 | 997 | 5 | 881 | 997 | 1 | 1,524 | 1,698 | 3 | 1,524 | 1,698 |

Table 5: Invalid generations. Valid tokens are the event ontology (event types and argument roles) and the input sequence. For each item, the number of invalid instances are listed; “#Gold” and “#Pred” refer to the number of reference and predicted instances. “Trans” refers translated ontology into Norwegian.

considerably.

In terms of event types, as shown in Table 3, difficult event types to predict are largely shared across all of our models, and these event types are somewhat less frequent (as shown in Figure 2a). In particular, three event types (END-ORG, START-ORG, TRIAL-HEARING) are always among the top 5 difficult event types. Under different experimental setups, certain event types can also be difficult to predict; for instance, INJURE event even ranks as the most difficult event type for North-T5-base model trained on EDEN with translated labels.

5.2 Label translation

We further find that translating the language of the event ontology is beneficial for all models, in particular for the NorT5 model. The fact that the gain for North-T5 is less could be due to the fact that the model is continually trained from a multilingual T5 model, so it has substantial knowledge of English. In contrast, as a monolingual model trained from scratch for Norwegian, NorT5 is able to benefit more from the translated labels.

5.3 The importance of trigger generation

From Table 4, it is clear that excluding trigger generation (in both training and testing) dramatically affects the performance negatively for both event type prediction and argument extraction, in particular the latter. The scores for argument identification and classification are almost halved across all models. For event type classification, the F1 scores are also considerably lower. To sum up, trigger word(s) generation lies at the core of structured event record generation, since it is the strong indicator of event types, which further affect the evaluation of event arguments.

In terms of pretrained LMs, NorT5 performs better than North-T5 in both base and large variants. Considering the individual subtasks, the large models tend to perform better than the base versions on event type generation, but worse on argument generation, in this particular set-up.

5.4 Analysis of generated event structures

The task of event extraction relies on extraction and classification, namely extracting text spans (event trigger / argument) from the input sequence and labelling (event type / argument role) them. As such, in the context of generation, only tokens from the event ontology and the input sequence are valid generations. Since we do not apply additional decoding constraints during generation, the model is forced to learn the event ontology knowledge and attend to input tokens. Table 5 presents statistics for the generated event type labels, trigger words, argument role labels and argument words for the various model configurations. In general, models trained with NorT5 tend to under-predict, while models trained with North-T5 tend to over-predict. The number of predicted arguments is strongly influenced by the number of predicted event triggers, i.e., more predicted triggers come with more predicted arguments.

When it comes to the generation of invalid event triggers or arguments, as shown in Table 5, such invalid generations are minimal. In terms of event ontology, across all settings, the model rarely generates event type or argument role labels outside the ontology knowledge contained in the training data. There are maximum 2 cases out of hundreds of instances, for both event type and argument role. When it comes to extracting text spans from the input tokens for event triggers and arguments, we find that there are more cases of invalid generations. In general, the number of invalid trigger words is

consistently lower than that of invalid arguments for the same model. We also find that the models using label translation seem to generate a higher proportion of invalid arguments than the models trained on non-translated event structures. The last rows of Table 6 provides an example of invalid trigger/argument generations.

5.5 Error analysis

There are various errors made by our model, as summarized in Table 6. Similar to classification-based models, our model predicts either wrong event type or argument role, and can extract wrong text spans for trigger or argument, e.g. in the case of the partially overlapping triggers “statlige tilskudd”. Errors are also prevalent in cases of nested event arguments, which is a common challenge for event extraction systems (You et al., 2023). In Table 6, we see that the `Entity` argument of the `End-Position` event is nested within the `Position` argument, a relation that the system does not accurately predict.

Generation-based methods also introduce some new error types, namely invalid generations, as discussed above. These errors commonly occur in generated trigger word(s) or argument word(s) where the model generates words that do not occur in the original input text. We find that our model would generate synonyms of the gold tokens, like the listed example; “frijent” and “frifinn” are synonyms, both meaning “acquit”. We also find that it is possible for the model to output just part of a token, like “sør” from “sørøver”.

6 Use case: event extraction from Norwegian news

One of the main use cases for event extraction systems is the automated analysis of large collections of news texts. An interesting question is whether the distribution of event types in newer news sources is similar to that found for the EDEN dataset (based on the somewhat dated news sources from the Norwegian Dependency Treebank (Øvrelid and Hohle, 2016)). We here apply our best model² on a newly collected news corpus dubbed the Norwegian MediaCorpus³. The MediaCorpus collects millions of news articles in 2010s

²Our best model is trained with NorT5-large on EDEN with translated event ontology.

³The corpus can be accessed online on: <https://clarino.uib.no/korpuskel/corpora>

from three major media houses in Norway: Amedia, Schibsted, and TV 2. Given its size, the corpus provides a representative sample of the Norwegian news landscape. Table 7 provides detailed statistics of the corpus. We randomly select a smaller set from the entire MediaCorpus to test our model; specifically, we select 200,000 articles from each media house. Detailed statistics are shown in Table 7.

6.1 Event types distribution in MediaCorpus

As shown in Figure 2b, on the test set of EDEN, the event types produced by our model share a similar distribution with the gold event types (Figure 2a). The distribution of predicted event types for the selected subset of MediaCorpus is shown in Figure 3, which also resembles the one on the EDEN test set, with a long tail. Even though the most frequent event type is still `MEET`, the proportion is much larger, and none other event types are on par. As shown in Table 8, among the top 10 trigger words for `MEET` event, apart from explicit words related to meetings, half of them are related to sport matches and Word Cup even ranks as the top 10. The event ontology of EDEN does not cover sports event types, though they are often news-worthy, but those events are predicted into the closet event type in the ontology, namely `MEET`. This phenomenon may indicate that frequent event types reported in the news will still be predicted, though not covered by the ontology itself.

Other frequent event types are `TRANSPORT`, `TRANSFER-OWNERSHIP`, `TRANSFER-MONEY`, `ATTACK`, and `INJURE`. Similarly, the least frequent event types in the MediaCorpus overlap with those in EDEN, such as `SUE`, `ACQUIT`, and `DIVORCE`. In summary, EDEN represents the Norwegian news landscape relatively well, and our NorEventGen model trained on the same dataset has value in real-life application.

6.2 Article tag vs event types

Each article in MediaCorpus has one or more custom tags. These are tags that have been manually assigned by journalists to the article in question. There are 287,687 unique tags in the entire MediaCorpus. Such a large set of article tags can be attributed to the authors’ creativity and the lack of a consistent tag set. The most frequent tag `nyheter` (“news”) is incredibly vague, and about 20% of the articles would be assigned this tag. Sports related tags are also among the most

| Error type | Gold | Pred |
|------------------|--|---|
| Wrong event type | Input: Skole evakuert etter trusler på Internett
<i>School evacuated after threats on the Internet</i>
Event type: TRANSPORT
Trigger: evakuert
Artifact: Skole | Event type: ATTACK
Trigger: evakuert
Target: Skole |
| Wrong trigger | Input: nye St. Olavs hospital ikke kan forvente flere statlige tilskudd.
<i>new St. Olavs hospital cannot expect more government grants.</i>
Event type: TRANSFER-MONEY
Trigger: tilskudd | Event type: TRANSFER-MONEY
Trigger: statlige tilskudd |
| Missing argument | Input: Ledelsen av EU skifter fortsatt hvert halvår.
<i>The leadership of EU changes still every six months.</i>
Event type: END-POSITION
Trigger: skifter
Position: Ledelsen av EU
Entity: EU | Event type: END-POSITION
Trigger: skifter
Position: Ledelsen av EU |
| Invalid trigger | Input: Han tilsto det ene drapet, men ble frikjent for drapet på sløgedal Paulsen.
<i>He confessed the one murder, was acquitted of the murder of Sløgedal Paulsen.</i>
Event type: ACQUIT
Trigger: frijent | Event type: ACQUIT
Trigger: frifinn |
| Invalid argument | Input: ... å selge trålfartøy med konsesjon sørover, mens det er helt kurant å selge andre veien.
<i>... to sell trawlers with license in the south, while it is normal to sell the other way</i>
Event type: TRANSFER-OWNERSHIP
Trigger: selge
Artifact: trålfartøy med konsesjon | Event type: TRANSFER-OWNERSHIP
Trigger: selge
Artifact: trålfartøy med konsesjon
Place: sør |

Table 6: Typical errors made by our best-performing model trained with NorT5-large.

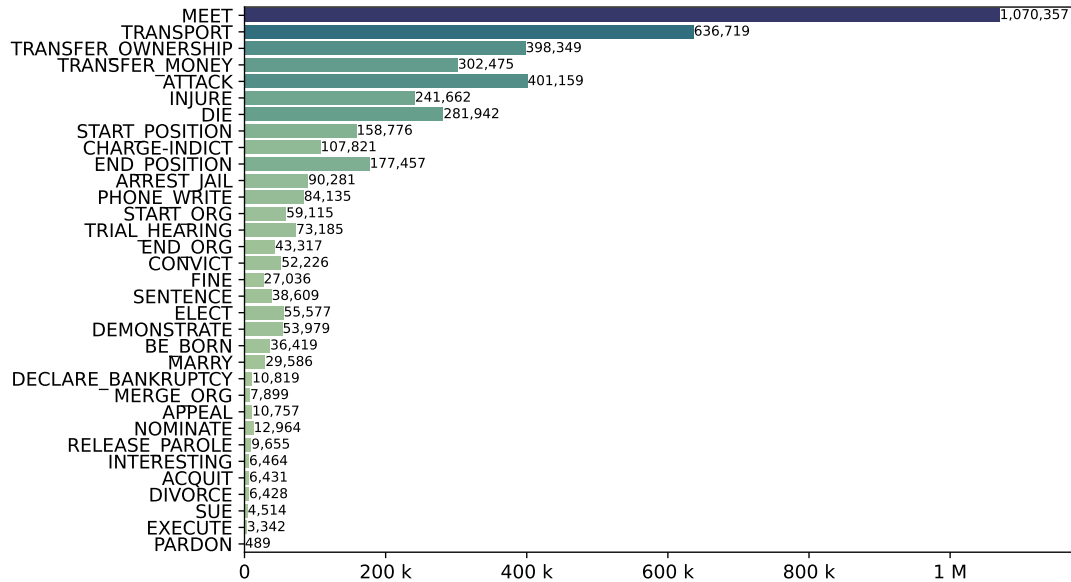


Figure 3: Predicted event types distribution on selected set of MediaCorpus.

frequent tags, and football stands out from other sports as `fotball` (“football”) is the third most frequent tag. In real life, sports is an important

news-worthy topic, but the related event types are not covered in the event ontology of EDEN.

To better evaluate the relationship between

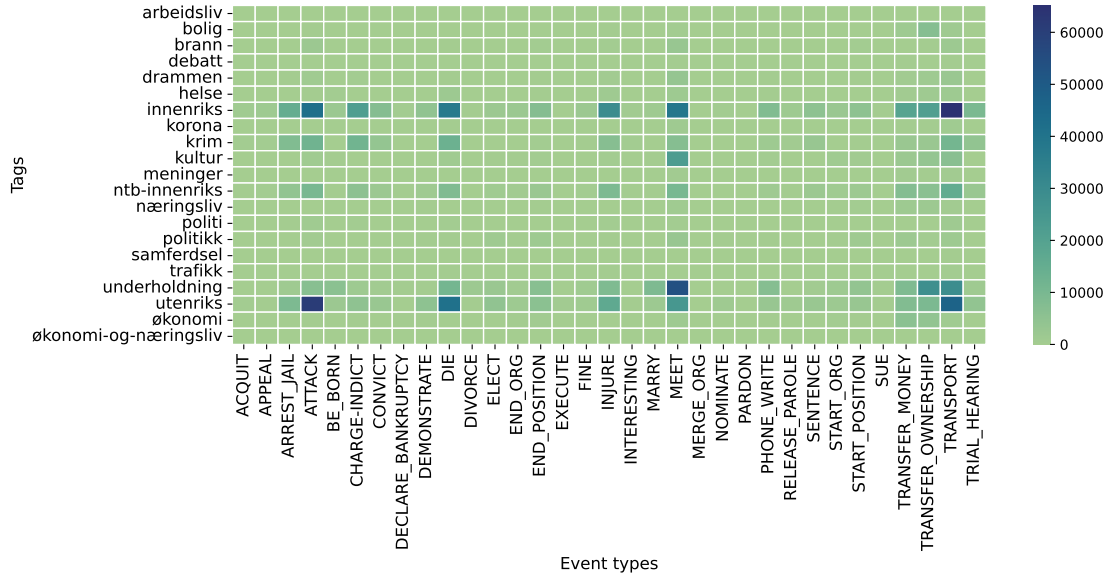


Figure 4: Frequencies of document tag and event type (predictions on selected set of MediaCorpus)

| Source | #Docs | #Sents | #Tokens |
|----------------------|-----------|-------------|---------------|
| Entire corpus | | | |
| Amedia | 5,263,591 | 139,482,285 | 2,218,694,185 |
| Schibsted | 2,710,885 | 67,802,274 | 1,089,613,721 |
| TV 2 | 585,772 | 12,741,165 | 192,327,865 |
| Selected set | | | |
| Amedia | 200,000 | 3,767,797 | 58,515,290 |
| Schibsted | 200,000 | 5,572,248 | 91,012,216 |
| TV 2 | 200,000 | 3,914,595 | 55,298,721 |

Table 7: Statistics of MediaCorpus.

article tags and event types, tags similar to *nyheter* and sports-related tags are excluded. The frequencies of article tag vs event type are shown in Figure 4. In general, a strong correlation between article tag and event type is not clear. There are several tags that frequently co-occur with events: *innenriks* (“domestic”), *krim* (“crime”), *utenriks* (“abroad”), and *underholdning* (“entertainment”). These tags often occur together with *ATTACK*, *DIE*, *MEET*, *TRANSPORT*, *TRANSFER-MONEY* and *TRANSFER-OWNERSHIP* events. It is clear that events about violence and economic activities are news-worthy both domestically and abroad.

