ALTA 2023

Proceedings of the 21st Annual Workshop of the Australasian Language Technology Association

November 29 - December 1, 2023
University of Melbourne
Melbourne, Australia

The ALTA organizers gratefully acknowledge the support from the following sponsors.

Platinum

MELBOURNE
CONNECT

Gold

POSTERA CRESCAM LAUDE
THE UNIVERSITY OF
MELBOURNE

Google

Australian Government
Defence

Silver

seek
seek.com.au

Bronze

REDENLAB

CommonwealthBank

In cooperation with

THE UNIVERSITY OF MELBOURNE

MELBOURNE CONNECT

RMIT UNIVERSITY

MELBOURNE DATA ANALYTICS PLATFORM

MACQUARIE University
SYDNEY · AUSTRALIA

Introduction

Welcome to ALTA2023, the 21st Annual Workshop of the Australasian Language Technology Association. We're thrilled to have you join us in Melbourne, Australia, from November 29 to December 1, 2023. After several years of hybrid format due to COVID, this year ALTA will largely operate as an offline event with limited online capability, so as to encourage the community to come together in person.

ALTA is the premier workshop for natural language processing or computational linguistics in the Australasia region (despite being downgraded to C by CORE2023) and it is now indexed by Scopus (after efforts by the ALTA executive committee). This year the programme includes 3 keynote talks (with 1 more joint keynote with AI@Melbourne Connect Symposium), 1 tutorial, 1 panel discussion, 5 oral sessions and 1 poster/demo session. In addition to long/short academic papers, shared task papers and non-archival abstract presentations, we've introduced a new category of submission — industry demonstrations — to foster greater industry involvement. In terms of submission statistics, we received 25 long/short papers, 6 shared task papers, 6 abstract presentations and 2 industry demonstrations, and accepted 17 long/short papers (breakdown: 10 long and 7 short) and all submitted shared task papers, abstract presentations and industry demonstrations. Our acceptance rate for long/short papers (68%) aligns with last year's figures. Note that all long/short papers went through the double-blind peer-review process (shared task papers, abstract presentations and industry demonstrations, however, did not).

2023 has been an interesting year. We saw large language models breaking into mainstream consciousness, and within a year (ChatGPT was released on 30 November 2022) they have transformed the field in both academia and industry. This is reflected in our keynote talks, panel discussion and industry demonstrations, which all feature large language models to some extent. One of the accepted papers even credits a large language model as a co-author.

This workshop could not have happened without the help and enthusiastic participation of many parties, and we would like to give a big 'thank you' to all of them. Specifically, we want to thank our keynote speakers — Reza Haffari (Monash), Heng Ji (Illinois Urbana-Champaign) and Terrence Szymanski (SEEK) — for their inspiring talks. Special thanks to the organising and program committee whose hard work made ALTA a reality. Lastly, we want to express our appreciation to our sponsors: Melbourne Connect and Telstra (Platinum); The University of Melbourne AI Group, Google and Defense Science and Technology Group (Gold); SEEK (Silver); and Redenlab and Commonwealth Bank (Bronze). 2023 financially hasn't been the best year, and we're incredibly thankful for the support you've provided.

Welcome to Melbourne! Our submissions have come from many places, and we look forward to a rich and rewarding time together.

Jey Han Lau

Program Chair

Organizing Committee

General Chair

Eduard Hovy, The University of Melbourne and Carnegie Mellon University

Program Chair

Jey Han Lau, The University of Melbourne

Local Chairs

Lea Frermann, The University of Melbourne
Ekaterina Vylomova, The University of Melbourne

Sponsorship Chair

Karin Verspoor, Royal Melbourne Institute of Technology
Christine de Kock, The University of Melbourne

Publicity Chair

Ekaterina Vylomova, The University of Melbourne

Liaison Chair

Mel Mistica, The University of Melbourne

Publication Chair

Qiongkai Xu, The University of Melbourne and Macquarie University

Technology Chair

Rui Xing, The University of Melbourne

Program Committee

Area Chairs

Antonio Jimeno, The University of Melbourne
Afshin Rahimi, Amazon
Lan Du, Monash University

Reviewers

Dilek Hakkani-Tur, Amazon
Arpit Sood, Facebook Research
Daniel Beck, The University of Melbourne
Diego Molla, Macquarie University
Dominique Estival, Western Sydney University
Ekaterina Vylomova, The University of Melbourne
Fajri Koto, Mohamed bin Zayed University of Artificial Intelligence
Gabriela Ferraro, Australian National University
Guanqun Yang, Stevens Institute of Technology
Hamed Hassanzadeh, CSIRO
Hiyori Yoshikawa, Tokyo Institute of Technology
Jeffrey Chan, Royal Melbourne Institute of Technology
Jonathan Kummerfeld, University of Sydney
Karin Verspoor, Royal Melbourne Institute of Technology
Kemal Kurniawan, The University of Melbourne
Lea Frermann, The University of Melbourne
Mac Kim, Macquarie University
Maria Myung-Hee, University of New South Wales
Massimo Piccardi, University of Technology Sydney
Meladel Mistica, The University of Melbourne
Mike Conway, The University of Melbourne
Nitin Indurkhya, The University of New South Wales
Reza Haf, Monash University
Rolf Schwitter, Macquarie University
Sarvnaz Karimi, CSIRO
Timothy Baldwin, Mohamed bin Zayed University of Artificial Intelligence
Xiang Dai, CSIRO
Xiuzhen Zhang, Royal Melbourne Institute of Technology
Xudong Han, The University of Melbourne
Zhuohan Xie, The University of Melbourne

Keynote Talk: Towards Effective NLP Systems: Cultural Norms, Explainability, and Reasoning Enhancement

Reza Haffari
Monash University
2023-11-30 09:00:00 – Room: **The Forum, Superfloor, Melbourne Connect**

Abstract: Effective real-world NLP systems need to be trustworthy and take users' contexts into account. As such, they need to recognise and adhere to cultural norms, be able to explain their outcomes, and be capable of complex reasoning with certified knowledge. In this talk, we describe our research towards these goals. First, we discuss the challenges of adhering to socio-cultural norms in cross-cultural and multilingual communications. We present our work on assistive dialogue systems that can identify and address norm violations. Second, we highlight the importance of explaining the outcomes of NLP models while considering the user context. We present methods for generating fast, high-quality, and context-aware explanations. Finally, we look into the limitations of large language models (LLMs) in knowledge hallucination and complex reasoning. We propose a systematic approach to assess LLMs' knowledge using knowledge graphs and enhance their complex reasoning capabilities for improved accuracy and trustworthiness.

Bio: Gholamreza (Reza) Haffari is a Professor in the Department of Data Science and Artificial Intelligence (DSAI), Monash University, Australia. He is an Australian Research Council (ARC) Future Fellow and the Director of the Vision and Language Group. Reza's research has been supported by awards and grants from government and industry, including ARC, Google Research, Amazon Research, eBay Research, and Adobe Research. His research is in the intersection of Natural Language Processing, Deep Learning, and Machine Learning.

Keynote Talk: SmartBook: An AI Prophetess for Disaster Reporting and Forecasting

Heng Ji
University of Illinois Urbana-Champaign
2023-11-30 12:00:00 – Room: **The Forum, Superfloor, Melbourne Connect**

Abstract: History repeats itself, sometimes in a bad way. Preventing natural or man-made disasters requires being aware of these patterns and taking pre-emptive action to address them. Effective response to emerging events like the COVID pandemic and the Ukraine Crisis require a time-sensitive comprehensive understanding of the situation. Automated generation of situation reports can significantly reduce the time, effort, and cost for domain experts when preparing their official human-curated reports. However, AI research toward this goal has been very limited, and no successful trials have yet been conducted to automate such report generation and "what-if" disaster forecasting. In this talk I present SmartBook, a novel framework that cannot be solved by ChatGPT, to consume large volumes of news data and produce a structured situation report with multiple hypotheses (claims) summarized and grounded with rich links to factual evidence through claim detection, fact checking, misinformation detection and factual error correction. SmartBook can also serve as a novel news event simulator or an intelligent prophetess when given "What-if" conditions and dimensions elicited from a domain expert user concerning a disaster scenario.

Bio: Heng Ji is a professor at Computer Science Department of University of Illinois Urbana-Champaign. She is an Amazon Scholar. She is the Founding Director of Amazon-Illinois Center on AI for Interactive Conversational Experiences (AICE). She received her B.A. and M.A. in Computational Linguistics from Tsinghua University and her M.S. and Ph.D. in Computer Science from New York University. Her research interests focus on Natural Language Processing, especially on Multimedia Multilingual Information Extraction and Knowledge-enhanced Large Language Models. Some awards include "Young Scientist" by World Laureates Association in 2023; "Young Scientist" and a member of the Global Future Council on the Future of Computing by the World Economic Forum in 2016 and 2017; "Women Leaders of Conversational AI" (Class of 2023) by Project Voice; "AI's 10 to Watch" Award by IEEE Intelligent Systems in 2013, NSF CAREER award in 2009; Best Demo Paper Awards at ACL2020 and NAACL2021.

Keynote Talk: The commoditisation of NLP in industry

Terrence Szymanski
SEEK
2023-12-01 09:00:00 – Room: **The Forum, Superfloor, Melbourne Connect**

Abstract: Is NLP becoming a commodity? In 2023, thanks to a thriving ecosystem of open-source libraries and commercial services, building applied NLP solutions is easier than it ever has been before. The barrier to entry is low, and NLP technology which previously would have required a team of experts is now accessible to a vast audience of data scientists, developers, and enthusiasts. Even the most cutting-edge developments in large language models are readily accessible, and consumers can pick and choose between different models and nearly interchangeable service providers. In my talk, I will share some of my experiences building applied NLP services at SEEK, as well as some of my observations on how the "commoditisation of NLP" has (and has not) impacted the way that these services are currently developed and deployed. I will also offer my perspectives on some broad questions raised by the commoditisation of NLP: What new opportunities are unlocked? What new risks and challenges do we face? And what does this mean for the NLP community and the future of NLP?