7 Conclusion

In this paper, we address event extraction from Norwegian news with a generation-based method. Our experiments on the Norwegian EDEN dataset show that our NorEventGen model is able to ac-

| | |
|---------|-------------|
| kampen | match |
| møte | meeting |
| kamp | match |
| kamper | matches |
| møter | meet |
| møtet | the meeting |
| møtte | met |
| besøk | visit |
| kampene | the matches |
| VM | World Cup |

Table 8: Top 10 trigger words for *MEET* event in the predictions of the selected MediaCorpus.

quire event ontology knowledge and generate tokens from the input sequence for event triggers and arguments, thus it is not necessary to implement constraints during the generation process. In our experiments, we also find that it is highly beneficial to localize the event ontology to the target language, in our case Norwegian, and using a monolingual Norwegian model is more beneficial. Beyond the EDEN dataset, we extend our system to process a large corpus of raw Norwegian news texts. By applying our model to this broader dataset, we analyze the predicted event distribution, providing insights into the types of events prevalent in Norwegian news. This analysis serves as a snapshot of the Norwegian news landscape and illustrates the potential applications of our approach for large-scale event analysis in less-resourced languages.

Acknowledgments

This work was supported by industry partners and the Research Council of Norway with funding to MediaFutures: Research Centre for Responsible Media Technology and Innovation, through the centers for Research-based Innovation scheme, project number 309339.

References

- Jacqueline Aguilar, Charley Beller, Paul McNamee, Benjamin Van Durme, Stephanie Strassel, Zhiyi Song, and Joe Ellis. 2014. A comparison of the events and relations across ACE, ERE, TAC-KBP, and FrameNet annotation standards. In *Proceedings of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 45–53, Baltimore, Maryland, USA. Association for Computational Linguistics.
- Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley framenet project. In *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*.
- George R Doddington, Alexis Mitchell, Mark A Przybicki, Lance A Ramshaw, Stephanie M Strassel, and Ralph M Weischedel. 2004. The automatic content extraction (ACE) program-tasks, data, and evaluation. In *International Conference on Language Resources and Evaluation*, volume 2, pages 837–840. Lisbon.
- Xinya Du and Claire Cardie. 2020. Event extraction by answering (almost) natural questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 671–683, Online. Association for Computational Linguistics.
- Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In *Proceedings of ACL-08: HLT*, pages 254–262, Columbus, Ohio. Association for Computational Linguistics.
- Fayuan Li, Weihua Peng, Yuguang Chen, Quan Wang, Lu Pan, Yajuan Lyu, and Yong Zhu. 2020. Event extraction as multi-turn question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 829–838, Online. Association for Computational Linguistics.
- Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 73–82, Sofia, Bulgaria. Association for Computational Linguistics.
- Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7999–8009, Online. Association for Computational Linguistics.
- Jian Liu, Yubo Chen, Kang Liu, Wei Bi, and Xiaojiang Liu. 2020. Event extraction as machine reading comprehension. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1641–1651, Online. Association for Computational Linguistics.
- Xiao Liu, Zhunchen Luo, and Heyan Huang. 2018. Jointly multiple events extraction via attention-based graph information aggregation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1247–1256, Brussels, Belgium. 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.
- Yaojie Lu, Hongyu Lin, Jin Xu, Xianpei Han, Jialong Tang, Annan Li, Le Sun, Meng Liao, and Shaoyi Chen. 2021. Text2Event: Controllable sequence-to-structure generation for end-to-end event extraction. 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 2795–2806, Online. Association for Computational Linguistics.
- Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5755–5772, Dublin, Ireland. Association for Computational Linguistics.
- Thien Huu Nguyen, Kyunghyun Cho, and Ralph Grishman. 2016. Joint event extraction via recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 300–309, San Diego, California. Association for Computational Linguistics.
- Lilja Øvrelid and Petter Hohle. 2016. Universal Dependencies for Norwegian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 1579–1585, Portorož, Slovenia. European Language Resources Association (ELRA).
- 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.

- David Samuel, Andrey Kutuzov, Samia Touileb, Erik Velldal, Lilja Øvrelid, Egil Rønningstad, Elina Sigdel, and Anna Palatkina. 2023. NorBench – a benchmark for Norwegian language models. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 618–633, Tórshavn, Faroe Islands. University of Tartu Library.
- Étienne Simon, Helene Bøsei Olsen, Huiling You, Samia Touileb, Lilja Øvrelid, and Erik Velldal. 2024. Generative Approaches to Event Extraction: Survey and Outlook. In *Proceedings of FuturED 2024: Workshop on the Future of Event Detection*, Miami, USA.
- Zhiyi Song, Ann Bies, Stephanie Strassel, Tom Riese, Justin Mott, Joe Ellis, Jonathan Wright, Seth Kulick, Neville Ryant, and Xiaoyi Ma. 2015. From light to rich ERE: Annotation of entities, relations, and events. In *Proceedings of the 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 89–98, Denver, Colorado. Association for Computational Linguistics.
- Samia Touileb, Jeanett Murstad, Petter Mæhlum, Lubos Steskal, Lilja Charlotte Storset, Huiling You, and Lilja Øvrelid. 2024. Eden: A dataset for event detection in norwegian news. In *Proceedings of the Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation ((LREC-COLING 2024))*. European Language Resources Association (ELRA).
- David Wadden, Ulme Wennberg, Yi Luan, and Hanneh 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.
- Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, et al. 2023. Instructuie: Multi-task instruction tuning for unified information extraction. *arXiv preprint arXiv:2304.08085*.
- Xiaozhi Wang, Ziqi Wang, Xu Han, Wangyi Jiang, Rong Han, Zhiyuan Liu, Juanzi Li, Peng Li, Yankai Lin, and Jie Zhou. 2020. Maven: A massive general domain event detection dataset. *arXiv preprint arXiv:2004.13590*.
- 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*, pages 483–498, Online. Association for Computational Linguistics.
- Bishan Yang and Tom M. Mitchell. 2016. Joint extraction of events and entities within a document context. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 289–299, San Diego, California. Association for Computational Linguistics.
- Huiling You, David Samuel, Samia Touileb, and Lilja Øvrelid. 2022. Eventgraph: Event extraction as semantic graph parsing. In *Proceedings of CASE: The 5th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text*.
- Huiling You, Lilja Vrelid, and Samia Touileb. 2023. JSEEGraph: Joint structured event extraction as graph parsing. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 115–127, Toronto, Canada. Association for Computational Linguistics.




Lessons Learned from Training an Open Danish Large Language Model

Mike Zhang  Max Müller-Eberstein  Elisa Bassignana 

Rob van der Goot 

 Aalborg University, Denmark

 IT University of Copenhagen, Denmark

 Pioneer Center for Artificial Intelligence, Denmark

jjz@cs.aau.dk {mamy, elba, robv}@itu.dk

Abstract

We present SNAKMODEL, a Danish large language model (LLM) based on LLAMA2-7B, which we continuously pre-train on 13.6B Danish words, and further tune on 3.7M Danish instructions. As best practices for creating LLMs for smaller language communities have yet to be established, we examine the effects of early modeling and training decisions on downstream performance throughout the entire training pipeline, including (1) the creation of a strictly curated corpus of Danish text from diverse sources; (2) the language modeling and instruction tuning training process itself, including the analysis of intermediate training dynamics, and ablations across different hyperparameters; (3) an evaluation on eight language and culturally-specific tasks. Across these experiments SNAKMODEL achieves the highest overall performance, outperforming multiple contemporary LLAMA2-7B-based models. By making SNAKMODEL, the majority of our pre-training corpus, and the associated code available under open licenses, we hope to foster further research and development in Danish Natural Language Processing, and establish training guidelines for languages with similar resource constraints.¹

1 Introduction

The landscape of large language models (LLMs) has seen rapid expansion, with an increasing

trend towards open-weight releases for a broader range of languages. Notable English-centric examples include Pythia (Biderman et al., 2023), Vicuna (Zheng et al., 2023), Mistral (Jiang et al., 2023), Qwen (Bai et al., 2023), Llama2 (Touvron et al., 2023), Llama3 (Dubey et al., 2024), OLMo (Groeneveld et al., 2024), and Phi (Abdin et al., 2024). Simultaneously, recent efforts have extended LLMs to multilingual settings, including models such as mT5 (Xue et al., 2021), Bloom (Le Scao et al., 2023), Aya (Üstün et al., 2024; Singh et al., 2024), RomanSetu (J et al., 2024), and EuroLLM (Martins et al., 2024).

As anglocentric and/or multilingual LLMs have nonetheless continued struggling to adapt to lower-resource settings—especially with respect to pragmatic and sociolinguistic factors (Hershcovich et al., 2022; Cao et al., 2023; Naous et al., 2024; Wang et al., 2024)—there is growing interest in language-specific LLMs, either tailored to a single language (see Related Work; Section 2) or specialized for a small set of similar languages (SiloAI, 2024; Dou et al., 2024). However, the best practices for creating such language-adapted LLMs have yet to be established—especially for smaller language communities with resource limitations with respect to data, compute, or both.

Danish offers a particularly interesting testbed among these smaller languages. As a mid-resource language, which is typologically related to English and has largely overlapping character sets, it has sufficient textual data for LLM adaptation, yet is far from the levels of its neighbors (e.g., Swedish; Ekgren et al., 2024). Additionally, it lacks advanced resources like native instruction-tuning data or human-preference data, making it necessary to use translated datasets for which the downstream effects on model functionality are not yet well understood. Linguistically, Danish has also been shown

[⊗]These authors contributed equally.

¹The code and data scripts are available here:
<https://github.com/nlpnorth/snakmodel/>.

to be more challenging to learn for humans than its neighbors due its phonological complexity (Trecca et al., 2021; Christiansen et al., 2023), which results in downstream effects on discourse, such as additional conversational redundancy (Christiansen et al., 2023; Dideriksen et al., 2023).

With the goal to provide the Danish community with a custom-adapted resource, as well as to establish better-grounded guidelines for creating LLMs in languages with similar linguistic characteristics and resource constraints, we present and analyze SNAKMODEL-7B_{base/instruct}, two LLMs designed specifically for the Danish language. Our base model builds upon LLAMA2-7B, which we continuously pre-train on a diverse collection of Danish corpora comprising 350M documents (sentences/paragraphs) and 13.6B words, before tuning it on 3.7M Danish instruction-answer pairs. We evaluate our model against contemporary LLAMA2-7B-based models on the Danish part of the ScandEval benchmark (Nielsen, 2023) that encompasses both language and culture-specific tasks. By releasing not just the related artifacts (final model, intermediate checkpoints, pre-training data, code), but by also analyzing the effects of early decisions in the training and model design process on intermediate training dynamics and downstream performance, we aim to provide resources that are not just relevant for Danish, but for LLM adaptation in general.

Contributions. This work contributes:

- A large, diverse, high-quality collection of Danish corpora, totaling 350M documents with 13.6B words (Section 3). We provide scripts to collect and process the data.
- SNAKMODEL-7B_{base/instruct}, two open-weight 7B-parameter language models continuously pre-trained and instruction-tuned specifically for Danish, for which we release all related artefacts, and extensively analyze the model’s intermediate training dynamics (Section 4).
- An evaluation comparing SNAKMODEL-7B_{instruct} and contemporary Danish models, which analyzes performance with respect to language and cultural tasks (Section 5).
- A consolidation of our findings into recommendations for efficiently training LLMs under similar resource constraints (Section 6).

2 Related Work

Continuously Pre-trained LLMs. Previous work has shown that for both encoder and decoder language models (LM), continuous pre-training is the de facto standard for adapting an LM to a specific domain (Han and Eisenstein, 2019; Alsentzer et al., 2019; Lee et al., 2020; Gururangan et al., 2020; Nguyen et al., 2020) or another language, such as German (LeoLM-Team, 2024), Spanish and Catalan (Àguila Team, 2023), Finnish (Luukkonen et al., 2023), Dutch (Rijgersberg and Lucassen, 2023; Vanroy, 2024), Italian (Bacciu et al., 2024), Japanese (Rakuten Group et al., 2024), Basque (Etxaniz et al., 2024), Swedish (AI-Sweden, 2024), Modern Greek (Voukoutis et al., 2024), Norwegian (NORA.LLM-Team, 2024), or multiple languages (Xue et al., 2021; Alves et al., 2024; Üstün et al., 2024; Costa-jussà et al., 2022; Martins et al., 2024; Dou et al., 2024; Nguyen et al., 2024; Aryabumi et al., 2024; Dang et al., 2024).

Open Large Language Models. Recent open language models can be broadly divided into *open-source* LLMs and *open-weight* LLMs. The main difference is that open-weight releases include at least a basic description of the training data, as well as the model weights themselves. For open-source LLMs, instead, the (non-trivial) expectation is to have all resources released, including data, training scripts, evaluation scripts, and model weights. We follow previous endeavors such as Pythia (Biderman et al., 2023), OLMo (Groeneveld et al., 2024), Latxa (Etxaniz et al., 2024), and Meltemi (Voukoutis et al., 2024), and release most sources of our training data, including training and evaluation scripts, as well as the model weights.

Danish Language Resources. In-language resources are the fundamental building block for further training an LLM for the Danish language. There are several open-source toolkits for Danish, including models and datasets (Pauli et al., 2021; Enevoldsen et al., 2021). Additionally, there are several Danish-specific large corpora of raw text, such as DaNewsroom (Varab and Schluter, 2020) and Danish Gigaword (Strømberg-Derczynski et al., 2021). Additionally, Danish subsets can be found in public resources built on crawled web data such as CommonCrawl (Wenzek et al., 2020) and CulturaX (Nguyen et al., 2023). In this work, we collect and combine a variety of

sources for wider coverage, before pre-processing them through a joint pipeline.

Danish Large Language Models. Previous endeavors at training LLMs that cover the Danish language include Ciosici and Derczynski (2022), who trained a T5 model (Raffel et al., 2020) for Danish. More recently, within the decoder-only family of models, Munin (Danish-Foundation-Models-Team, 2024) and Viking (SiloAI, 2024) were released. Munin is based on Mistral-7B (v0.1 Jiang et al., 2023) and is further pre-trained on the Danish Gigaword Corpus (Strømberg-Derczynski et al., 2021) containing 1B words. However, the model seems to underperform compared to its base model counterpart, indicating some form of catastrophic forgetting. Viking is based on LLAMA2-7B, and pre-trained from scratch on a mix of English, Finnish, Swedish, Danish, Norwegian, Icelandic and code (SiloAI, 2024). In this work, SNAKMODEL-7B_{instruct} is continuously pre-trained for Danish, and outperforms its original checkpoint, as well as all other currently available Danish models with a comparable size.

3 Data & Pre-processing

3.1 Pre-training

Our Danish pre-training data, as shown in Table 1, initially encompassed 927M documents and 24.6B words, as measured by the Unix `wc` command. The data is sourced from diverse platforms, for which we verify appropriate licensing (wherever possible), and include:

Bookshop (cc-by-4.0). EU Bookshop text from OPUS (Tiedemann, 2012), as integrated by Skadiņš et al. (2014). It contains well-edited, official EU publications across diverse topics, converted automatically from PDFs.

CC-100 (UNK). A cleaned version of a 2018 CommonCrawl dump (Wenzek et al., 2020), reproducing data from Conneau et al. (2020). It consists of web data, filtered using the fastText language classifier (Joulin et al., 2017).

CulturaX (odc-by + cc0). mC4 (v3.1.0) combined with accessible OSCAR corpora (Nguyen et al., 2023).

DaNewsroom (UNK). Scraped from 19 news outlets (Varab and Schluter, 2020), originally for summarization. We use the full news articles instead of summaries.

| DATASET | ORIGINAL | | + FASTTEXT | |
|------------------------|-------------|--------------|-------------|--------------|
| | Docs | Words | Docs | Words |
| Bookshop | 8.65M | 208M | 6.80M | 187M |
| CC-100 | 344M | 7.82B | 256M | 7.16B |
| CulturaX | 449M | 14.8B | 333M | 13.7B |
| DaNewsroom | 24.2M | 391M | 11.3M | 369M |
| Dawiki | 1.70M | 62.4M | 1.20M | 57.3M |
| FTSpeech | 2.03M | 43.3M | 1.69M | 40.9M |
| Gigaword | 62.0M | 1.02B | 39.3M | 898M |
| OpenSubtitles | 30.2M | 207M | 19.6M | 156M |
| Reddit | 4.50M | 73.9M | 2.37M | 64.0M |
| Twitter | 1.69M | 21.9M | 406K | 6.61M |
| TOTAL | 927M | 24.6B | 672M | 22.6B |
| + DEDUPLICATION | | | 350M | 13.6B |

Table 1: **Preprocessing Steps.** Data in number of words using `wc` command. In the **Original** column, we already use a pre-defined Danish slice of the dataset. In the **FastText** column, we apply another round of language identification to the data. In the **Deduplication** row, we combine all data and deduplicate it, which results in around 350M documents and 13.6B words for the pre-training process.

Dawiki (cc-by-sa). Cleaned Wikipedia data from 01-01-2024 (Attardi, 2015).

FTSpeech (FT-OD + FT-TV). A transcription-based corpus from Danish parliamentary data (Kirkedal et al., 2020), used in language modeling due to its large text volume.²

Gigaword (cc0 + cc-by). Danish Gigaword (Strømberg-Derczynski et al., 2021) covers a range of domains including wiki, books, web, and social media data.

OpenSubtitles (UNK). Danish data from OPUS OpenSubtitles (Lison and Tiedemann, 2016; Tiedemann, 2016).³

Reddit (UNK). Danish Reddit data from ConvoKit (Chang et al., 2020), specifically `Denmark.corpus.zip`.

Twitter (MIT). Data from the public Twitter stream,⁴ reclassified using our own pipeline due to inaccurate language labels.

To refine the overall concatenated dataset, we implemented a preprocessing pipeline using `fastText` (Joulin et al., 2017)⁵ for language iden-

²FT-OD and FT-TV refer to Folketing’s open data and Folketing TV license.

³<http://www.opensubtitles.org/>

⁴<https://archive.org/details/twitterstream>

⁵Using the `lid.176.bin` model with a threshold of 0.6.

tification and `text-dedup` (Mou et al., 2023)⁶ for text deduplication. The language identification process eliminated 28% of the documents while retaining 92% of the tokens, indicating that many short documents were removed, where language prediction was less confident. The deduplication step further reduced the corpus by 48% in document count and 40% in token count. We anticipated significant content overlap between CC-100 and CulturaX, which underlines the importance of deduplication in creating a more efficient and representative dataset. These preprocessing steps reduced our dataset to approximately 350M documents with 13.6B words. Following the open LLM approach, we release all scripts used for collecting and processing the data.

3.2 Instruction Tuning

As for most mid-to-low resource languages, Danish (Joshi et al., 2020) currently lacks human-generated instruction tuning data, and instead relies on automatically translated data from English, which itself may be generated by LLMs. From these sources, we select the following three after manually inspecting them for quality:

SkoleGPT (Professionshøjskole, 2024) : A subset of OpenOrca (Lian et al., 2023), which was automatically translated into Danish and filtered for quality, containing 21.6k instruction-output pairs.

Danish OpenHermes (Mabeck, 2024) : A subset of the automatically generated OpenHermes dataset (“Teknium”, 2023), which was automatically translated into Danish. It contains 98.7k instruction-output pairs.

Aya Collection (Singh et al., 2024) : A collection of 44 datasets, which were automatically translated based on instruction templates from fluent speakers. While the underlying Aya Dataset, on which these translations are based, was created by native speakers, the Danish portion of this data contains less than 100 instances, leading us to opt for the translations instead. We use 3.6M instruction-output pairs from the Danish subset of the data.

Together, these data sources sum up to a total of 3.7M instruction-answer pairs, which we train SNAKMODEL-7B_{base} on in Section 4.2.

⁶<https://github.com/ChenghaoMou/text-dedup>

| Parameter | Value |
|-------------------------------------|----------------------|
| <i>Data Split</i> | |
| Training data | 96.9% |
| Validation data | 3.1% |
| <i>Training Configuration</i> | |
| Vocabulary size | 32,000 |
| Context length | 4,096 |
| Training steps | 12,500 |
| Warmup steps | 1,250 |
| Number of epochs | 1 |
| Global batch size | 512 |
| <i>Optimizer Parameters (AdamW)</i> | |
| $\beta_1; \beta_2$ | 0.9; 0.95 |
| ϵ | 10^{-5} |
| Peak learning rate | 1.5×10^{-5} |
| Minimum learning rate | 5×10^{-8} |
| Weight decay | 0.1 |
| Gradient clipping | 1.0 |

Table 2: **Pre-training Hyperparameters and Configuration Details.** We show the hyperparameter details of SNAKMODEL-7B_{base} pre-training.

3.3 Evaluation Framework

For evaluation, we use the SCANDEVAL benchmark (Nielsen, 2023) covering eight tasks. The tasks cover named entity recognition (NER; DANSK by Hvingelby et al., 2020), sentiment analysis (SENTI; AngryTweets by Pauli et al., 2021), linguistic acceptability (LA; ScaLA⁷), abstractive summarization (SUMM; Nordjylland-News by Kinch, 2023), commonsense reasoning (CSR; translated HellaSwag by Zellers et al., 2019), and question answering (QA; ScandiQA⁸). The benchmark also include culture-specific datasets, namely Danske Talemåder (TM; Nielsen, 2023), which prompts for meanings behind common proverbs, and a collection of official Danish Citizenship Tests (CT; Nielsen, 2024). Evaluation metrics differ per task, and are indicated as F_1 , macro-averaged F_1 (mF_1), micro-averaged F_1 (μF_1), BERTScore (BERTS.; Zhang et al., 2020), and Accuracy (Acc.).

4 Model Training

4.1 Language Modeling Pre-training

Training Details. We continuously pre-train from LLAMA2-7B_{base} (Touvron et al., 2023). We show configuration and hyperparameter details

⁷Based on the Universal Dependencies dataset from (Krohnmann and Lyng, 2004).

⁸ScandiQA is a translation of the English MKQA dataset (Longpre et al., 2021) and does not strictly focus on Scandinavian knowledge.

in Table 2. For further pre-training and fine-tuning, we make use of the Megatron-LLM library (Cano et al., 2023), based on the Megatron-LM library.⁹ We use the same tokenizer as LLAMA2-7B, byte-pair encoding (BPE; Sennrich et al., 2016) as implemented in the SentencePiece toolkit (Kudo and Richardson, 2018), with a vocabulary size of 32K subwords. As Danish and English share the same Indo-European language family, we assume large overlap in vocabulary subwords. Hence, we do not re-train nor extend the vocabulary.

Hardware and Emissions. SNAKMODEL-7B_{base} is trained on private infrastructure with one node, containing four NVIDIA A100-PCIe 40GB GPUs. The node has an AMD Epyc 7662 128 Core Processor and 1TB of RAM. Total time of training took 8,928 GPU hours (93 days \times 24 hours \times 4 GPUs) between March–June 2024. The average carbon efficiency was 0.122 $kgCO_2eq/kWh$ during this time in Denmark.¹⁰ This is equivalent to 272.3 $kg CO_2 eq.$ emitted, based on the Machine Learning Impact calculator (Lacoste et al., 2019).¹¹

Loss Trajectories. In Figure 1, we show the continuous pre-training process of SNAKMODEL-7B_{base} in terms of loss curve based on perplexity. The loss shows a declining gain over time. We speculate that the model is close to convergence or that the learning rate is reduced, although previous work has shown that downstream performance can still increase with more training after loss and perplexity have converged (Liu et al., 2023).

Leakage. The training data of LLAMA2-7B is not public. However, since it was released in July 2023 after the ScandEval benchmark, we investigate potential test data leakage by prompting the model for information about the dataset (inspired by Sainz et al., 2023; Balloccu et al., 2024), as well as completions for the first five sentences of each dataset. This process yielded no evidence that the evaluation datasets were included during training.

For SNAKMODEL-7B_{base}, we have access to all training data, such that we can search for 200 random 8-grams from each of our datasets in the raw data. We find that a small amount (6/200) of the tweets from AngryTweets are included in our Twit-

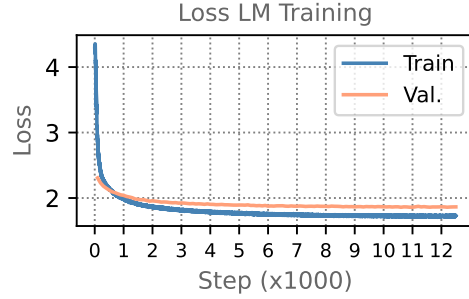


Figure 1: **SNAKMODEL-7B_{base} Pre-training Behaviour.** We report the stable language model loss during training and validation.

ter sample (without labels). The DANSK NER dataset was completely included (without labels), as it was sampled from Gigaword, and many parts of the ScaLA dataset were also included in its original form in GigaWord and CC100. The code for all leakage tests is included in our code repository.