Bio: Terrence Szymanski is a Principal Data Scientist at SEEK, where he leads a team of data scientists responsible for the Selection and Standout domains. His team build and deploy ML models to match jobs and candidates; NLP models to extract and normalise information from unstructured documents; deep learning models to learn fine-tuned representations of text; and multiple other related AI services. Terrence obtained his PhD in Linguistics from the University of Michigan, and he performed postdoctoral research at University College Dublin. His research has spanned diverse areas such as morphological inference, computational historical linguistics, and text analytics. He has been practicing data science in industry since 2016, working at ANZ bank before joining SEEK. He is a founder and organiser of the NLP Reading Group at SEEK, a winner of hackathons inside and outside of SEEK, and a regular attendee (and occasional organiser) of data-science-related Meetups around Melbourne.

Table of Contents

BanglaClickBERT: Bangla Clickbait Detection from News Headlines using Domain Adaptive BanglaBERT and MLP Techniques
Saman Sarker Joy, Tanusree Das Aishi, Naima Tahsin Nodi and Annajiat Alim Rasel 1

Story Co-telling Dialogue Generation based on Multi-Agent Reinforcement Learning and Story Highlights
Yu-Kai Lee and Chia-Hui Chang 11

Using C-LARA to evaluate GPT-4's multilingual processing
ChatGPT C-LARA-Instance, Belinda Chiera, Cathy Chua, Chadi Raheb, Manny Rayner, Annika Simonsen, Zhengkang Xiang and Rina Zviel-Girshin 20

Exploring Causal Directions through Word Occurrences: Semi-supervised Bayesian Classification Framework
King Tao Jason Ng and Diego Molla 30

The sub-band cepstrum as a tool for local spectral analysis in forensic voice comparison
Shunichi Ishihara and Frantz Clermont 40

Right the docs: Characterising voice dataset documentation practices used in machine learning
Kathy Reid and Elizabeth T. Williams 51

MCASP: Multi-Modal Cross Attention Network for Stock Market Prediction
Kamaladdin Fataliyev and Wei Liu 67

Catching Misdiagnosed Limb Fractures in the Emergency Department Using Cross-institution Transfer Learning
Filip Rusak, Bevan Koopman, Nathan J. Brown, Kevin Chu, Jinghui Liu and Anthony Nguyen 78

Turning Flowchart into Dialog: Augmenting Flowchart-grounded Troubleshooting Dialogs via Synthetic Data Generation
Haolan Zhan, Sameen Maruf, Lizhen Qu, Yufei Wang, Ingrid Zukerman and Gholamreza Haffari 88

Encoding Prefixation in Southern Min
Yishan Huang 100

An Ensemble Method Based on the Combination of Transformers with Convolutional Neural Networks to Detect Artificially Generated Text
Vijini Liyanage and Davide Buscaldi 107

Chat Disentanglement: Data for New Domains and Methods for More Accurate Annotation
Sai R. Gouravajhala, Andrew M. Vernier, Yiming Shi, Zihan Li, Mark S. Ackerman and Jonathan K. Kummerfeld 112

Enhancing Bacterial Infection Prediction in Critically Ill Patients by Integrating Clinical Text
Jinghui Liu and Anthony Nguyen 118

Predicting Empathic Accuracy from User-Designer Interviews
Steven Nguyen Fataliyev, Daniel Beck and Katja Holtta-Otto 125

CRF-based recognition of invasive fungal infection concepts in CHIFIR clinical reports
Yang Meng, Vlada Rozova and Karin Verspoor 130

The uncivil empathy: Investigating the relation between empathy and toxicity in online mental health support forums
Ming-Bin Chen, Jey Han Lau and Lea Frermann 136

Shared Task (Not Peer Reviewed)

Overview of the 2023 ALTA Shared Task: Discriminate between Human-Written and Machine-Generated Text
Diego Molla, Haolan Zhan, Xuanli He and Qiongkai Xu 148

A Prompt in the Right Direction: Prompt Based Classification of Machine-Generated Text Detection
Rinaldo Gagiano and Lin Tian 153

Automatic Detection of Machine-Generated Text Using Pre-Trained Language Models
Yunhao Fang 159

An Ensemble Based Approach To Detecting LLM-Generated Texts
Ahmed El-Sayed and Omar Nasr 164

Feature-Level Ensemble Learning for Robust Synthetic Text Detection with DeBERTaV3 and XLM-RoBERTa
Saman Sarker Joy and Tanusree Das Aishi 169

Stacking the Odds: Transformer-Based Ensemble for AI-Generated Text Detection
Duke Nguyen, Khaing Myat Noe Naing and Aditya Joshi 173

Tutorial (Not Peer Reviewed)

Natural Language Processing for Clinical Text
Vlada Rozova, Jinghui Liu and Mike Conway 179

Program

Wednesday, November 29, 2023

12:00 - 13:00	*Lunch*
13:00 - 14:00	*Tutorial Part 1*
14:00 - 14:15	*Afternoon Tea*
14:15 - 15:15	*Tutorial Part 2*
15:15 - 15:30	*Break*
15:30 - 16:30	*Tutorial Part 3*
16:30 - 17:15	*Tutorial Conclude and NLP Meetup Refreshments*
17:15 - 19:30	*NLP Meetup @ Melbourne Connect*

Thursday, November 30, 2023

08:45 - 09:00	*Opening*
09:00 - 10:00	*ALTA Keynote 1 - Reza Haffari, Towards Effective NLP Systems: Cultural Norms, Explainability, and Reasoning Enhancement*
10:00 - 10:30	*Morning Tea*
10:30 - 12:00	*Oral Presentations (Applications) [Chair: Massimo Piccardi]*
12:00 - 13:00	*ALTA Keynote 2 - Heng Ji, SmartBook: An AI Prophetess for Disaster Reporting and Forecasting*
13:00 - 14:00	*Lunch*
14:00 - 15:00	*Oral Presentations (Speech / Linguistics) [Chair: Daniel Beck]*
15:00 - 15:15	*Afternoon Tea*
15:15 - 16:15	*Panel Discussion [Panellists: Long Duong, Reza Haffari, Ed Hovy, Karin Verspoor]*
16:15 - 17:15	*Abstract Presentations / Industry Demonstrations*
18:30 - 22:00	*Dinner @ Stomping Ground*

Friday, December 1, 2023

09:00 - 10:00	*ALTA Keynote 3 - Terrence Szymanski*
10:00 - 10:30	*Morning Tea*
10:30 - 12:00	*Oral Presentations (Medical / ML) [Chair: Antonio Jimeno Yepes]*
12:00 - 13:00	*ALTA Keynote 4 (Joint Session with AI@Melbourne Connect Symposium) - Heng Ji*
13:00 - 14:00	*Lunch*
14:00 - 15:30	*Oral Presentations (Dialogue) [Chair: Jonathan Kummerfeld]*
15:30 - 15:45	*Afternoon Tea*
15:45 - 16:45	*Oral Presentations (Shared Task)*
16:45 - 17:00	*ALTA AGM*
17:00 - 17:30	*Best Paper Award / Shared Task Award / Closing*

BanglaClickBERT: Bangla Clickbait Detection from News Headlines using Domain Adaptive BanglaBERT and MLP Techniques

Saman Sarker Joy, Tanusree Das Aishi, Naima Tahsin Nodi, Annajiat Alim Rasel
Department of Computer Science and Engineering
BRAC University, 66 Mohakhali, Dhaka-1212, Bangladesh
`{saman.sarker.joy, tanusree.das.aishi, naima.tahsin.nodi}@g.bracu.ac.bd`
`annajiat@bracu.ac.bd`

Abstract

News headlines or titles that deliberately persuade readers to view a particular online content are referred to as clickbait. There have been numerous studies focused on clickbait detection in English language, compared to that, there have been very few researches carried out that address clickbait detection in Bangla news headlines. In this study, we have experimented with several distinctive transformers models, namely BanglaBERT and XLM-RoBERTa. Additionally, we introduced a domain-adaptive pretrained model, BanglaClickBERT. We conducted a series of experiments to identify the most effective model. The dataset we used for this study contained 15,056 labeled and 65,406 unlabeled news headlines; in addition to that, we have collected more unlabeled Bangla news headlines by scraping clickbait-dense websites making a total of 1 million unlabeled news headlines in order to make our BanglaClickBERT. Our approach has successfully surpassed the performance of existing state-of-the-art technologies providing a more accurate and efficient solution for detecting clickbait in Bangla news headlines, with potential implications for improving online content quality and user experience.

1 Introduction

The Internet has led to a surge in the use of online news media, which provides users with easy access to information at any time. However, news websites use clickbait headlines that can be misleading and frustrating to users. These headlines are designed to attract users and create suspense, containing exaggerated information that does not match the content. Clickbait headlines aim to lure users into clicking on them but ultimately cause frustration. Pengnate et al. concluded a research and found that clickbait headlines can lead to higher click-through rates, but may lead to negative user experiences such as frustration and disappointment. Examples of clickbait headlines in Bangla are in Figure 1. The core differences between clickbait and non-clickbait is described in Appendix A.

খালের মাঝে টানা জাল দিতেই উঠলো বিশাল বড় চিতই মাছ, জেলেদের মাছ ধরার ভিডিও ভাইরাল

As the nets drag in the canal, big tuna catfish, the fishing video went viral! **- Type: Formatting**

আমার স্ত্রী প্রাইমারী স্কুলের টিচার একদিন রাতে ডিনারের শেষে ...