4.2 Instruction Tuning

Starting from SNAKMODEL-7B_{base}, we train our model on the Danish instruction datasets outlined in Section 3.2.

Training Details. For instruction tuning, we opt for the more parameter-efficient low-rank adaptation (LoRA; Hu et al., 2022), to enable faster iterations across multiple ablations (different template formats and base models), and to more easily analyze the intermediate training dynamics (Section 4.3). Nonetheless, we choose a substantially higher-parameter setup than is commonly employed when using LoRA (Hu et al., 2022; Dettmers et al., 2023), in order to approximate full fine-tuning as closely as possible given our computational budget. Specifically, we use rank $r = 128$ adaptation matrices, which are applied to all parameters within the model without quantization (Dettmers et al., 2023). We train for one epoch over our instruction data using the AdamW optimizer with a constant learning rate of 2×10^{-4} , and a global batch size of 64.

Instruction Template. The formatting of instruction-answer pairs is an important design decision with significant downstream impacts (Sclar et al., 2024). For our adaptation context (LLAMA2-7B + Danish), we therefore ablate across three templates: (1) CONCAT, which concatenates instructions and answers; (2) CHAT, which wraps the instruction in special [INST]/[/INST]

⁹<https://github.com/NVIDIA/Megatron-LLM>.

¹⁰According to <https://app.electricitymaps.com/map>.

¹¹<https://mlco2.github.io/impact>.

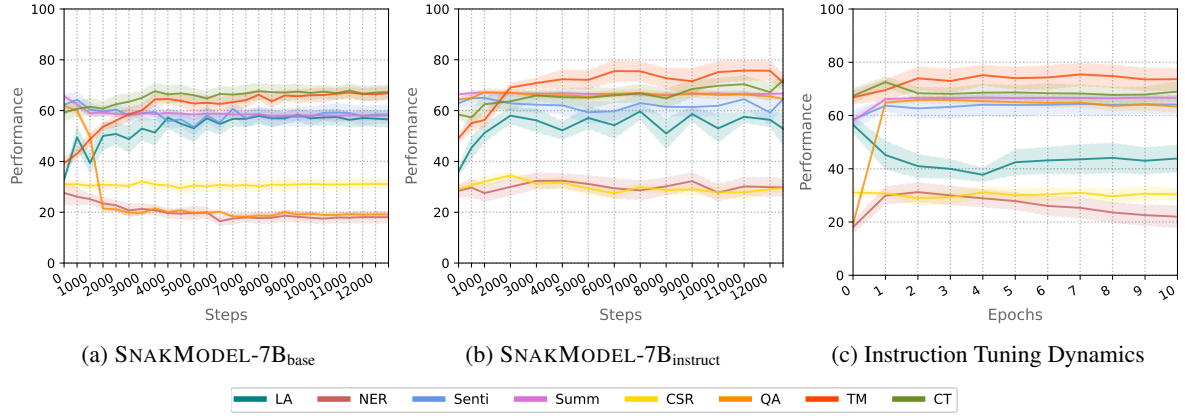


Figure 2: **SNAKMODEL Training Dynamics** of LM pre-training, instruction tuning, and multi-epoch instruction tuning, as measured on the ScandEval (validation) tasks of linguistic acceptability (LA), named entity recognition (NER), sentiment analysis (SENTI), summarization (SUMM), commonsense reasoning (CSR), question answering (QA), proverb meaning (TM), and citizenship tests (CT).

delimiters following LLAMA2-7B_{chat}¹²; (3) ALPACA, following a multi-line format with instruction/input/answer headers (Wang et al., 2023), which we translate into Danish.

Instruction tuning using the CHAT format leads to the highest overall scores on the validation split of our evaluation benchmark (56.37 avg.). CON-CAT performs comparably (55.52 avg.), however we observe that models trained using this template frequently generate continuations to an instruction, instead of an answer. ALPACA performs worst (53.26 avg.), and we observe that when prompting models without correctly terminating the instruction, the CHAT model consistently terminates the instruction on its own (by generating [/INST]), while the ALPACA model often struggles to do so.

4.3 Training Dynamics

We next investigate our models’ intermediate training dynamics to establish how much language modeling and/or instruction tuning are required to obtain a certain level of performance (evaluated according to Section 3.3), and whether these trajectories differ across task types.¹³

Language Modeling. By tracking the validation performance of the non-instruction-tuned SNAKMODEL-7B_{base} checkpoints across pre-training, we aim to identify when the English base

model begins adapting to Danish. Figure 2a shows performance on the Danish ScandEval tasks from start (LLAMA2-7B_{base}) to finish (SNAKMODEL-7B_{base}). For SENTI, SUMM and CSR, performance remains relatively consistent, while for LA, TM and CT performance gradually increases until 4,000–6,000 steps before converging.

Meanwhile, we see performance decreases for NER and QA, with the latter dropping from 61.9% F1 to around 20% within the first 2,000 steps. We attribute these changes to two respective hypotheses: for NER, answers are enforced to be in JSON-format in ScandEval. As our pre-training data consists exclusively of natural language, the model’s output distribution may skew away from tokens such as “{}”, required for this task. For QA, we qualitatively observe that SNAKMODEL-7B_{base} tends to generate continuations to the provided questions, instead of answers. Additionally, it does so in Danish, which may be detrimental to performance, since many answers in QA are English names.

Instruction Tuning. Next, we investigate the effect of applying instruction tuning at different points during Danish pre-training, in order to assess when it starts becoming beneficial. Figure 2b shows the validation performance of intermediate SNAKMODEL-7B_{base} checkpoints after instruction-tuning, i.e., from LLAMA2-7B_{base} + INST_{da} (instruction-tuning on Danish instruction-completion pairs) until our final SNAKMODEL-7B_{instruct} (fully pre-trained SNAKMODEL-7B_{base} + INST_{da}). Once again, performance for most tasks

¹²Note that these delimiters are not split by the tokenizer.

¹³The intermediate checkpoints can be found here: <https://huggingface.co/NLPnorth/snakmodel-7b-base/tree/main> for SNAKMODEL-7B_{base} and <https://huggingface.co/NLPnorth/snakmodel-7b-instruct> for SNAKMODEL-7B_{instruct}.

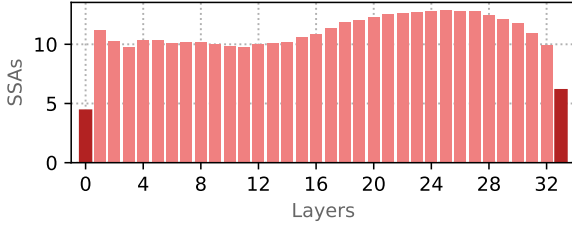


Figure 3: **Layer-wise Weight Divergence of SNAKMODEL-7B_{base}** as measured in total SSAs. Darker bars represent EMB and LMH respectively.

is surprisingly stable throughout training. We further do not observe the same performance drops for NER and QA as during language modeling pre-training, showing that instruction tuning recovers these original functionalities. Additionally, we observe a general performance increase across the board. In particular, performance for LA, TM, and CT climbs and converges after 2,000–5,000 steps of Danish pre-training, and subsequent instruction-tuning. This indicates that training on less than half of our corpus may already be sufficient to obtain close-to-final performance. Interestingly, the largest performance improvements are observed for benchmark tasks based on Danish data, instead of translations (e.g., LA, TM, CT).

In terms of the training dynamics of instruction tuning itself, Figure 2c shows how one epoch of instruction tuning is already sufficient to obtain most performance gains, including the performance recovery of NER and QA. While there may be some benefit to one or two additional instruction tuning epochs, we believe that at this scale, they can be skipped in favor of efficiency. Since the use of duplicate data across epochs has however also been shown to negatively affect downstream performance (Biderman et al., 2023), we leave the exploration of this trade-off to future work.

Weight Divergence Analysis. Lastly, we take a closer look at changes *within* the model to identify which parameters are most strongly affected by Danish language adaptation. To measure weight divergence, we follow Müller-Eberstein et al. (2024) and measure the principal subspace angles (SSAs; Knyazev and Argentati, 2002) of each parameter before and after adaptation ($0^\circ/90^\circ \leftrightarrow$ similar/dissimilar). Across layers, Figure 3 shows how there is a slightly higher rate of change towards the penultimate layers of the model. This may be representative of cross-lingual encoding early in

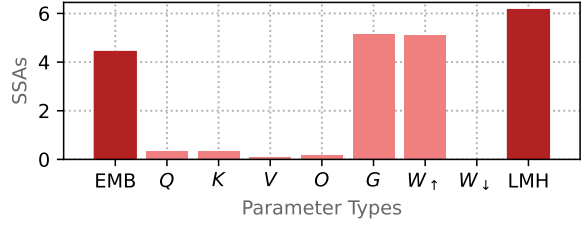


Figure 4: **Parameter-wise Weight Divergence of SNAKMODEL-7B_{base}** as measured in mean SSA. Darker bars represent EMB and LMH respectively.

the model, and subsequent target language specialization in later layers (Wendler et al., 2024).

Figure 4 provides a more granular view of which parameter types are changing within each layer: Most updates per layer appear to be concentrated in the gate G and up-projection W_{\uparrow} of the SwiGLU feed-forward block (Shazeer, 2020), while the down-projection W_{\downarrow} and self-attention parameters (Q, K, V, O) are relatively unaffected. For the self-attention parameters, we hypothesize that this lack of change could be an effect of the relatively high syntactic similarity of English and Danish, requiring less adaptation for in-sequence dependencies. Interestingly, this pattern is also observed when adapting speech recognition models to under-resourced settings (Müller-Eberstein et al., 2024).

The initial embedding layer (EMB) as well as final language modeling head (LMH) also diverge to a comparable degree as G and W_{\uparrow} , which is to be expected given their importance to receiving and generating text in a new language. In terms of token-level changes within EMB and LMH (as measured by the absolute difference of each token row before and after adaptation), we observe larger updates to subwords, which occur both in Danish and other Germanic languages (e.g., “_er”, “_ik”, “_billion”), while subwords in other scripts appear to be least affected. Overall, our findings indicate that future work may be able to train language-specific models more efficiently by focusing exclusively on the EMB, G , W_{\uparrow} and LMH parameters.

5 Final Results and Analysis

Benchmark Results. Using our final model configurations, we present our results on the test split of the Danish portion of ScandEval in Table 3. We compare SNAKMODEL-7B_{instruct} against variants built on the same base model, including the original LLAMA2-7B_{base} and LLAMA2-7B_{chat}. In addition, we train +INST_{da} variants of these English

| TASK →
↓ MODEL | LA
(mF_1) | NER
(μF_1) | SENTI
(mF_1) | SUMM
(BERTS.) | CSR
(Acc.) | QA
(F_1) | TM
(Acc.) | CT
(Acc.) | AVG. |
|--|------------------|----------------------|---------------------|------------------|---------------|-----------------|--------------|--------------|-------------------------------|
| LLAMA2-7B BASED LLMs | | | | | | | | | |
| LLAMA2-7B _{base} | 33.43 | 22.31 | 61.54 | 65.50 | 29.76 | 63.54 | 38.69 | 57.05 | 46.48 |
| LLAMA2-7B _{chat} | 47.42 | 24.63 | 62.35 | 66.15 | 32.24 | 61.34 | 46.67 | 55.18 | 49.50 |
| LLAMA2-7B _{base} + INST _{da} | 36.10 | 28.48 | 62.86 | 66.43 | 29.04 | 64.40 | 49.10 | 58.46 | 49.35 |
| LLAMA2-7B _{chat} + INST _{da} | 43.40 | 29.70 | 65.92 | 65.81 | 30.95 | 62.46 | 57.26 | 55.59 | 51.39 |
| VIKING-7B | 33.67 | 17.18 | 49.48 | 61.96 | 25.11 | 56.29 | 23.97 | 34.90 | 37.82 |
| SNAKMODEL-7B _{base} | 56.28 | 19.91 | 57.42 | 58.95 | 30.47 | 18.52 | 69.14 | 60.93 | 46.45 |
| SNAKMODEL-7B _{instruct} | 52.91 | 29.76 | 66.70 | 66.61 | 29.46 | 64.66 | 71.05 | 71.88 | 56.63^{+10.15} |
| MISTRAL-7B BASED LLMs | | | | | | | | | |
| MISTRAL-7B-v0.1 | 38.38 | 32.66 | 54.53 | 66.47 | 37.39 | 64.55 | 64.50 | 71.56 | 53.76 |
| MUNIN-7B-ALPHA | 53.03 | 28.71 | 43.77 | 67.27 | 42.68 | 63.44 | 83.01 | 77.91 | 57.48 |
| MUNIN-7B-v0.1 DEV0 | 57.02 | 28.74 | 50.72 | 67.89 | 42.17 | 64.41 | 93.45 | 85.82 | 61.28^{+7.52} |

Table 3: **Results (Test) on the ScandEval Benchmark.** We evaluate LLAMA2-7B_{base}, as well as the chat version against SNAKMODEL-7B_{instruct} and other 7B models in ScandEval (best results in blue). In the subsequent rows, we test the same LLAMA2-7B tuned the Danish instruction tuning data (+ INST_{da}). In the final rows, we show the Mistral-based models (best results in orange). We evaluate in F_1 , macro-averaged F_1 (mF_1), micro-averaged F_1 (μF_1), BERTScore (BERTS.; Zhang et al., 2020), and Accuracy (Acc.).