My wife is a primary school teacher, one night at dinner... **- Type: Curiosity Gap/Teasing**

রিমান্ডে যাদের বিষয়ে গুরুত্বপূর্ণ তথ্য দিলেই পরীমনি

Porimoni gave important information about those on remand **- Type: Ambiguous**

Figure 1: Examples of Bangla clickbait news headlines with its corresponding English translation and type of clickbait

The use of online news media has increased rapidly in Bangladesh, with an estimated 66.3 million internet users[1] and 14 million online readers of Prothom Alo (Correspondent, 2022), one of the top newspapers in the country. However, the increasing number of clickbait titles on news websites has become a significant issue, leading to frustration and disappointment among users. While research has been conducted on clickbait detection in English, very little has been done in Bangla, a language spoken by millions of people in Bangladesh and other countries. In English, for The Clickbait Challenge 2017, Webis Clickbait Corpus 2017 (Potthast et al., 2018b) was created which had a total of 38,517 sentences from major US news publishers. In Bangla, Mahtab et al. have constructed a Bangla clickbait detection dataset containing 15,056 labeled news articles and 65,406 unlabelled news articles. In this paper, we present BanglaClickBERT, a pretrained model for clickbait detection in

---

[1] `https://www.cia.gov/the-world-factbook/countries/bangladesh/`

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 1–10
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

Bangla news websites. We use the labeled dataset for training and validating our model and scrape clickbait-dense websites to gather more unlabelled news article headlines, increasing the number of unlabelled news headlines to around 1 million. We use this to pretrain the BanglaBERT (Bhattacharjee et al., 2022) model, which we then pretrain to create BanglaClickBERT.

The main contributions of this paper can be summarized as follows:

- We scrape clickbait-dense websites and create an unlabelled news headlines dataset of around 1 million which we use to pretrain BanglaBERT model converting it to BanglaClickBERT.
- We experiment with different machine learning models, deep neural network models, and transformers models like BanglaBERT, XLM-RoBERTa, and our BanglaClickBERT to develop a Bangla Clickbait Detection model for Bangla news headline data. We compare the performance of our model using different metrics.

2 Literature Review

The roots of clickbait can be found in tabloids, a form of journalism that has existed since the 1980s (Bird, 2008). The three primary sources from which clickbait identification attributes may be generally retrieved are (1) the related article that the post text wants the user to visit, (2) metadata for both, and (3) the connected article (Munna and Hossen, 2021). Potthast et al. and Biyani et al. additionally took into account metadata, related content, and handcrafted elements in addition to the post-text analysis. They used methods like Gradient Boosted Decision Trees (GBDT) and assessed the TF-IDF similarity between the headline and article content. Potthast et al. in another paper also mentioned the Clickbait Challenge 2017, which invited the affirmation of 13 detectors were presented as the clickbait detectors for screening, realizing considerable enhancements in detecting performance above the prior state of the art. Zhou first used a self-attentive RNN to choose the crucial terms in the title before building a BiGRU network to encode the contextual information for the 2017 Clickbait Challenge. On the contrary, Thomas used an LSTM model for the clickbait challenge that included article content. To create the word embedding of clickbait titles, Rony et al. applied the continuous skip-gram model. Nevertheless, Indurthi et al. were the first to study the use of transformer regression models in clickbait identification and won the clickbait challenge. Additionally, Hossain et al. produced the first dataset of Bengali newspapers for Bengali false news detection of around 50K Bangla news articles in an annotated dataset. Besides Bangla, we have explored about clickbait detection techniques in news and social media in other languages. Genç and Surer used Logistic Regression (85% accuracy), Random Forest (86% accuracy), LSTM (93% accuracy), ANN (93% accuracy), Ensemble Classifier (93% accuracy), and BiLSTM (97% accuracy) on 48,060 headlines from news sources pulled from Twitter for Turkish clickbait detection. Moreover, Razaque et al. used Long short-term memory, Word2vec and compared their models with Naive Bayes classifier for clickbait detection on social media. Bronakowski et al. achieved 98% accuracy in recognizing clickbait headlines by using thirty distinct types of semantic analysis and six different machine-learning approaches, both individually and in groups. The suggested models can be used as a model for creating useful programs that swiftly identify clickbait headlines. Farhan et al. used Gated Recurrent Unit (GRU) and Convolutional Neural Network (CNN)-based ensemble model for sarcasm detection for Bangla language achieving 96% F1-score and accuracy. It gave us an insight on what type of work can be done using NLP and for gathering knowledge and examples related to our work. Additionally, Beltagy et al. created SciBERT which is a pretrained language model, based on BERT used unsupervised pretraining on scientific articles, providing us knowledge about domain-adaptive BERT which can help enhance efficiency on a range of scientific NLP tasks and produce cutting-edge results. Moreover, Jahan et al. created BanglaHateBERT, which is a retrained version of the pre-existing BanglaBERT model, and trained it having a widespread corpus of hostile, insulting, and offensive Bengali language, and outperformed the generic pretrained language model in various datasets. So, to sum up with, we have analysed about the origins of clickbait, checked different datasets on different languages, learned about differnet NLP methods, and observed the potentials of specialized transformers models like SciBERT and BanglaHateBERT.

3 Problem Statement

We approach the task of clickbait detection as a decision-making challenge; a binary classification task problem with two main categories $C = \{clickbait, non - clickbait\}$. Given a set of Bangla news headlines $T = \{t_1, t_2, t_3, \ldots, t_N\}$, our objective is to predict labels $Y = \{y_1, y_2, y_3, \ldots, y_n\}$ for these headlines. Here, y_i assumes the value 1 if title t_i is classified as clickbait and 0 if it is classified as non-clickbait. The problem can be formulated as,

$$< C, Y >= \{non - clickbait : 0, clickbait : 1\}$$

4 Dataset Description

The dataset (Mahtab et al., 2023) we used consists of two sets: an annotated set and an unannotated set of clickbait news information. The information with our augmentation is shown in Table 1.

4.1 Annotated Dataset

The annotated dataset comprises 15,056 articles, each labeled with one of two categories: Clickbait as 1 and Non-clickbait as 0. The articles in this subset cover a diverse range of topics. For our task, we focus only on the columns "Headlines" and "Labels" as they are essential. This dataset will be used for the classification task.

4.2 Unannotated Dataset

The unannotated dataset consists of 65,406 Bangla articles with clickbait titles. These articles were gathered from clickbait-dense websites. However, since 65k unlabelled samples may not be sufficient for our task, we expanded the dataset by scraping more clickbait-dense websites using *Selenium*[2] library. This effort resulted in a total of 1,078k or 1 Million unlabelled clickbait headlines. This unannotated dataset will be used for the pretraining.

Information	Value
Crawling Period	Feb 2019 - June 2023
Total Clickbait	5,239
Total Non-clickbait	9,817
Total Unlabelled Before	65,406
Total Unlabelled After	1,078,234

Table 1: Information of both the annotated and unannotated datasets

5 Methodology

We have used some Statistical Models and Deep Learning Models and then we have implemented Transformers models Like BanglaBERT, XLM-RoBERTa and Domain Adaptive BanglaClickBERT with several variations. Based on these variation, we try to come up with the best model.

5.1 Statistical Models

For statistical models, we will employ Logistic and Random Forest classifiers on a combination of various features like TF-IDF (term frequency–inverse document frequency) of the word and character n-grams, Bangla pretrained word embeddings, punctuation frequency, and normalized Parts-of-Speech frequency.

5.2 Deep Learning Models

When it comes to deep learning models, there are several powerful techniques that can be employed e.g. Long Short-Term Memory (LSTM), Bidirectional LSTM (BiLSTM) and ensemble methods. These models have shown great success in various natural language processing tasks, including sentiment analysis and text classification.

5.3 Transformer Models

5.3.1 BanglaBERT

BanglaBERT (Bhattacharjee et al., 2022) is a BERT-based Natural Language Understanding (NLU) model pretrained specifically on Bangla using a massive 27.5GB pretraining corpus. BanglaBERT has demonstrated remarkable performance in achieving state-of-the-art results across diverse NLP tasks.

5.3.2 XLM-RoBERTa

XLM-RoBERTa (Conneau et al., 2020), a large-scale multilingual language model based on Facebook's RoBERTa (Liu et al., 2019). XLM-RoBERTa undergoes pretraining on an extensive 2.5TB dataset of filtered CommonCrawl data.

5.3.3 Domain Adaptive Pretraining

We also propose to further pretrain BanglaBERT using a large number of headlines extracted from clickbait-filled websites. Gururangan et al. finds that tailoring pretrained language models to specific domains through adaptive pretraining techniques leads to significant improvements in task performance.

---

[2] `https://www.selenium.dev/`

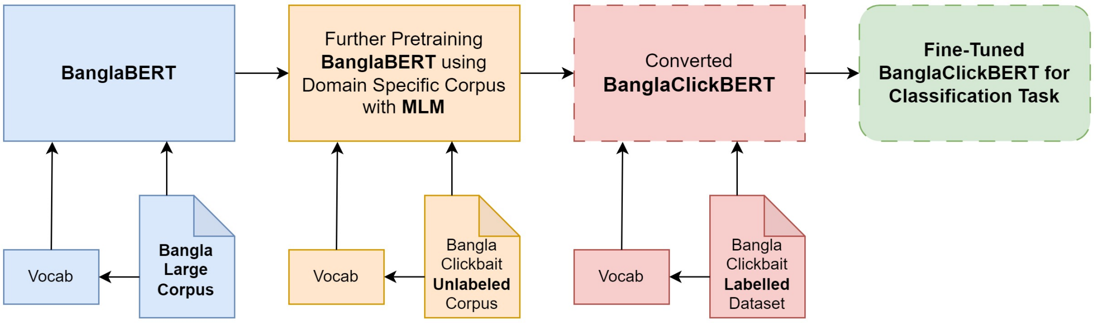

BanglaBERT
Further Pretraining BanglaBERT using Domain Specific Corpus with MLM
Converted BanglaClickBERT
Fine-Tuned BanglaClickBERT for Classification Task
Vocab
Bangla Large Corpus
Vocab
Bangla Clickbait Unlabeled Corpus
Vocab
Bangla Clickbait Labelled Dataset

Figure 2: Workflow of *BanglaClickBERT* Creation

6 Creation of BanglaClickBERT

Language models like BERT have revolutionized the field of NLP by introducing context-aware learning and significantly improving performance across various NLU tasks. However, applying these models to low-resource languages such as Bangla requires specialized adaptation to achieve optimal results. To address this challenge, we propose the development of BanglaClickBERT by further pretraining BanglaBERT with a vast dataset of clickbait news headlines. A workflow of this is shown in Figure 2.

6.1 Reason for Pretraining

Gururangan et al. investigated whether it is still helpful to tailor a pretrained model to the domain of a target task. From their research, it was found that a second phase of pretraining in-domain (domain-adaptive pretraining) leads to performance gains, in both high and low-resource settings. Also, in the BanglaHateBERT paper (Jahan et al., 2022), we found performance gains after pretraining.

6.2 Pretraining Data

We collected a diverse set of clickbait news headlines mentioned in Section 4, comprising 1 million samples from various online sources. These headlines were chosen to cover a wide range of clickbait headlines, ensuring the model's adaptability to different contexts like news on lifestyle, entertainment, business, viral videos etc.

6.3 Training Strategy

The retraining process was carried out using the Masked Language Model (MLM) approach. During training, we masked 15% of the tokens in each sequence, forcing the model to predict these masked tokens and thus gain contextual understanding. Additionally, we set the model to accept up to 128 sentence tokens to capture more extensive contextual dependencies. BanglaClickBERT was pretrained for 10 epochs, on an *NVIDIA GeForce RTX 3070*. It took us almost 28 hours to pretrain for 10 epochs. We adopted the Adamw (Loshchilov and Hutter, 2019) optimization solver, known for its computational efficiency and memory-friendly characteristics, with a learning rate of 5e-5. The maximum sequence length was set to 32 as there was no sentence bigger than 30 shown in Figure 3. The pretrained models are uploaded on Hugging face website.[3]. The unannotated dataset of clickbaits will also be provided on request.[4].

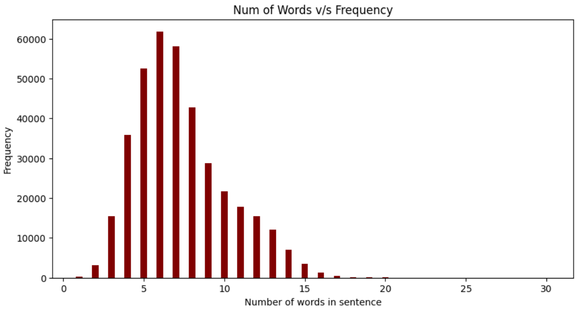


Figure 3: Frequnecy of all the sentences in the unannotated corpus. It shows that all the sentence lengths are less than 30.

7 System Overview

7.1 Statistical Models

We used two Statistical models: Logistic Regression and Random Forest. Logistic Regression and Random Forest both are widely used classification algorithms that are particularly well-suited for binary classification tasks. We used TF-IDF vectors.

---

[3] `https://huggingface.co/samanjoy2/banglaclickbert_base`

[4] `https://tinyurl.com/BanglaClickBERTdata`

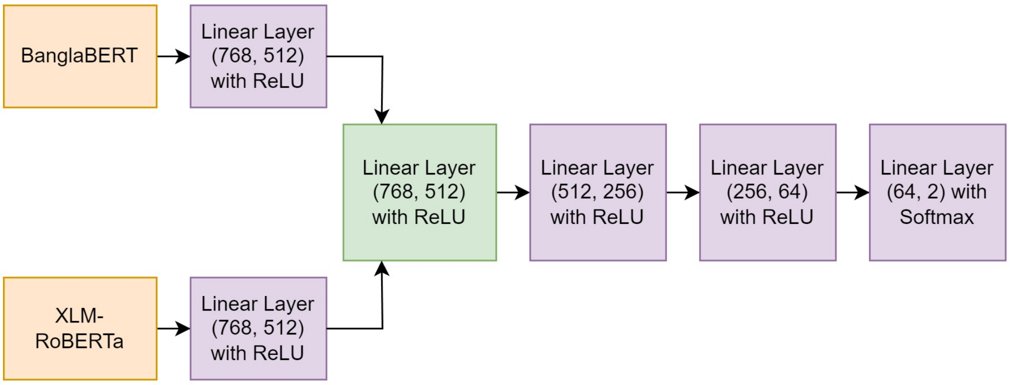

BanglaBERT
Linear Layer (768, 512) with ReLU
XLM-RoBERTa
Linear Layer (768, 512) with ReLU
Linear Layer (768, 512) with ReLU
Linear Layer (512, 256) with ReLU
Linear Layer (256, 64) with ReLU
Linear Layer (64, 2) with Softmax

Figure 4: BanglaBERT and XLM-RoBERTa concatenation of the last layer + MLP Architecture

This captures the sequential patterns of characters in the text using character n-grams of lengths 1, 2, 3, 4 and 5. For example, for n=3, the word "hello" would be represented as [hel, ell, llo]. These character n-grams can capture important linguistic information and patterns in the text, such as common prefixes, suffixes, and other recurring character sequences.

7.2 Deep Learning Models

We used two deep learning models: the Bi-LSTM Network model and Ensemble of Convolutional neural network + Gated recurrent unit (Farhan et al., 2023) both with Bengali GloVe Embeddings (Sarker, 2021). Bengali GloVe Pretrained Word Vectors was pretrained with Wikipedia and crawled news articles with 39 million tokens and has a 0.18 million vocab size. We used the 300d vector version.

7.3 Transformer Models

Throughout our experimentation, we have explored various architectural configurations for these transformer models. To illustrate the general architecture that we employed, we present an example in Figure 4.

7.3.1 BanglaBERT / XLM-RoBERTa / BanglaClickBERT (last layer) + MLP

In this setup, the last layer of the BanglaBERT and XLM-RoBERTa base models are used as the input. The last layer contains contextualized information learned from pretraining on the Bangla and multilingual data, respectively. These representations are then passed through additional linear layers and fine-tuned on the specific task or dataset during the training phase. This allows the model to adapt to the task while benefiting from the pretrained language representation capabilities of BanglaBERT and XLM-RoBERTa.

7.3.2 BanglaBERT / XLM-RoBERTa / BanglaClickBERT (average of all layers) + MLP

Instead of using only the last layer, this setup takes the average of all layers in the BanglaBERT and XLM-RoBERTa base models. By doing so, the model can incorporate information from various depths of the transformers, capturing different levels of context and features. The averaged representations are then fed into linear layers and fine-tuned for the specific task.

7.3.3 BanglaBERT / BanglaClickBERT and XLM-RoBERTa concatenation of the last layer + MLP

In this approach, the outputs from the last layers of BanglaBERT and XLM-RoBERTa are concatenated together. This allows the model to combine the representations learned by each transformer independently. The concatenated representations are then fed into an MLP (multi-layer perceptron) with fully connected layers before producing the final output, which is the prediction for the given task.

8 Experimental Setup

8.1 Prepossessing

The dataset already underwent comprehensive preprocessing, removing HTML tags, URL links, newline escape sequences and emojis. They also preserved all syntactically correct punctuation in the titles and removed punctuation that appeared in the middle of words.

SL	Model Names	Precision	Recall	F1-Score	Accuracy
1	Logistic Regression (with TF-IDF 1-5 n-grams)	0.6540	0.3745	0.4763	0.7102
2	Random Forest (with TF-IDF 1-5 n-grams)	0.6789	0.4509	0.5419	0.7317
3	Bi-LSTM Network (with GloVe Embeddings)	0.6544	0.5877	0.6192	0.7457
4	Ensemble of CNN + GRU (with GloVe Embeddings) (Farhan et al., 2023)	0.6774	0.6103	0.6421	0.7606
5	GAN-BanglaBERT (Mahtab et al., 2023)	0.7545	0.7481	0.7512	**0.8257**
6	BanglaBERT last layer + MLP	0.7377	0.7241	0.7308	0.8088
7	BanglaBERT Large last layer + MLP	0.7349	0.7328	0.7338	0.8124
8	XLM-RoBERTa last layer + MLP	0.7038	**0.7505**	0.7264	0.8134
9	Domain Adaptive BanglaClickBERT last layer + MLP	0.7802	0.7081	0.7424	0.8094
10	BanglaBERT avg of all layers + MLP	0.7293	0.7138	0.7214	0.8018
11	XLM-RoBERTa avg of all layers + MLP	0.6962	0.6474	0.6709	0.7596
12	Domain Adaptive BanglaClickBERT avg of all layers + MLP	0.7717	0.7343	0.7525	0.8214
13	BanglaBERT + XLM-RoBERTa + Embeddings concatenated. Before concatenating passed through one linear layer. Followed by MLP	0.7821	0.7153	0.7472	0.8138
14	Domain Adaptive BanglaClickBERT + XLM-RoBERTa + Embeddings concatenated. Before concatenating passed through one linear layer. Followed by MLP	**0.7896**	0.7234	**0.7551**	0.8197

Table 2: Performance comparison of different Models. Precision, Recall and F1-Score are for the *clickbait* class.

We, furthermore, for our research, extended the preprocessing paradigm by using the Abugida Normalizer and Parser for Unicode Texts (bnunicodenormalizer)[5], enhancing the overall data quality and compatibility. This advanced technique played a pivotal role in fine-tuning later on.