LLAMA2-7B models on the same Danish instruction datasets as SNAKMODEL-7B_{instruct}, in order to isolate the effect of Danish language modeling pre-training. Finally, we include comparisons to the Viking-7B model (SiloAI, 2024) and similarly-sized models based on the Mistral model suite (Jiang et al., 2023; Danish-Foundation-Models-Team, 2024).

Overall, SNAKMODEL-7B_{instruct} outperforms all other LLAMA2-7B-based models, including those with access to the same set of Danish instruction-tuning data, with a final average benchmark score of 56.63. The performance improvements over the English model are particularly pronounced for sub-tasks based on natural Danish data, including LA (33.43 → 52.91), TM (38.69 → 71.05), and CT (57.05 → 71.88). While the Mistral-7B-based models outperform SNAKMODEL-7B_{instruct} by up to 4.65% abs., this approximately matches the base model performance difference between Mistral-7B-v0.1 and LLAMA2-7B_{base} which spans 7.28%.

Qualitative Behaviors. Since ScandEval scores are largely computed using constrained generation, we would like to highlight some qualitative observations from when models generate text without constraint. First, we find that LLAMA2-7B models fail to generate Danish text consistently, even when explicitly prompted to do so (confirming the findings by Puccetti et al., 2024). Since they nonethe-

less achieve non-trivial benchmark scores under constrained generation, we hypothesize, that they obtain some Danish language functionality during their original, primarily English pre-training. Our custom LLAMA2-7B models to which we add Danish instruction tuning (+INST_{da}) generate Danish responses (even when prompted in English), highlighting that a relatively small amount of translated Danish instructions is sufficient to bias models towards generating output in a new language. Nonetheless, the fact that SNAKMODEL-7B_{instruct}, which is trained on non-translated Danish text outperforms the models trained on translated data, highlights the importance of curating high-quality native-language data for the adaptation target.

6 Guidance for Future Work

From our final evaluation, as well as our analysis of the training dynamics of SNAKMODEL-7B_{instruct}, we next consolidate some guidance for future work adapting English LLMs to languages with similar linguistic properties and resource constraints.

Data. As we found large overlaps across data sources, as well as large amounts of non-Danish or irrelevant data (Section 3), applying stringent pre-processing standards is important when working with smaller languages—especially when automatic filtering tools may be biased towards larger, related languages (e.g., Swedish).

Training. Our training dynamics analysis (Section 4.3) showed that despite our total 13.6B word pre-training corpus, applying instruction tuning after 2,000–5,000 steps of Danish pre-training (i.e., less than half of the corpus) may already be sufficient to obtain close-to-final performance. For instruction tuning itself, one epoch over translated data appears to be sufficient to amplify instruction-following functionalities in the target language. Nonetheless, training on non-translated target language data is important to improve performance on more culturally specific tasks based on native data (i.e., LA, TM, and CT).

Finally, our weight divergence analysis revealed that most parameter updates are consolidated in the embeddings, feed-forward up-projections, and language modeling head. As English and Danish share a relatively similar syntactic structure, languages with more distinctive typologies may nonetheless exhibit larger changes to the self-attention parameters. For model adaption across a comparable typological distance as English and Danish however, focusing training efforts on the aforementioned parameter types—in addition to employing existing parameter-efficient fine-tuning techniques (e.g., Hu et al., 2022; Dettmers et al., 2023)—may therefore yield even higher efficiency gains.

7 Conclusion

In this work, we introduced the SNAKMODEL suite, which includes a 7B-parameter base and instruction-tuned LLM for Danish, in addition to its pre-training and instruction-tuning data, intermediate checkpoints, and evaluation. By analyzing design decisions related to data curation and training dynamics, we further consolidated guidelines for future work adapting LLMs to new languages, to foster research not just in Danish, but in language communities with similar resource constraints.

Limitations

What Went Wrong and What Decisions We Took. Our training process encountered several challenges across multiple runs. In Run 1, we began by restarting training from the LLAMA2-7B checkpoint using the identical learning rate the original model had been trained on. However, we faced gradient explosion at iteration 2,031, which we attempted to mitigate through gradient clipping. Despite this effort, server crashes at step 3,500 and persistent gradient explosions forced us to halt the

run after approximately 46 days, with a final language model loss of ± 1.77 . For Run 2, we halved the peak learning rate to 1.5×10^{-4} and adjusted other parameters, but gradient explosion recurred at step 1,390, leading us to terminate the run after about 10 days with a final loss of ± 1.79 . In Run 3, we significantly reduced the peak learning rate to 1.5×10^{-5} , reasoning that as we were continuing pre-training, we should aim for a rate lower than Llama2’s final learning rate. This approach has shown effective, with the training reaching iteration 12,500 after approximately 93 days and achieving a language model loss of ± 1.72 .

Acknowledgments

First, we would like to thank Barbara Plank for allowing us to use her compute hardware for this period of time. Additionally, this work was impossible without the stable High Performance Compute cluster at the IT University of Copenhagen, being able to train a model for ± 90 days without a single interruption is extraordinary. Second, we thank Ahmet Üstun for giving us invaluable and concrete comments on hyperparameter setup for continuous pre-training. Last, we would also like to thank the reviewers for their valuable comments. Elisa Bassignana is supported by a research grant (VIL59826) from VILLUM FONDEN. Mike Zhang is supported by a research grant (VIL57392) from VILLUM FONDEN.

References

- Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *ArXiv preprint*, abs/2404.14219.
- AI-Sweden. 2024. Ai-sweden-models/llama-3-8b.
- 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.
- Duarte M Alves, José Pombal, Nuno M Guerreiro, Pedro H Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, et al. 2024. Tower: An open multilingual large language model for translation-related tasks. *ArXiv preprint*, abs/2402.17733.

- Viraat Aryabumi, John Dang, Dwarak Talupuru, Saurabh Dash, David Cairuz, Hangyu Lin, Bharat Venkitesh, Madeline Smith, Jon Ander Campos, Yi Chern Tan, et al. 2024. Aya 23: Open weight releases to further multilingual progress. *ArXiv preprint*, abs/2405.15032.
- Giuseppe Attardi. 2015. Wikiextractor. <https://github.com/attardi/wikiextractor>.
- Andrea Bacciu, Cesare Campagnano, Giovanni Trapolini, and Fabrizio Silvestri. 2024. DanteLLM: Let’s push Italian LLM research forward! In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4343–4355, Torino, Italia. ELRA and ICCL.
- Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *ArXiv preprint*, abs/2309.16609.
- Simone Balloccu, Patrícia Schmidová, Mateusz Lango, and Ondrej Dusek. 2024. Leak, cheat, repeat: Data contamination and evaluation malpractices in closed-source LLMs. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 67–93, St. Julian’s, Malta. Association for Computational Linguistics.
- Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O’Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR.
- 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. epfilm megatron-llm.
- Yong Cao, Li Zhou, Seolhwa Lee, Laura Cabello, Min Chen, and Daniel Hershcovich. 2023. Assessing cross-cultural alignment between ChatGPT and human societies: An empirical study. In *Proceedings of the First Workshop on Cross-Cultural Considerations in NLP (C3NLP)*, pages 53–67, Dubrovnik, Croatia. Association for Computational Linguistics.
- Jonathan P. Chang, Caleb Chiam, Liye Fu, Andrew Wang, Justine Zhang, and Cristian Danescu-Niculescu-Mizil. 2020. ConvoKit: A toolkit for the analysis of conversations. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 57–60, 1st virtual meeting. Association for Computational Linguistics.
- Morten H. Christiansen, Kristian Tylén, Riccardo Fusaroli, Dorthe Bleses, Anders Højen, Fabio Trecca, Christina Dideriksen, and Byurakn Ishkhanyan. 2023. The puzzle of danish.
- Manuel R. Ciosici and Leon Derczynski. 2022. Training a t5 using lab-sized resources.
- 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.
- Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *ArXiv preprint*, abs/2207.04672.
- John Dang, Shivalika Singh, Daniel D’souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, et al. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *ArXiv preprint*, abs/2412.04261.
- Danish-Foundation-Models-Team. 2024. Releasing munin 7b alpha - a danish llm.
- 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.
- Christina Dideriksen, Morten H. Christiansen, Mark Dingemanse, Malte Højmark-Bertelsen, Christer Johansson, Kristian Tylén, and Riccardo Fusaroli. 2023. Language-specific constraints on conversation: Evidence from danish and norwegian. *Cognitive Science*, 47(11). Publisher Copyright: © 2023 Cognitive Science Society LLC.
- Longxu Dou, Qian Liu, Guangtao Zeng, Jia Guo, Jiahui Zhou, Wei Lu, and Min Lin. 2024. Sailor: Open language models for south-east asia. *ArXiv preprint*, abs/2404.03608.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *ArXiv preprint*, abs/2407.21783.
- Ariel Ekgren, Amaru Cuba Gyllensten, Felix Stoltenwerk, Joey Öhman, Tim Isbister, Evangelia Gogoulou, Fredrik Carlsson, Judit Casademont, and Magnus Sahlgren. 2024. GPT-SW3: An autoregressive language model for the Scandinavian languages.

- In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7886–7900, Torino, Italia. ELRA and ICCL.
- Kenneth Enevoldsen, Lasse Hansen, and Kristoffer Nielbo. 2021. Dacy: A unified framework for danish nlp. *ArXiv preprint*, abs/2107.05295.
- Julen Etxaniz, Oscar Sainz, Naiara Miguel, Itziar Aldabe, German Rigau, Eneko Agirre, Aitor Ormazabal, Mikel Artetxe, and Aitor Soroa. 2024. Latxa: An open language model and evaluation suite for Basque. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14952–14972, Bangkok, Thailand. Association for Computational Linguistics.
- Dirk Groeneveld, Iz Beltagy, Evan Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muenighoff, Aakanksha Naik, Crystal Nam, Matthew Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, William Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah Smith, and Hannaneh Hajishirzi. 2024. OLMo: Accelerating the science of language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15789–15809, Bangkok, Thailand. 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, Online. Association for Computational Linguistics.
- Xiaochuang Han and Jacob Eisenstein. 2019. Unsupervised domain adaptation of contextualized embeddings for sequence labeling. 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 4238–4248, Hong Kong, China. Association for Computational Linguistics.
- Daniel Hershcovich, Stella Frank, Heather Lent, Miryam de Lhoneux, Mostafa Abdou, Stephanie Brandl, Emanuele Bugliarello, Laura Cabello Piqueras, Ilias Chalkidis, Ruixiang Cui, Constanza Fierro, Katerina Margatina, Phillip Rust, and Anders Søgaard. 2022. Challenges and strategies in cross-cultural NLP. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6997–7013, Dublin, Ireland. 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 *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.
- Rasmus Hvingelby, Amalie Brogaard Pauli, Maria Barrett, Christina Rosted, Lasse Malm Lidegaard, and Anders Søgaard. 2020. DaNE: A named entity resource for Danish. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4597–4604, Marseille, France. European Language Resources Association.
- Jaavid J, Raj Dabre, Aswanth M, Jay Gala, Thanmay Jayakumar, Ratish Puduppully, and Anoop Kunchukuttan. 2024. RomanSetu: Efficiently unlocking multilingual capabilities of large language models via Romanization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15593–15615, Bangkok, Thailand. 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*, abs/2310.06825.
- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.
- Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.
- Oliver Kinch. 2023. Nordjylland news summarization. Accessed on 22-08-2024.
- Andreas Kirkedal, Marija Stepanovic, and Barbara Plank. 2020. FT speech: Danish parliament speech corpus. In *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, pages 442–446. ISCA.
- Andrew V Knyazev and Merico E Argentati. 2002. Principal angles between subspaces in an A-based scalar product: algorithms and perturbation estimates. *SIAM Journal on Scientific Computing*, 23(6):2008–2040.