8.2 Experimental Settings

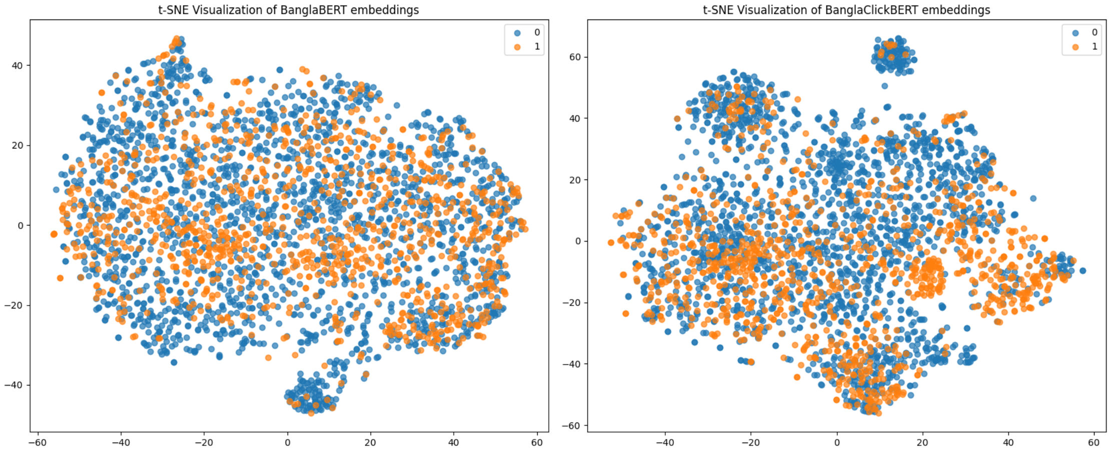
We will be using Statistical models and Deep learning models for Baseline Creation. Then we will be using transformer models. Our main focus is on using Transformer. We have used Transformers with several variations. Based on this variation we try to come up with the best model. For the statistical models, we used TF-IDF vectors and n-grams length from 1 to 5. For the deep learning models, we used 300d Bangla GloVe embeddings. We used a variation of transformers models which we described earlier. We have chosen to mainly use the base (12 layers) versions of these models, as the large (24 layers) models will be computationally expensive and unnecessary for our task. We experimented with BanglaBERT Large model, however, it was providing similar results (discussed in section 9) to the BanglaBERT base model. So, for further experimentation, we continued with the base models. For hyperparameters, we have taken the number of epochs for training as 20, the learning rate is 1e-5, maximum length is 32, batch size of 128, the loss function is Cross Entropy Loss and the optimizer is AdamW (Loshchilov and Hutter, 2019) in all the models. The labeled dataset is divided into three distinct subsets: the training set, test set, and validation set. This allocation was thoughtfully proportioned, with 70% (10839 headlines) of the data reserved for training, 20% (3012 headlines) for testing, and 10% (1205 headlines) for validation purposes. We used the same data splits used in (Mahtab et al., 2023) that helps us to compare with this technique properly. We have used the precision, recall, macro F1-Score and accuracy as measures of evaluation.

9 Results and Analysis

As depicted in Table 2, the statistical models, namely Logistic Regression and Random Forest, failed to identify the clickbait articles fruitfully and exhibited unsatisfactory performance. The deep learning model; The Bi-LSTM model achieved an F1-score of 61.92%. The Ensemble of CNN + GRU (Farhan et al., 2023) performed even better with an F1-score of 64.21%. This highlights the advantages of using word embeddings and sequence modeling for clickbait detection in Bangla. However, there is still considerable room for improvement, as the overall F1-scores remained relatively low.

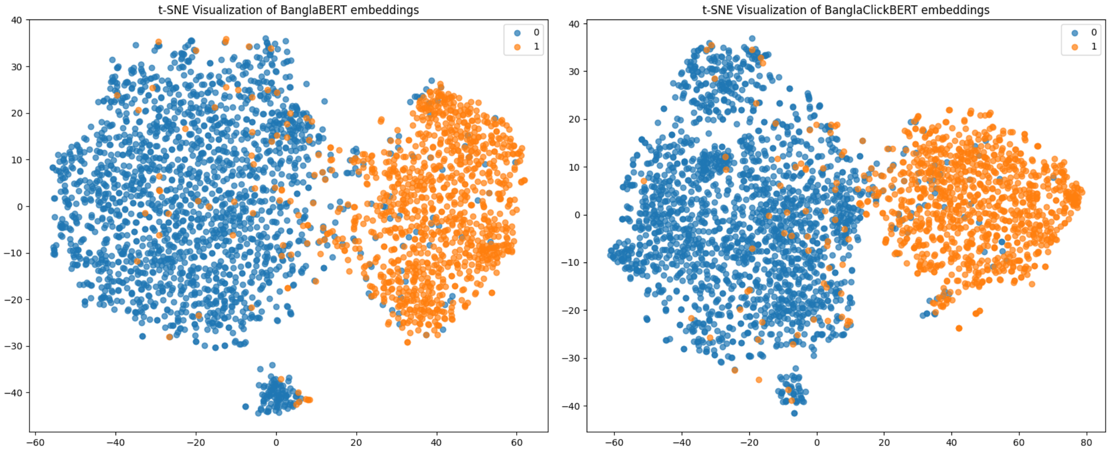
---

[5] `https://pypi.org/project/bnunicodenormalizer/`


t-SNE Visualization of BanglaBERT embeddings

t-SNE Visualization of BanglaClickBERT embeddings

Figure 5: Visualization of last layer hidden representations using t-SNE (van der Maaten and Hinton, 2008) for BanglaBERT (Left) and BanglaClickBERT (Right) without any fine-tuning. 0 represents *non-clickbait* and 1 represents *clickbait* in both figures.


t-SNE Visualization of BanglaBERT embeddings

t-SNE Visualization of BanglaClickBERT embeddings

Figure 6: Visualization of last layer hidden representations using t-SNE (van der Maaten and Hinton, 2008) for BanglaBERT (Left) and BanglaClickBERT (Right) with fine-tuning. 0 represents *non-clickbait* and 1 represents *clickbait* in both figures.

We then from (Mahtab et al., 2023) paper found that, their approach GAN-BanglaBERT achieved a 82.57% accuracy which is the highest accuracy among all the models we described or experimented with. However, this high accuracy does not give us the whole picture as the dataset is imbalanced and a proper evaluation should be according to the macro F1-score which in its case is 75.12% for clickbait class.

Transformer models; BanglaBERT, XLM-RoBERTa and BanglaClickBERT demonstrated consistent improvements over all other approaches. In particular, using only the last layer in conjunction with MLP yielded excellent results. Notably, both BanglaBERT base and BanglaBERT Large performed similarly, 73.08% and 73.38%, indicating that increasing the model's parameters did not contribute significantly to this specific clickbait detection task. On the other hand, the Domain Adaptive BanglaClickBERT exhibited better than that of BanglaBERT and XLM-RoBERTa, respectively, with F1 score of 74.24%. This underscores the effectiveness of pretraining the model with domain-adaptive data for clickbait detection.

Considering the average of all layers proved to be both advantageous and disadvantageous as it captured more informative representations. The F1-score of BanglaBERT and XLM-RoBERTa decreased, whereas, it proved to be beneficial for Domain Adaptive BanglaClickBERT as taking the average of all its layer increased the F1-score by 1.01%. It again proves that pretraining the layers of BanglaBERT has helped all its layers to understand more about clickbait sentences.

Model Names	Attention Weighted Words
BanglaBERT	[CLS] এক মিস ##কল ##েই মধুর সম্পর্ক সর্বনাশ তরুণীর [UNK] . দেখুন (ভিডিও) [SEP]
BanglaClickBERT	[CLS] এক মিস ##কল ##েই মধুর সম্পর্ক সর্বনাশ তরুণীর [UNK] . দেখুন (ভিডিও) [SEP]
Raw Headline	এক মিস কলেই মধুর সম্পর্ক সর্বনাশ তরুণীর. দেখুন (ভিডিও)
Translated Headline	One missed call destroys the sweet relationship of the young woman. Watch (Video)

Table 3: Comparison between finetuned BanglaBERT and finetuned BanglaClickBERT. A clickbait sentence is chosen and both the model predict it as clickbait. Each word is highlighted according to their attention weights.

Moreover, concatenating the embeddings from two pretrained language models further enhanced performance, illustrating that combining related models could capture complementary information for clickbait detection in Bangla. The combination of Domain Adaptive BanglaClickBERT and XLM-RoBERTa achieved the highest F1-score of 75.51% for the clickbait class surpassing other models we discussed about including the GAN-BanglaBERT (Mahtab et al., 2023).

In terms of precision, recall, F1-score, and accuracy, the Domain Adaptive BanglaClickBERT model proved to be more consistent to all other models. However, since the F1-scores were tightly clustered within the range of 0.74 to 0.75, to support our claims, we ran each model ten times with different seeds and conducted a statistical test. The model, labeled *"Domain Adaptive BanglaClickBERT + XLM-RoBERTa + Embeddings concatenated. Before concatenating passed through one linear layer. Followed by MLP"* outperforms all other models, and the difference in performance is statistically significant ($p < 0.05$) according to McNemar's test (Dietterich, 1998)."

This finding is further supported by the t-SNE visualization depicted in Figure 5 and Figure 6. The t-SNE visualization effectively shows how these models, even without fine-tuning of the training data, group their predictions. It becomes evident that BanglaClickBERT exhibits better clustering than BanglaBERT, underscoring the idea that training BanglaClickBERT can enhance the learned representations and subsequently improve overall performance. This can be shown more prominently in Figure 6 that BanglaClickBERT managed to cluster the embeddings of clickbait headlines better than BanglaBERT.

Additionally, as illustrated in Table 3, using the *Transformers Interpret* (Pierse, 2021) we tried to analyse how the models predict their predictions. Green highlights indicate supportive words for the prediction, while red highlights show opposing words. Brightness reflects the strength of their contribution or opposition. We can see that, finetuning the BanglaBERT and BanglaClickBERT models results in different attention patterns for words. In particular, BanglaClickBERT allocates greater attention to words related to clickbait, a characteristic that BanglaBERT does not achieve.

In conclusion, the results suggest that BanglaClickBERT, proves to be highly effective for clickbait detection in Bangla. If more and better labelled data is used to finetune this, this approach will perform better than other approaches.

10 Conclusion

In conclusion, this study represents a significant advancement in the field of clickbait detection, particularly for the Bangla language, where research has been limited. While clickbait detection in English has been extensively studied, the Bangla news headlines have been largely overlooked. To address this gap, we conducted a comprehensive analysis using state-of-the-art transformer models, such as BanglaBERT, XLM-RoBERTa, and the newly developed BanglaClickBERT. We enhanced the performance of these models by incorporating MLP methods to achieve the best results. To bolster the research, we augmented the dataset by including an additional 1 million unlabeled Bangla news headlines, sourced from clickbait-dense websites. This expanded dataset significantly empowered the BanglaClickBERT model. Through rigorous experimentation and testing, our approach showed better results compared to existing state-of-the-art techniques. Our work not only contributes to the improvement of clickbait identification in Bangla news headlines but also fills the void in research in this language domain. As clickbait continues to impact the way information is consumed, our findings will be valuable for media organizations, content creators, and platforms to promote responsible and reliable information dissemination in the Bangla-speaking community.

11 Acknowledgement

We extend our sincere appreciation to the dedicated faculty members of the Computer Science and Engineering department at BRAC University, Bangladesh. Their unwavering support and valuable guidance have been integral to our research efforts. Additionally, we express our gratitude to the anonymous reviewers whose constructive feedback has significantly contributed to the improvement of our research work.

References

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Ahmad, Kazi Samin Mubasshir, Md Saiful Islam, Anindya Iqbal, M. Sohel Rahman, and Rifat Shahriyar. 2022. BanglaBERT: Language model pretraining and benchmarks for low-resource language understanding evaluation in Bangla. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1318–1327, Seattle, United States. Association for Computational Linguistics.

S. Elizabeth Bird. 2008. *Tabloidization*. John Wiley & Sons, Ltd.

Prakhar Biyani, Kostas Tsioutsiouliklis, and John Blackmer. 2016. "8 amazing secrets for getting more clicks": Detecting clickbaits in news streams using article informality. In *AAAI Conference on Artificial Intelligence*.

Mark Bronakowski, Mahmood Al-khassaweneh, and Ali Al Bataineh. 2023. Automatic detection of clickbait headlines using semantic analysis and machine learning techniques. *Applied Sciences*, 13(4).

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.

Staff Correspondent. 2022. Prothom alo at the top with 5 million readers. *Prothomalo*. [Online].

Thomas G. Dietterich. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Comput*, 10(7):1895–1923.

Niloy Farhan, Ishrat Tasnim Awishi, Md Humaion Kabir Mehedi, MD. Mustakin Alam, and Annajiat Alim Rasel. 2023. Ensemble of gated recurrent unit and convolutional neural network for sarcasm detection in bangla. In *2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 0624–0629.

Sura Genç and Elif Surer. 2021. Clickbaittr: Dataset for clickbait detection from turkish news sites and social media with a comparative analysis via machine learning algorithms. *Journal of Information Science*, 49:480 – 499.

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.

Md Zobaer Hossain, Md Ashraful Rahman, Md Saiful Islam, and Sudipta Kar. 2020. BanFakeNews: A dataset for detecting fake news in Bangla. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2862–2871, Marseille, France. European Language Resources Association.

Vijayasaradhi Indurthi, Bakhtiyar Syed, Manish Gupta, and Vasudeva Varma. 2020. Predicting clickbait strength in online social media. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4835–4846, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Md Saroar Jahan, Mainul Haque, Nabil Arhab, and Mourad Oussalah. 2022. BanglaHateBERT: BERT for abusive language detection in Bengali. In *Proceedings of the Second International Workshop on Resources and Techniques for User Information in Abusive Language Analysis*, pages 8–15, Marseille, France. European Language Resources Association.

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

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Md. Motahar Mahtab, Monirul Haque, and Farig Sadeque. 2023. Banglabait: Semi-supervised adversarial approach for clickbait detection on bangla clickbait dataset. In *Proceedings of Recent Advances in Natural Language Processing*, pages 744–754, Varna, Bulgaria.

Mahmud Hasan Munna and Md Shakhawat Hossen. 2021. Identification of clickbait in video sharing platforms. In *2021 International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI)*, pages 1–6.

Supavich Fone Pengnate, Jeffrey Chen, and Alex Young. 2021. Effects of clickbait headlines on user responses: An empirical investigation. *Journal of International Technology and Information Management*, 30(3):1.

Charles Pierse. 2021. Transformers Interpret.

Martin Potthast, Tim Gollub, Matthias Hagen, and Benno Stein. 2018a. The clickbait challenge 2017: Towards a regression model for clickbait strength. *CoRR*, abs/1812.10847.

Martin Potthast, Tim Gollub, Matti Wiegmann, Benno Stein, Matthias Hagen, Kristof Komlossy, Sebstian Schuster, and Erika P. Garces Fernandez. 2018b. Webis clickbait corpus 2017 (webis-clickbait-17).

Martin Potthast, Sebastian Köpsel, Benno Stein, and Matthias Hagen. 2016. Clickbait detection. In *Advances in Information Retrieval*, pages 810–817, Cham. Springer International Publishing.

Abdul Razaque, Bandar Alotaibi, Munif Alotaibi, Shujaat Hussain, Aziz Alotaibi, and Vladimir Jotsov. 2022. Clickbait detection using deep recurrent neural network. *Applied Sciences*, 12(1).

Md Main Uddin Rony, Naeemul Hassan, and Mohammad Yousuf. 2017. Diving deep into clickbaits: Who use them to what extents in which topics with what effects? In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017*, ASONAM '17, page 232–239, New York, NY, USA. Association for Computing Machinery.

Sagor Sarker. 2021. BNLP: natural language processing toolkit for bengali language. *CoRR*, abs/2102.00405.

Philippe Thomas. 2017. Clickbait identification using neural networks.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

Yiwei Zhou. 2017. Clickbait detection in tweets using self-attentive network.

A Appendix

A.1 Distinguishing between clickbait and non-clickbait

The following examples shows the differences between clickbait headlines and non-clickbait headlines:

Examples 1:

নায়কের চেয়েও বেশি টাকা আয় করেন এই কমেডিয়ান ! জানেন কত... ?

Translated: This comedian earns more money than the hero! Do you know how much...?

Examples 2:

দীপিকার ব্যাগে যা থাকে! (ভিডিও)

Translated: What's in Deepika's bag! (video)

Examples 3:

বলিউডের ফিল্মফেয়ার অ্যাওয়ার্ড পেতে যাচ্ছেন বাংলাদেশের তন্বী

Translated: Bangladesh's Tanvi is going to receive Bollywood's Filmfare Award

In the above mentioned examples we can observe that clickbait headlines have distinguish patterns. In the Example 1, the headline does not immediately reveal who the more-earning comedian is. Instead, it keeps the reader in suspense, prompting them to click in order to satisfy their curiosity and uncover the answer.

Example 2, the headline uses the "Curiosity Gap" technique by teasing an intriguing element of the story without giving away the full details. Moreover, the excessive use of punctuation or other symbols, such as exclamation points, is often used to heighten the reader's curiosity and create a sense of urgency or excitement.

On the other hand, Example 3 falls into the category of non-clickbait due to its straightforward and informative headline. It effectively communicates the content and purpose of the article, leaving no room for curiosity or teasing. This type of headline is transparent and does not rely on sensationalism or misleading tactics to attract readers, making it a clear example of non-clickbait.

Story Co-telling Dialogue Generation based on Multi-Agent Reinforcement Learning and Story Hightlights

Yu-Kai Lee and **Chia-Hui Chang**
Department of Computer Science & Information Engineering,
National Central University

Abstract

Retelling a story is one way to develop narrative skills in students, but it may present some challenges for English as Second Language (ESL) students who are learning new stories and vocabularies at the same time. The goal of this research is to develop a dialogue module for story co-telling for ESL students in order to help students to co-narrate an English story and enhance their narrative skills. However, story co-telling is a relatively underexplored and novel task. In order to understand the story content and select the right plot to continue the story co-telling based on the current dialogue, we utilize open domain information extraction techniques to construct a knowledge graph, and adopt multi-agent reinforcement learning methods to train two agents to select relevant facts from the knowledge graph and generate responses, jointly accomplishing the task of story co-telling. Compared to models that reply on chronological order, our model improves the performance from 67.01% to 70.81% through self-training with reward evaluation, achieving an increase of approximately 3.8%.

1 Introduction

Story retelling is one of the methods to enhance students' narrative abilities. However, due to weaker language proficiency, difficulty in organizing complex plots, or encountering obstacles in expressing ideas and emotions, not every student can fully elaborate on a story independently. To address this issue, we propose the task of Story Co-telling based on the concept of Scaffolding Theory (Wood et al., 1976) to assist students in story retelling. The notion of Scaffolding Theory draws an analogy from construction, where temporary support is provided during building construction, and it is removed once the construction is complete or learning is mature. Similar to training wheels when learning to ride a bicycle, Story Co-telling offers necessary support to students when needed and gradually reduces assistance as their narrative skills improve.

The objective of this study is to develop a Story Co-telling dialogue module aimed at assisting ESL students in collaboratively narrating lengthy English stories to foster narrative abilities. To refine the study's focus, we constrain the dialogue module to engage only in conversations related to story co-telling, rather than purposeless chitchat. Thus, our dialogue module is designed as a Supportive Story Chatbot, which, based on the student's ongoing narrative, determines the next plot to be told, achieving the collaborative narration of the story between two participants.