- Matthias Trautner Kromann and Stine Kern Lynge. 2004. The danish dependency treebank v. 1.0.
- Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.
- Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. 2019. Quantifying the carbon emissions of machine learning. *ArXiv preprint*, abs/1910.09700.
- Teven Le Scao, Angela Fan, Christopher Akiki, Elie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2023. Bloom: A 176b-parameter open-access multilingual language model.
- 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.
- LeoLM-Team. 2024. Leolm: Igniting german-language llm research.
- Wing Lian, Bley Goodson, Eugene Pentland, Austin Cook, Chanvichet Vong, and “Teknium”. 2023. Openorca: An open dataset of gpt augmented flan reasoning traces. <https://huggingface.co/Open-Orca/OpenOrca>.
- Pierre Lison and Jörg Tiedemann. 2016. OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).
- Hong Liu, Sang Michael Xie, Zhiyuan Li, and Tengyu Ma. 2023. Same pre-training loss, better downstream: Implicit bias matters for language models. In *International Conference on Machine Learning*, pages 22188–22214. PMLR.
- Shayne Longpre, Yi Lu, and Joachim Daiber. 2021. MKQA: A linguistically diverse benchmark for multilingual open domain question answering. *Transactions of the Association for Computational Linguistics*, 9:1389–1406.
- Risto Luukkonen, Ville Komulainen, Jouni Luoma, Anni Eskelinen, Jenna Kanerva, Hanna-Mari Kupari, Filip Ginter, Veronika Laippala, Niklas Muennighoff, Aleksandra Piktus, Thomas Wang, Nouamane Tazi, Teven Scao, Thomas Wolf, Osmo Suominen, Samuli Sairanen, Mikko Merioksa, Jyrki Heinonen, Aija Vahtola, Samuel Antao, and Sampo Pyysalo. 2023. FinGPT: Large generative models for a small language. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2710–2726, Singapore. Association for Computational Linguistics.
- Magnus Mabeck. 2024. Danish openhermes. <https://huggingface.co/datasets/Mabeck/danish-OpenHermes>.
- Pedro Henrique Martins, Patrick Fernandes, João Alves, Nuno M. Guerreiro, Ricardo Rei, Duarte M. Alves, José Pombal, Amin Farajian, Manuel Faysse, Mateusz Klimaszewski, Pierre Colombo, Barry Haddow, José G. C. de Souza, Alexandra Birch, and André F. T. Martins. 2024. Eurollm: Multilingual language models for europe.
- Chenghao Mou, Chris Ha, Kenneth Enevoldsen, and Peiyuan Liu. 2023. Chenghaomou/text-dedup: Reference snapshot.
- Max Müller-Eberstein, Dianna Yee, Karren Yang, Gautam Varma Mantena, and Colin Lea. 2024. Hypernetworks for Personalizing ASR to Atypical Speech. *Transactions of the Association for Computational Linguistics*, 12:1182–1196.
- Tarek Naous, Michael Ryan, Alan Ritter, and Wei Xu. 2024. Having beer after prayer? measuring cultural bias in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16366–16393, Bangkok, Thailand. Association for Computational Linguistics.
- Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 9–14, Online. Association for Computational Linguistics.
- 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 preprint*, abs/2309.09400.
- Xuan-Phi Nguyen, Wenxuan Zhang, Xin Li, Mahani Aljunied, Zhiqiang Hu, Chenhui Shen, Yew Ken Chia, Xingxuan Li, Jianyu Wang, Qingyu Tan, Liying Cheng, Guanzheng Chen, Yue Deng, Sen Yang, Chaoqun Liu, Hang Zhang, and Lidong Bing. 2024. SeaLLMs - large language models for Southeast Asia. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 294–304, Bangkok, Thailand. Association for Computational Linguistics.
- Dan Nielsen. 2023. ScandEval: A benchmark for Scandinavian natural language processing. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 185–201, Tórshavn, Faroe Islands. University of Tartu Library.

- Dan Saattrup Nielsen. 2024. Danish citizen test. Accessed on 22-08-2024.
- NORA.LLM-Team. 2024. Instruction-tuned normistral-7b-warm. Accessed on 17-10-2024.
- Amalie Brogaard Pauli, Maria Barrett, Ophélie Lacroix, and Rasmus Hvingelby. 2021. DaNLP: An open-source toolkit for Danish natural language processing. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 460–466, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- Københavns Professionshøjskole. 2024. Skolegpt instruct. <https://huggingface.co/datasets/kobprof/skolegpt-instruct>.
- Giovanni Puccetti, Anna Rogers, Chiara Alzetta, Felice Dell’Orletta, and Andrea Esuli. 2024. AI ‘news’ content farms are easy to make and hard to detect: A case study in Italian. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15312–15338, Bangkok, Thailand. 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. *J. Mach. Learn. Res.*, 21:140:1–140:67.
- Rakuten Group, Aaron Levine, Connie Huang, Chenguang Wang, Eduardo Batista, Ewa Szymanska, Hongyi Ding, Hou Wei Chou, Jean-François Pesiot, Johaness Effendi, et al. 2024. Rakutenai-7b: Extending large language models for japanese. *ArXiv preprint*, abs/2403.15484.
- Edwin Rijgersberg and Bob Lucassen. 2023. Geitje: een groot open nederlands taalmodel.
- Oscar Sainz, Jon Ander Campos, García-Ferrero Iker, Julen Etxaniz, and Eneko Agirre. 2023. Did ChatGPT cheat on your test?
- Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models’ sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations*.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.
- Noam Shazeer. 2020. Glu variants improve transformer. *ArXiv preprint*, abs/2002.05202.
- SiloAI. 2024. Viking 7b/13b/33b: Sailing the nordic seas of multilinguality.
- Shivalika Singh, Freddie Vargus, Daniel D’souza, Börje Karlsson, Abinaya Mahendiran, Wei-Yin Ko, Herumb Shandilya, Jay Patel, Devidas Mataciunas, Laura O’Mahony, Mike Zhang, Ramith Hetiarachchi, Joseph Wilson, Marina Machado, Luisa Moura, Dominik Krzemiński, Hakimeh Fadaei, Irem Ergun, Ifeoma Okoh, Aisha Alaagib, Oshan Mudannayake, Zaid Alyafeai, Vu Chien, Sebastian Ruder, Surya Guthikonda, Emad Alghamdi, Sebastian Gehrmann, Niklas Muennighoff, Max Bartolo, Julia Kreutzer, Ahmet Üstün, Marzieh Fadaee, and Sara Hooker. 2024. Aya dataset: An open-access collection for multilingual instruction tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11521–11567, Bangkok, Thailand. Association for Computational Linguistics.
- Raivis Skadiņš, Jörg Tiedemann, Roberts Rozis, and Daiga Deksnė. 2014. Billions of parallel words for free: Building and using the EU bookshop corpus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14)*, pages 1850–1855, Reykjavik, Iceland. European Language Resources Association (ELRA).
- Leon Strömberg-Derczynski, Manuel Ciosici, Rebekah Baglini, Morten H. Christiansen, Jacob Aarup Dalsgaard, Riccardo Fusaroli, Peter Juel Henriksen, Rasmus Hvingelby, Andreas Kirkedal, Alex Speed Kjeldsen, Claus Ladefoged, Finn Årup Nielsen, Jens Madsen, Malte Lau Petersen, Jonathan Hvithamar Rystrom, and Daniel Varab. 2021. The Danish Gigaword corpus. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 413–421, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.
- Águila Team. 2023. Introducing aguila, a new open-source llm for spanish and catalan.
- “Teknium”. 2023. OpenHermes dataset. <https://huggingface.co/datasets/teknium/openhermes>.
- Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).
- Jörg Tiedemann. 2016. Finding alternative translations in a large corpus of movie subtitle. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 3518–3522, Portorož, Slovenia. European Language Resources Association (ELRA).
- 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.

- Fabio Trecca, Kristian Tylén, Anders Højen, and Morten H. Christiansen. 2021. Danish as a window onto language processing and learning. *Language Learning*, 71(3):799–833.
- Ahmet Üstün, Viraat Aryabumi, Zheng Yong, Wei-Yin Ko, Daniel D’souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, Freddie Vargus, Phil Blunsom, Shayne Longpre, Niklas Muennighoff, Marzieh Fadaee, Julia Kreutzer, and Sara Hooker. 2024. Aya model: An instruction fine-tuned open-access multilingual language model. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15894–15939, Bangkok, Thailand. Association for Computational Linguistics.
- Bram Vanroy. 2024. Fietje 2: An open and efficient llm for dutch.
- Daniel Varab and Natalie Schluter. 2020. DaNewsroom: A large-scale Danish summarisation dataset. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6731–6739, Marseille, France. European Language Resources Association.
- Leon Voukoutis, Dimitris Roussis, Georgios Paraskevopoulos, Sokratis Sofianopoulos, Prokopis Prokopidis, Vassilis Papavasileiou, Athanasios Katsamanis, Stelios Piperidis, and Vassilis Katsouras. 2024. Meltemi: The first open large language model for greek.
- Wenxuan Wang, Wenxiang Jiao, Jingyuan Huang, Ruyi Dai, Jen-tse Huang, Zhaopeng Tu, and Michael Lyu. 2024. Not all countries celebrate thanksgiving: On the cultural dominance in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6349–6384, Bangkok, Thailand. Association for Computational Linguistics.
- Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.
- Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in English? on the latent language of multilingual transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394, Bangkok, Thailand. Association for Computational Linguistics.
- Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.
- 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*, pages 483–498, Online. Association for Computational Linguistics.
- Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. 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.
- 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. *Advances in Neural Information Processing Systems*, 36:46595–46623.

Got Compute, but No Data: Lessons From Post-training a Finnish LLM

Elaine Zosa¹ Ville Komulainen² Sampo Pyysalo²

¹Silo AI, Finland ²TurkuNLP, University of Turku, Finland
firstname.lastname@{silo.ai, utu.fi}

Abstract

As LLMs gain more popularity as chatbots and general assistants, methods have been developed to enable LLMs to follow instructions and align with human preferences. These methods have found success in the field, but their effectiveness has not been demonstrated outside of high-resource languages. In this work, we discuss our experiences in post-training an LLM for instruction-following for English and Finnish. We use a multilingual LLM to translate instruction and preference datasets from English to Finnish. We perform instruction tuning and preference optimization in English and Finnish and evaluate the instruction-following capabilities of the model in both languages. Our results show that with a few hundred Finnish instruction samples we can obtain competitive performance in Finnish instruction-following. We also found that although preference optimization in English offers some cross-lingual benefits, we obtain our best results by using preference data from both languages. We release our model, datasets, and recipes under open licenses at <https://huggingface.co/LumiOpen/Poro-34B-chat-OpenAssistant>.

1 Introduction

Foundational LLMs are language completion models that need to be finetuned after pretraining to be able to respond to user questions and follow instructions (Ouyang et al., 2022). This post-training process involves *supervised finetuning* where the model is trained to act as an assistant by training on a dataset of prompt-response pairs. *Preference optimization* further aligns the model

to human preferences such as helpfulness, harmlessness, and honesty (Bai et al., 2022). These methods have resulted in LLMs becoming more capable of answering complex questions involving reasoning, coding, math, and science (e.g., Dubey et al., 2024; Jiang et al., 2024; Team et al., 2024). The effectiveness of these methods, however, have not been demonstrated for smaller and less-resourced languages, such as Finnish.

One of the major challenges we face in post-training in smaller languages is the scarcity of training data. The situation is even more challenging for commercial settings as most of the datasets available today are generated by LLMs with restrictive licenses. The availability of evaluation benchmarks for chat models in small languages is also an issue. Popular benchmarks such as MT-Bench (Zheng et al., 2024) and IFEval (Zhou et al., 2023) are designed for English models and have not been adapted for use in a multilingual setting.¹

In this paper, we discuss our experiences in post-training an LLM in Finnish and English. We use the LLM that we want to finetune to machine-translate instruction and preference datasets into Finnish. We use a commercial machine-translation service to translate a widely-used instruction-following evaluation (IFEval) benchmark into Finnish. We experimented with different combinations of Finnish and English data in instruction tuning and preference optimization. We also experimented with different methods to improve vanilla instruction tuning.

2 Related Work

The post-training of base LLMs, popularised in InstructGPT (Ouyang et al., 2022), can be broadly divided into two categories: instruction tuning and preference optimization. Instruction tuning,

¹While revising this paper, a multilingual, multi-turn IFEval was released (He et al., 2024), but it does not include Finnish.

also known as supervised finetuning (SFT), trains a base LLM to answer questions and follow instructions by training on a dataset of prompt-response pairs with a language modeling objective. Preference optimization further improves the model’s ability to follow conversations and teaches a model to generate responses that align with human preferences by showing the model samples of desirable and undesirable responses (or a ranking of responses). Direct preference optimization (DPO; Rafailov et al. (2024)) is a reward-free preference optimization technique that optimizes directly on the preference data and does not require training a separate reward model. It is a popular alternative to reward-based methods such as proximal policy optimization (PPO; Schulman et al. (2017)) because it is less computationally expensive and achieves promising results.

Post-training LLMs in a multilingual setting is an under-explored topic (Üstün et al., 2024; Lai et al., 2023; Martins et al., 2024). Previous studies have experimented with monolingual and multilingual instruction tuning of multilingual base LLMs (Shaham et al., 2024; Chen et al., 2024). These studies show that monolingual instruction tuning transfers some instruction-following capability to the other languages in the model but is dependent on the amount of multilingual data that the base LLM was trained on. A few studies have investigated multilingual preference optimization (Lai et al., 2023; Dang et al., 2024). Lai et al. generated synthetic preference datasets for 26 languages and performed reward-based preference optimization on BLOOM and Llama 7B models. Their results show that preference optimization offers a slight improvement over SFT. Dang et al., however, point out that these preference-optimized models still underperform compared to massively multilingual LLMs that are finetuned only with SFT.