Story Co-telling is a relatively less explored and novel task, distinct from common story generation tasks. While story generation concentrates on generating logical subsequent plots, story co-telling is grounded in the content of the original story. This difference necessitates a reconsideration of model design and training methods. Since story co-telling is an interactive process between two participants, we anticipate employing reinforcement learning techniques to implement the Story Co-telling module.

However, designing a story Co-telling dialogue system based on reinforcement learning presents four primary challenges. First, it can be time-consuming and costly if we would train a dialogue system through online reinforcement learning, where the system learns from actual interactions with people. Second, utilizing offline reinforcement learning requires suitable dialogue corpora for Story Co-telling, which currently do not exist, necessitating the generation of relevant datasets. Third, the efficacy of reinforcement learning models hinges on well-defined reward functions. The task of determining how to establish appropriate environmental rewards for each dialogue round constitutes a significant challenge. Finally, when dealing with long story texts, how the agents can

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 11–19
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

comprehend the entire content and choose the next coherent story plot or event is a significant challenge.

Inspired by the research by Andrus et al. (2022), we develop a Story Co-telling dialogue module based on an open-domain information extraction to condense the content of lengthy story texts,and introduce Multi-Agent Reinforcement Learning (MARL) technology to enhance the coherance and relevance of the story co-telling task. MARL involves two agents making optimal responses based on dialogue history and the Knowledge Graph built on OpenIE.

We further leverage the power of large-scale language models (LLM) to design reward functions to evaluate the quality of narratives. Specifically, we can train the reward function by carefully preparing the training data: assuming that the story highlights summarized by the LLM represent good storytelling, then modifying the story highlights by removing and adding irrelevant storylines can represent poor narrative.

By using the subjects, predicates and relationships extracted by OpenIE as the agent's action set, our model can make more informed choices across different decision contexts. Through self-trained reward evaluation, we observe that our model's performance improves from 67.01% to 70.81%, a gain of approximately 3.8%, as compared to responding solely in chronological order. This improvement indicates the feasibility of our model.

2 Related Work

The application of dialogue robots in education has garnered widespread attention. Various educational practitioners hold diverse expectations for the roles and functionalities that educational robots should embody.

For instance, the education team at the University of California, Irvine developed a system named StoryBuddy that accompanies parents and children in reading stories together. During the reading process, this system integrates question-and-answer interactions to enhance parent-child engagement (Zhang et al., 2022). They introduced the FairyTaleQA dataset (Xu et al., 2022) and employed Question Answer Generation (QAG) to address the challenge of generating questions for parents. Through experiments, it was found that implementing companion-based reading through questioning and answering enhances children's comprehension when responding to questions (Xu et al., 2021).

On the other hand, Chu and Min (2021) developed a dialogue robot specifically for retelling elementary school English storybooks. This dialogue robot assists learners in retelling stories by asking questions and utilizes rule-based mechanisms to determine whether each scene has been accurately recounted. For instance, if the first scene has been correctly mentioned, the robot prompts the student to narrate subsequent scenes. If a scene hasn't been correctly mentioned, the student is asked to retell it. Through this iterative process, students are guided step by step to independently retell the entire story. The aforementioned approach demonstrates the potential of story dialogue robots in promoting parent-child interactions, cultivating reading interests, and enhancing narrative skills.

Continuing with the theme of enhancing children's narrative abilities, recent research has also focused on utilizing information extraction techniques to comprehend and analyze long-text narratives. These techniques aim to transform unstructured textual data into structured information. For instance, Xu et al. (2023) developed a Document-level Narrative Event Chain Extraction Toolkit (NECE). This approach employs technologies such as Semantic Role Labeling (SRL) to extract relevant information about characters and events from stories. Furthermore, a specific TF-IDF algorithm is used to identify the most important events. Through this framework, the narrative structure within lengthy textual stories can be effectively parsed, enabling the extraction of essential elements like characters and events.

Similarly, Andrus et al. (2022) address the challenge of understanding long-text narratives using dynamic knowledge graphs. Unlike static commonsense knowledge graphs that involve real-world information, Andrus et al. (2022) utilize OpenIE (Open Information Extraction) technology to construct dynamic knowledge graphs. These dynamic knowledge graphs are then applied to tasks such as question answering and story completion. This approach proves effective in overcoming the limitations imposed by language model input constraints when dealing with lengthy documents, and its effectiveness has been demonstrated.

3 Method

The MARL structure for story co-telling based on knowledge graph construction is shown in Figure

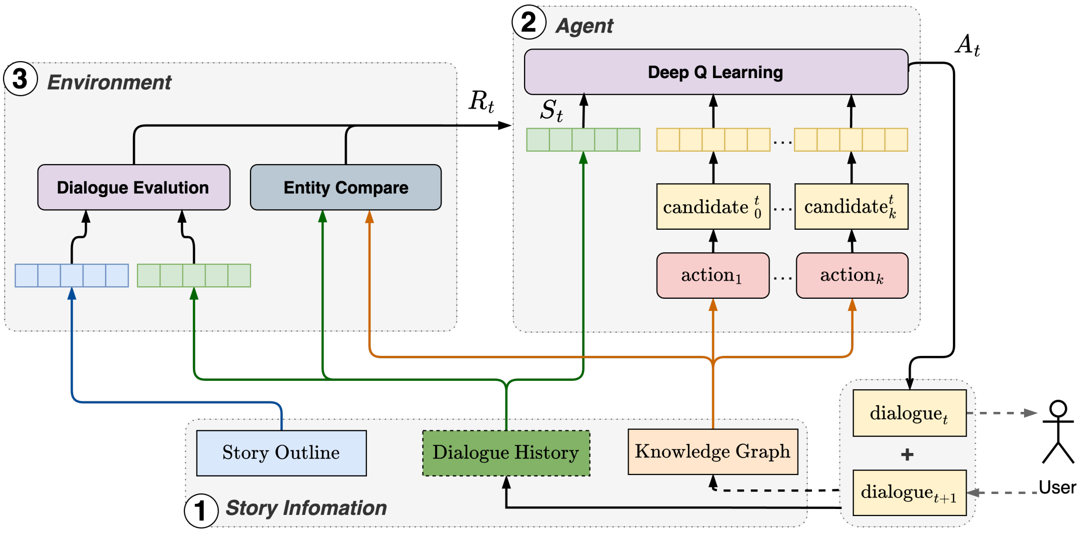


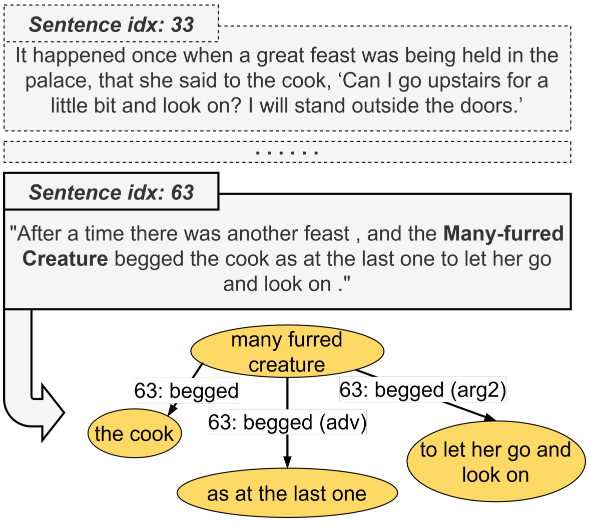
Figure 1: Architecture of the Story Co-telling Module via Reinforcement Learning and Knowledge Graph

1. We will start from how to convert a long text story into a knowledge graph and introduce how the agent uses conversation history and knowledge graph to select the plot to be told next. Secondly, we explain how to construct a dialogue history evaluation model for evaluating the current performance of story co-telling. Finally we will explain how to use reinforcement learning to integrate the above parts into a story sharing dialogue module that can make decisions based on the current dialogue history.

3.1 Long Text to Knowledge Graph

The purpose of constructing knowledge graphs is to distill information from lengthy text narratives and transform unstructured data into a structured form. This enables our model to effectively comprehend the storyline of the narrative. We utilize Stanford CoreNLP toolkit, the OpenIE (Open Information Extraction) framework (Angeli et al., 2015), version 4.5.4, to extract structured fact triples (i.e. subject, relation, and object) from text.

For example, consider the sentence "After a time there was another feast, and the Many-furred Creature begged the cook as at the last one to let her go and look on." Even though this sentence describes "the Many-furred Creature begged the cook to let her go and look on, just like the last time," due to the constraints of the triple representation, the second object, time, location, and other words need to be separately recorded. Hence, the preceding sentence can be represented by three fact triples: [many furred creature, begged, the cook], [many furred creature begged the cook, adv, as at the last one], and [many furred creature begged the cook, arg2, to let her go and look on]. These triples are then visualized as a directed graph, as depicted in Figure 2.



It happened once when a great feast was being held in the palace, that she said to the cook, 'Can I go upstairs for a little bit and look on? I will stand outside the doors.'

"After a time there was another feast , and the **Many-furred Creature** begged the cook as at the last one to let her go and look on ."

Figure 2: Example of Constructing Knowledge Graph Using OpenIE

To mitigate potential redundancy in the fact triples produced by open-domain information extraction models, we remove duplicate triples and retain longer ones to preserve more information. Additionally, we also employ Coreference Resolution (Recasens et al., 2013) to process the text and replace pronouns with the nouns they refer to.

In practice, in addition to the subject, relation, and object triples, we also record the sentence index $sidx$ of each fact triple in the original story to understand the context of the fact triples. Additionally, we also keep a status indicator for each fact to record whether it has been mentioned in the conversation. This helps prevent repeated references to the same fact during the narrative.