The scarcity of instruction and preference datasets is a major challenge in post-training LLMs for smaller languages. Previous efforts to assemble finetuning datasets through machine translation, crowd-sourcing, and synthetic data generation include (Üstün et al., 2024; Lai et al., 2023; Dang et al., 2024). Evaluating open-ended responses of chat models in small languages is also a challenge. Previous studies have investigated using LLM-as-a-judge in multilingual settings but these studies focused on standard

NLP tasks such as summarization and question-answering (Hada et al., 2024; Ahuja et al., 2023).

3 Experimental setup

We use Poro 34B as the base LLM (Luukkonen et al., 2024). Poro is trained on 1T tokens of English, Finnish, and code, with 129B tokens for Finnish. We use the Transformer Reinforcement Learning library (TRL; (von Werra et al., 2020)) for instruction tuning and preference optimization. We finetune all of the model parameters in our experiments.²

We use 32 AMD MI250X GPUs in our experiments. For SFT, we use a micro batch size of 4 and a gradient accumulation step of 1, resulting in a global batch size of 128. We use a learning rate of $2e-5$ with a warmup rate of 0.1 and finetune for 3 epochs. For DPO, we use a global batch size of 64, learning rate of $5e-7$, warmup rate of 0.1, and train for 5 epochs.

4 Datasets

SFT We use a curated version OpenAssistant 2 (OASST2; (Köpf et al., 2024)) containing the top-ranked English conversations. This dataset has 4,692 samples.³

We translate OASST2 into Finnish using Poro with few-shot prompting. Poro has been shown to produce higher-quality Finnish translations compared to other open MT systems (Luukkonen et al., 2024). For this reason, we did not experiment with translations from other open MT models and focus our efforts on the Poro-translated dataset. We use heuristics to clean up the translations. After post-translation cleaning, our OASST2 Finnish data has 4,399 samples.

DPO We use the HelpSteer2 preference dataset (Wang et al., 2024), which consists of publicly-sourced prompts and LLM-generated completions⁴. We use the helpfulness scores included in the dataset to obtain 7,221 preference pairs (chosen and rejected responses). We also

²We experimented with LoRA finetuning (Hu et al., 2021), but our results indicated that full finetuning achieved better performance.

³The curated dataset is https://huggingface.co/datasets/sablo/oasst2_curated. The full dataset is <https://huggingface.co/datasets/OpenAssistant/oasst2>.

⁴We initially chose this dataset because it has a commercially-friendly license. Recently, however, Lambert et al. pointed out that HelpSteer2 includes ShareGPT prompts which have a questionable legal provenance.

translate this dataset into Finnish using Poro. After post-translation cleaning, we end up with 6,037 preference pairs in our Finnish HelpSteer2 dataset.

5 Evaluation

We use the Instruction Following (IFEval) benchmark to evaluate instruction-following performance (Zhou et al., 2023). IFEval has 541 prompts where a prompt contains verifiable instructions that can be checked with a deterministic program, circumventing the need of an LLM or human as judge. Examples of instructions include adding keywords to the response, formatting the response in JSON, or responding in a specified language.⁵

We translate the IFEval prompts into Finnish using DeepL⁶. IFEval has 31 prompts that require the response language to be in a language other than English. We exclude these prompts for this work due to Poro being constrained to only English and Finnish. We report the results for the remaining 510 prompts only. IFEval reports strict accuracy and loose accuracy where loose accuracy accepts minor transformations in the responses. For the sake of clarity, we report only the strict accuracy in this work.

We run evaluations through the LM Evaluation Harness (Gao et al., 2024). The translated IFEval is available at https://huggingface.co/datasets/LumiOpen/ifeval_mt.

6 Experiments

Multilingual SFT Finnish instruction data is more difficult to obtain compared to English; therefore, we want to investigate how the amount of Finnish instruction data affects performance. We construct data mixes from the English and Finnish OASST2 datasets such that we start with just the English data and gradually introduce more Finnish samples into the mix starting from 10% of the Finnish data and then going up to 100%. We call these data mixes `en-fi-Xpct` (i.e., the data mix with just the English samples is called `en-fi-0pct` while the data mix with all the English and Finnish samples is called `en-fi-100pct`). By default, we do not mask prompts during training (i.e., we incorporate the

⁵See the IFEval paper for the complete list of instructions and their descriptions.

⁶<https://www.deepl.com/>

| data mix | EN (%) | FI (%) | Resp lang (%) |
|--------------|--------------|--------------|---------------|
| en-fi-0pct | 36.39 | 31.41 | 47.45 |
| en-fi-10pct | 39.97 | 32.69 | 90.00 |
| en-fi-20pct | 37.67 | 28.60 | 93.52 |
| en-fi-40pct | 39.20 | 30.90 | 96.27 |
| en-fi-60pct | 39.20 | 32.95 | 94.90 |
| en-fi-80pct | 38.56 | 33.84 | 96.27 |
| en-fi-100pct | 39.97 | 34.48 | 95.68 |

Table 1: Instruction-level accuracy on English and Finnish IFEval of the SFT models trained on different data mixes. Response language refers to the proportion of responses classified as Finnish for the Finnish IFEval.

losses from the prompt and completion tokens). We train SFT models on all the data mixes using the same hyperparameters.

Improving vanilla SFT We investigate whether we can improve SFT by adding noise to the word embeddings in the instruction data (NEF-Tune; Jain et al. (2024)). We also experiment with *prompt masking* where the loss is calculated only on the completion tokens. Our baseline for these experiments is the SFT model trained on the `en-fi-100pct` data mix.

Multilingual DPO We opt to use DPO for preference tuning as it has been found to fare better in IFEval, in addition to being more stable and requiring less compute (Dubey et al., 2024). We tuned the β parameter of DPO with values $\{0.01, 0.05, 0.1\}$ and found $\beta = 0.05$ to be optimal. We experiment with using either the English or Finnish datasets and using both. As our baseline, we use the SFT model trained on the `en-fi-100pct` data mix.

7 Results and Discussion

Multilingual SFT In Table 1, we show the instruction-level accuracy of the SFT models on the English and Finnish IFEval. We also show the proportion of responses to Finnish IFEval that are in Finnish as classified by `langdetect`⁷. For English IFEval, the performance is comparable across the data mixes which is expected because the different data mixes contain same number of English samples. For Finnish, the best performance is from the data mix with all the English

⁷<https://pypi.org/project/langdetect/>

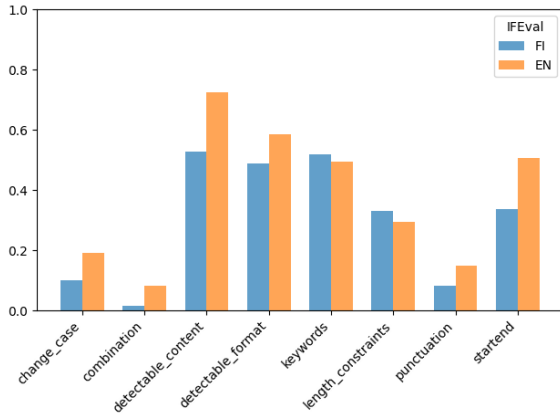


Figure 1: Accuracy by instruction category on English and Finnish IFEval of SFT model trained on the en-fi-100pct data mix.

and Finnish samples (en-fi-100pct). However, compared with the data mix with just 10% of the Finnish data (around 400 samples), the difference is less than 2 percentage points. If we finetune using only the English data, the resulting model can still follow Finnish instructions but less than half of the responses are in Finnish, which is not desirable.

For the response language, SFT models trained on data mixes containing 20% and above of the Finnish data have comparable rates of Finnish responses of over 93%. We reviewed the 22 responses from the en-fi-100pct model that were classified as *not* Finnish. We found that 19 of them are in Finnish but mixed with other languages such as English and German. We also found that the model tends to respond in a mixture of English and Finnish when asked to respond in a specific format, such as JSON or XML. This is likely because JSON tends to be treated as code in the instruction dataset and our translation pipeline did not translate code blocks which sometimes include comments in English.

Figure 1 shows the accuracy by instruction category of the en-fi-100pct model. The model struggles most with the combination category—this category includes combined instructions such as giving two responses that are separated by a given separator or repeating the prompt without modification before giving the response proper. The poor performance is probably because the instruction contains multiple steps that must be followed in order to give a correct response. For instance, if the model gives two responses but these

| Model | EN (%) | FI (%) | Resp lang (%) |
|----------------|--------------|--------------|---------------|
| baseline | 39.97 | 34.48 | 95.68 |
| prompt masking | 39.84 | 32.56 | 96.27 |
| NEFTune | 38.05 | 32.69 | 96.47 |
| DPO-en | 43.55 | 36.65 | 94.70 |
| DPO-fi | 41.76 | 36.01 | 95.49 |
| DPO-both | 44.69 | 37.80 | 95.49 |

Table 2: Instruction-level accuracy on English and Finnish IFEval English for the prompt masking, NEFTune, and DPO experiments. The baseline is an SFT model trained on the en-fi-100pct data mix.

responses are numbered instead of separated by an asterisk, the response is considered incorrect. The model also struggles with punctuation instructions such as avoiding the use of commas likely because texts without commas are rare in the dataset.

Overall, the results indicate that finetuning with a few hundred Finnish instruction samples achieves results close to finetuning with ten times that amount. In terms of instruction types, the model struggles with multi-step instructions and unusual instruction types. Previous work has indicated that carefully curating the instruction dataset is vital to a strong SFT model (Zhou et al., 2024). In future, we aim for a smaller but higher-quality data by, for instance, removing highly similar prompts, diversifying tasks, and curating the sources of the samples.

Prompt masking and NEFTune In Table 2 we show the accuracy of the models trained with prompt masking and NEFTune compared to the baseline model.

In our experiments, models trained with NEFTune fail to achieve better scores compared to the plain vanilla SFT baseline. As noted by Jain et al., the performance of NEFTune was found to be dataset dependent. One key difference in our study is that we train on a multi-turn dataset. We leave further examination of NEFTune and other noise augmentation techniques for future work. We find that prompt masking does not improve over the baseline. This result is in line with findings from Shi et al. where they show that incorporating the loss from the prompt is beneficial for smaller datasets such as LIMA (Zhou et al., 2024) with 1,030 examples.

Multilingual DPO Table 2 shows the results from our DPO experiments compared to the SFT model. The model optimized only on English preference data (DPO-en) improved performance on English IFEval by around 3 percentage points and also showed some improvement in Finnish IFEval. This provides further evidence that preference optimization in English benefits other languages in the model (Dang et al., 2024). The DPO model trained only on Finnish (DPO-fi), on the other hand, showed smaller improvements on both English and Finnish IFEval and, in fact, has slightly lower performance than DPO-en on the Finnish benchmark. The model trained on both languages (DPO-both) achieved the best performance on both benchmarks but compared to DPO-en, the improvements are marginal.

In terms of the response language, DPO did not improve the Finnish response rates compared to the SFT model. This might be because we optimized the model on monolingual preference pairs (the chosen and rejected responses are in the same language). Improving the response language of multilingual models through preference optimisation is an area we will explore in future work.

8 Conclusions and Future Work

In this work we share our findings from post-training Poro 34B in English and Finnish. Due to the scarcity of Finnish post-training datasets we opted to machine-translate instruction and preference datasets using Poro. To evaluate the results of our experiments, we translate IFEval, a widely-used instruction-following evaluation benchmark. We experimented with using different combinations of English and Finnish data in SFT and found that using all available data from both languages gave the best performance overall. Using only 10% of the Finnish instruction data (around 400 samples), however, still gives competitive performance. We contribute to Finnish LLM development by releasing our datasets, recipes, and model with open licenses at <https://huggingface.co/LumiOpen/Poro-34B-chat-OpenAssistant>.

In future we want to explore different ways of obtaining more Finnish data by, for instance, generating synthetic instruction and preference datasets. We will use these synthetic datasets to further investigate other alignment and finetuning methods. Additionally, we are interested on de-

veloping an evaluation benchmark for open-ended conversations in Finnish that takes cultural and linguistic nuances into account.

Acknowledgments

The authors wish to acknowledge CSC – IT Center for Science, Finland, for generous computational resources on the LUMI supercomputer. This project has received funding from the European Union’s Horizon Europe research and innovation programme under Grant agreement No 101070350. The contents of this publication are the sole responsibility of its authors and do not necessarily reflect the opinion of the European Union.

References

- Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, et al. 2023. Mega: Multilingual evaluation of generative ai. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4232–4267.
- Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.
- Pinzhen Chen, Shaoxiong Ji, Nikolay Bogoychev, Andrey Kutuzov, Barry Haddow, and Kenneth Heafield. 2024. Monolingual or multilingual instruction tuning: Which makes a better alpaca. In *The 18th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1–10. Association for Computational Linguistics.
- John Dang, Arash Ahmadian, Kelly Marchisio, Julia Kreutzer, Ahmet Üstün, and Sara Hooker. 2024. Rlhf can speak many languages: Unlocking multilingual preference optimization for llms. *CoRR*.
- Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac’h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailley Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024. A framework for few-shot language model evaluation.

- Rishav Hada, Varun Gumma, Adrian Wynter, Harshita Diddee, Mohamed Ahmed, Monojit Choudhury, Kalika Bali, and Sunayana Sitaram. 2024. Are large language model-based evaluators the solution to scaling up multilingual evaluation? In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1051–1070.
- Yun He, Di Jin, Chaoqi Wang, Chloe Bi, Karishma Mandyam, Hejia Zhang, Chen Zhu, Ning Li, Tengyu Xu, Hongjiang Lv, et al. 2024. Multi-if: Benchmarking llms on multi-turn and multilingual instructions following. *arXiv preprint arXiv:2410.15553*.
- 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*.
- Neel Jain, Ping-yeh Chiang, Yuxin Wen, John Kirchenbauer, Hong-Min Chu, Gowthami Somepalli, Brian R Bartoldson, Bhavya Kailkhura, Avi Schwarzschild, Aniruddha Saha, et al. 2024. Nef-tune: Noisy embeddings improve instruction fine-tuning. In *The Twelfth International Conference on Learning Representations*.
- 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*.
- Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, et al. 2024. Openassistant conversations-democratizing large language model alignment. *Advances in Neural Information Processing Systems*, 36.
- Viet Lai, Chien Nguyen, Nghia Ngo, Thut Nguyn, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 318–327.
- Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu Nouha Dziri Xinxu Lyu, Yuling Gu Saumya Malik Victoria Graf, Jena D Hwang, et al. 2024. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.
- Risto Luukkonen, Jonathan Burdge, Elaine Zosa, Aarne Talman, Ville Komulainen, Väinö Hatanpää, Peter Sarlin, and Sampo Pyysalo. 2024. Poro 34b and the blessing of multilinguality. *arXiv preprint arXiv:2404.01856*.
- Pedro Henrique Martins, Patrick Fernandes, João Alves, Nuno M Guerreiro, Ricardo Rei, Duarte M Alves, José Pombal, Amin Farajian, Manuel Faysse, Mateusz Klimaszewski, et al. 2024. Eurollm: Multilingual language models for europe. *arXiv preprint arXiv:2409.16235*.
- 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.
- 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.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.
- Uri Shaham, Jonathan Herzig, Roei Aharoni, Idan Szepktor, Reut Tsarfaty, and Matan Eyal. 2024. Multilingual instruction tuning with just a pinch of multilinguality. *arXiv preprint arXiv:2401.01854*.
- Zhengyan Shi, Adam X. Yang, Bin Wu, Laurence Aitchison, Emine Yilmaz, and Aldo Lipani. 2024. Instruction tuning with loss over instructions.
- Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhatipatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.
- Ahmet Üstün, Viraat Aryabumi, Zheng-Xin Yong, Wei-Yin Ko, Daniel D’souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, et al. 2024. Aya model: An instruction finetuned open-access multilingual language model. *arXiv preprint arXiv:2402.07827*.
- Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy J Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. 2024. Helpsteer2: Open-source dataset for training top-performing reward models. *arXiv preprint arXiv:2406.08673*.
- Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. 2020. Trl: Transformer reinforcement learning. <https://github.com/huggingface/trl>.
- 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.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. 2024. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

Author Index

- Al-Laith, Ali, 1, 470
Alumäe, Tanel, 136
Alves, Diego, 8
Andersson, Elsa, 17
Attieh, Joseph, 390
Avgustinova, Tania, 313
Ármannsson, Bjarki, 28, 37
- Barbu, Eduard, 383
Bassignana, Elisa, 223, 812
Benediktsdóttir, Ragnheiður María, 518
Berger, Maria, 48
Bergmanis, Toms, 287
Bergsson, Kormákur Logi, 518
Beyer, Yngvil, 98
Birkenes, Magnus Breder, 544
Bizzoni, Yuri, 142
Bjerring-Hansen, Jens, 1
Bjerva, Johannes, 480
Björklund, Johanna, 230
Boritchev, Maria, 252
Boyer, Matthieu Pierre, 55
Braaten, Rolv-Arild, 544
Breitung, Jens, 269
Brygfjeld, Svein Arne, 544
Burdge, Jonathan, 367
- Carlsson, Fredrik, 331
Casademont, Judit, 307
Charney, Joshua, 651
Charpentier, Lucas Georges Gabriel, 573
Chia, Noel, 66
Chiril, Patricia, 651
Conroy, Alexander, 1
Creutz, Mathias, 755
- de Boer, Maaike H. T., 508
de Vroe, Sander Bijl, 80
Debess, Iben Nyholm, 609, 622
Degn, Kirstine Nielsen, 1
Dehouck, Mathieu, 55
Dobnik, Simon, 697
Dorkin, Aleksei, 86
Dzielinski, Michal, 307
Dürlich, Luise, 331
- Edlund, Jens, 709
Einarsson, Hafsteinn, 181, 609, 622
- Enevoldsen, Kenneth, 561
Enstad, Tita, 98, 544
Ermus, Liis, 302
Etori, Naome A., 109
Ezquerro, Ana, 121
- Falkenjack, Johan, 17
Farsethås, Hans Christian, 544
Fedorchenko, Artem, 136
Feldkamp, Pascale, 142
Fischbach, Lea, 159
Fishel, Mark, 340, 458
Flek, Lucie, 159
Francis, Emilie Marie Carreau, 170
Friðriksdóttir, Kolbrún, 518
Friðriksdóttir, Steinunn Rut, 181
Färber, Michael, 499
- Getman, Yaroslav, 192
Ghatawala, Shashwat, 639
Gibert, Ona de, 201, 209
Ginter, Filip, 258, 424
Glišić, Isidora, 217
Gogoulou, Evangelia, 331
Goot, Rob van der, 223, 812
Grasmanis, Mikus, 359
Groh, Georg, 639
Grósz, Tamás, 192
Gudnason, Jon, 354, 518
Gulla, Jon Atle, 544
Gómez-Rodríguez, Carlos, 121
- Hafsteinsson, Hinrik, 28
Hakala, Kai, 80
Hansen, Dorte Haltrup, 470
Hansen, Lars Kai, 785
Hansson, Martin, 767
Hardmeier, Christian, 739
Harðarson, Þórir Hrafn, 241
Hatanpää, Väinö, 367
Hein, Indrek, 302
Heinecke, Johannes, 252
Henriksson, Aron, 767
Henriksson, Erik, 258
Herledan, Frédéric, 252
Hershovich, Daniel, 1
Hiovain-Asikainen, Katri, 192
Hu, Songbo, 755

Häglund, Emil, 230

Ingason, Anton Karl, 217

Ingimundarson, Finnur Ágúst, 37

Jacobson, Maria, 307

Jasonarson, Atli, 28, 680

Jensen, Anette, 223

Jäkel, Rene, 499

Jönsson, Arne, 17

Jørgensen, Tollef Emil, 269, 525

Kalnača, Andra, 279

Kanepajs, Arturs, 109

Kapočiūtė-Dzikiene, Jurgita, 287

Kardos, Márton, 142

Karisa, Randu, 109

Karlgren, Jussi, 80

Kaukonen, Elisabeth, 296

Kazos, Yannis, 776

Kiissel, Indrek, 302

Kildeberg, Mikkel Wildner, 223

Kleen, Caroline, 159

Klints, Agute, 359

Komulainen, Ville, 367, 826

Koponen, Maarit, 661

Korhonen, Anna, 755

Kristensen-McLachlan, Ross Deans, 480

Kroka, Marie Ingeborg, 729

Kukk, Kättriin, 307

Kunilovskaya, Maria, 313

Kunz, Jenny, 323, 433

Kuparinen, Olli, 634

Kurfalı, Murathan, 331

Kurimo, Mikko, 192

Kutuzov, Andrey, 544, 573

Kuulmets, Hele-Andra, 340

Kvale, Knut, 440, 448

Lameli, Alfred, 159

Langø, Victoria Ovedie Chruickshank, 397

Larsen, Nicolaj, 223

Lehtonen, Tommi, 192

Lent, Heather, 480

Levāne-Petrova, Kristīne, 279

Lhoneux, Miryam de, 492

Liu, Peng, 544

Loftsson, Hrafn, 241

Lokmane, Ilze, 359

Lu, Kevin, 109

Luukkonen, Risto, 367

Lág, Dávid í, 354

Magnifico, Giacomo, 383

Maher, Erik Anders, 518

Markantonatou, Stella, 776

Mihkla, Meelis, 302

Mikhailov, Vladislav, 397, 544, 573, 729

Muñoz Sánchez, Ricardo, 697

Myhre, Aslak Sira, 544

Myneni, Hemanadhan, 688

Männistö, Johanna, 390

Mæhlum, Petter, 397, 537, 544

Müller-Eberstein, Max, 812

Nešpore-Bērzkalne, Gunta, 359

Nielbo, Kristoffer, 142

Nielsen, Martin Carsten, 785

Nieminen, Tommi, 201, 408

Nikolaev, Alexandre, 661

Nimb, Sanni, 470

Nivre, Joakim, 331, 419

Nuutinen, Emil, 424

O'Brien, Dayyán, 209

Oepen, Stephan, 544, 573

Oji, Romina, 433

Olsen, Sussi, 470

Orlowski, Eric J. W., 307

Østgulen, Wilfred, 544

Øvrelid, Lilja, 397, 537, 544, 573, 729, 801

Paikens, Pēteris, 359

Pakalne, Tatjana, 279

Paperno, Denis, 508

Parsons, Phoebe, 440, 448

Pashchenko, Dmytro, 458

Pedersen, Bolette S., 470

Petrelli, Danila, 307

Piits, Liisi, 302

Pinnis, Mārcis, 287

Ploeger, Esther, 480

Poelman, Wessel, 492

Politov, Andrei, 499

Ponzetto, Simone Paolo, 66

Pretkalniņa, Lauma, 359

Purason, Taido, 340

Pyysalo, Sampo, 367, 826

Rastas, Iiro, 424

Reguera-Gómez, Cristina, 508

Rehbein, Ines, 66

Richter, Caitlin Laura, 217, 518
Riedel, Morris, 688
Riess, Mike, 525
Rituma, Laura, 359
Roald, Marie, 98
Roman, Maria-Alexandra, 639
Rosa, Javier de la, 544
Rouhe, Aku, 80
Rønningstad, Egil, 537
Røsok, Marie Iversdatter, 98

Saatrup Nielsen, Dan, 181, 561
Sabir, Ahmed, 296
Sahkai, Heete, 302
Salvi, Giampiero, 440, 448
Samuel, David, 544, 573
Sarlin, Peter, 367
Saucedo, Paola, 480
Scalvini, Barbara, 354, 609, 622
Scherrer, Yves, 201, 634
Schledermann, Emil Allerslev, 223
Schneider-Kamp, Peter, 561
Schuster, Carolin M., 639
Sharma, Rajesh, 296
Shastry, Rishabh, 651
Shkalikov, Oleh, 499
Sigtryggsson, Jóhannes B., 28
Sigurðsson, Einar Freyr, 28, 37, 680
Simonsen, Annika, 609, 622
Sirts, Kairit, 86
Solberg, Per Erik, 448
Souza, Karen de, 661
Stamou, Vivian, 776
Stampoulidis, George, 80
Steingrímsson, Steinþór, 28, 680
Stenger, Irina, 313
Stenlund, Mathias, 688
Storset, Lilja Charlotte, 537
Ståde, Madara, 359
Svendsen, Torbjørn, 440, 448
Szawerna, Maria Irena, 697
Sørensen, Nathalie, 470

Talman, Aarne, 367
Tarkka, Otto, 258
Taurina, Evelina, 359
Terenziani, Sofia Elena, 714
Tiedemann, Jörg, 201, 209, 390, 408, 755
Touileb, Samia, 729, 801
Trosterud, Trond, 98
Tånnander, Christina, 709

Uminsky, David, 651

Vaaben Bornerup, Jesper, 739
Vahtola, Teemu, 755
Vakili, Thomas, 767
Vakirtzian, Socrates, 776
van Heeswijk, Mark, 80
Variš, Dušan, 209
Vejlgaard Holm, Søren, 785
Vellidal, Erik, 397, 537, 544, 573, 729, 801
Vilares, David, 121
Virpioja, Sami, 408
Volodina, Elena, 697
Vulić, Ivan, 755

Wetjen, Freddy, 544

Xue, Wei, 313

Yankovskaya, Lisa, 458
You, Huiling, 801

Zahra, Shorouq, 331
Zaitova, Iuliia, 313
Zhang, Lemei, 544
Zhang, Mike, 223, 812
Zosa, Elaine, 367, 826

Ólafsson, Stefán, 241