3.2 Agent

In this paper, we employ Deep Q-Learning for designing the conversational agent. The agent makes decisions based on the current state S_t, takes the next action A_t, and adjusts its decisions according to the feedback rewards R_t generated by the environment. Here, S_t is a vector composed of various pieces of information, including the conversation history $D = [u_0, u_1, ..., u_t]$, and candidate responses $C_t = [c_0^t, c_1^t, ..., c_k^t]$ generated by corresponding strategies $A = [a_0, a_1, ..., a_k]$. We use Sentence Transformer (Reimers and Gurevych, 2019) to convert these text fragments into vectors expressing their underlying information. After passing through Deep Q Learning, the agent selects the candidate response to be used for the reply, which determines the next action $A_t = i$, where $i \in [0, k]$. We will now introduce the action design of the agent and the methods for generating candidate responses. The details of the reinforcement learning will be discussed in the subsequent sections.

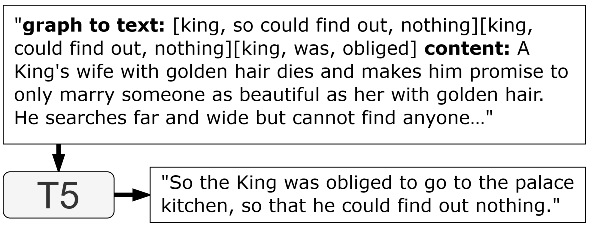
Action Design

To ensure coherence in the co-told story, the agent, based on the latest utterance u_t in the conversation history, utilizes the Sentence Transformer to find the top three relevant facts on the knowledge graph G as reference points p of the interlocutor's current narration. Subsequently, using these reference points, four distinct strategies are employed to extend the conversation, thereby generating candidate responses. Each strategy is treated as an action a_i. Here's a brief description of each action:

- a_0: Select subsequent events from the reference point. In other words, choose facts f where $f.sidx$ is greater than $p.sidx$.
- a_1: Choose facts with subjects similar to the subject of the reference point p.
- a_2: Choose facts with relations similar to the relation of the reference point p.
- a_3: Choose facts with objects similar to the object of the reference point p.
- a_4: Declare the end.

Response Generation

We can utilize the story sentences, along with their corresponding fact triples obtained using OpenIE, to prepare training data for T5 model training, i.e. create an input-output mapping using the facts triples and story outline as input and the sentence that contributes the fact triples as output. By fine-tuning, we enable the T5 model to generate results similar to the original sentences based on the given fact triples and the story outline. An example input format is depicted in Figure 3.



"**graph to text:** [king, so could find out, nothing][king, could find out, nothing][king, was, obliged] **content:** A King's wife with golden hair dies and makes him promise to only marry someone as beautiful as her with golden hair. He searches far and wide but cannot find anyone..."

"So the King was obliged to go to the palace kitchen, so that he could find out nothing."

Figure 3: Fine-tuning the T5 Model for Knowledge Graph to Text Generation

3.3 Environment: Reward Function Design

The reward function is mainly divided into two parts: dialogue history assessment and entity connection assessment. The former provides an overall rating of the dialogue up to the current point, while the latter calculates the connection rating between the current turn and the previous sentence.

Dialogue History Assessment

To evaluate the effectiveness of the co-told story dialogue history D, we require both positive and negative co-telling examples along with their ratings. These examples can be used to train a regression model for automatically assessing the quality of co-told stories.

Due to the lack of readily available co-telling dialogue datasets, we utilize ChatGPT to generate a specified number of bullet-pointed story highlights for each story. As shown in Table 1, we design a prompt to guide ChatGPT in generating the desired number of story highlights H for the story text. To facilitate further processing, the generated results are output in JSON format. Considering ChatGPT's generation diversity, the same prompt can lead to various outcomes.

We generated story highlights using ChatGPT and subsequently performed actions such as replacement or deletion to create lower-quality story highlights. This approach of generating story high-

Input
<|Plots|> = number of plots that you want to generate <Story_text> = story corpus
Prompt
Please summarize the following Story by outlining <$\|Plots\|$> plot points in JSON format in order. (example: [{“plot_id": 1, “plot_point": first plot point}, {“id": 2, “plot_point": second plot point}]) Do not provide additional information or comment. – Story: <Story_text>

Table 1: Prompt Format for Generating Story Highlights Using ChatGPT

lights can be seen as generating poor examples in co-telling, as they may disrupt the integrity and logic of the story. Depending on the number of replacements or deletions, we assign different scores.

As the impact of replacement and deletion on the quality of story highlights differs, we have formulated separate adjustment formulas and evaluation formulas for these two actions. The formula for deleting story highlights is presented in Eq.(1), while the formula for replacing story highlights is shown in Eq.(2).

$$score = e^{(-1.6\times\frac{n}{|Plots|})} \times 9 + 1 \tag{1}$$

$$score = e^{(-4\times\frac{n}{|Plots|})} \times 10 + 1 \tag{2}$$

Here, n represents the number of modifications, and $|Plots|$ represents the original number of story highlights. We believe that replacing an existing story highlight with another storyline has a greater impact on the overall quality compared to deleting a single story highlight. As a result, replacing a larger number of story highlights will receive a lower score compared to deleting the same number of story highlights (see Figure 4).

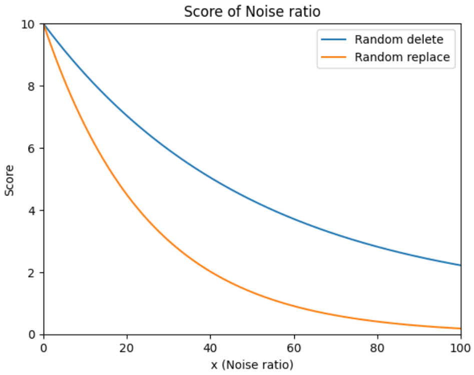


Figure 4: Score vs. # of edit operations

Dialogue history assessment is essentially a regression problem, as illustrated in Figure 5. We input both the dialogue history D and the story outline H into the same RoBERTa (Liu et al., 2019) model, and extract the hidden state of the CLS token from the model. Subsequently, the two hidden states are concatenated and fed into a neural network. This network outputs a score $DH(D, H)$ between 0 and 10 to evaluate the quality of the co-told story.

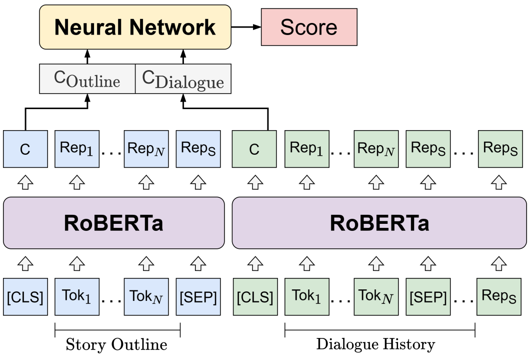


Figure 5: Architecture of the Dialogue History Assessment Model

Entity Relationship Evaluation

The purpose of entity relationship evaluation is to assess whether the current reply (R) is related to the entities (E) mentioned in the previous sentence of the story. We utilize OpenIE to parse these two sentences and employ BFS graph algorithm to determine if these two entities can be connected in the knowledge graph. If the two entities are linkable in the knowledge graph, we consider there is an entity relationship between these two sentences and provide quantitative rewards as feedback.

We compute the score $DH_t = DH(D_t, H)$ for dialogue history assessment and the entity connection assessment score $EC_t = EC(R_t, E_{t-1})$ for each round t. Since dialog history assessment is an accumulated score, we thus take the score difference of two subsequent rounds along with the entity connection score as the reward R_t for this round as indicated in Eq. (3). This reward is subsequently fed back to the agent.

$$R_t = DH_{t-1} - DH_t + EC_t \tag{3}$$

3.4 Multi-Agent Reinforcement Learning

Finally, we apply Deep Q Learning (DQL) and Multi-Agent Reinforcement Learning (MARL) methods to enable two agents to collaboratively perform the task of co-telling a story (see Figure 1). Through the guidance of reward scores, the agents

turn	history	score
...	...	...
6	The Princess falls asleep in a hollow tree and is discovered by the King's huntsmen.	7.37
7	The King's huntsmen bring the Princess to the palace and she is assigned to work in the kitchen as the Many-furred Creature.	7.34
8	The Many-furred Creature lives in poverty and works in the kitchen doing all the dirty work.	7.79
9	The Many-furred Creature attends a feast at the palace and enchants the King with her beauty.	7.95
10	The Many-furred Creature cooks soup for the King and hides a gold ring in it.	8.06
...	...	...
14	The King and the Princess live happily ever after.	8.02
Final Score=8.02, Gold=9.09		

turn	history	score
...	...	...
6	The Princess falls asleep in a hollow tree and is discovered by the King's huntsmen.	7.37
7	**The Emperor takes Confucius' shoes and staff as a joke, but the tablet's warning comes true and he dies soon after.**	**6.34**
8	**The cock gets the garland and trades it for red silk from the brook.**	**4.82**
9	**The jackdaws and magpie eat the leftover pie-crust and gravy.**	**2.74**
10	The Many-furred Creature cooks soup for the King and hides a gold ring in it.	2.17
...	...	...
14	The King and the Princess live happily ever after.	6.21
Final Score=6.21, Gold=7.38		

Table 2: Examples of conversation history evaluation model. The left table shows high-quality storyline highlights (which received a score of 9), while the right table shows cases where the inclusion of irrelevant content resulted in a drop in reward points.

learn how to continue the story. While the story co-telling agent will only focuses on a single story during the interaction with the user, updating the model based on a single story is dangeous because the model is likely to forget what it has learned in the past. Therefore, We choose to adopt experience replay mechanism to avoid catastrophic forgetting.

Our objective is to enable two agents to collaboratively co-tell a story. In each dialogue turn, the agents take turns transmitting the selected response through the environment, without sharing their respective knowledge graph states. This implies that each agent can only understand the co-told story and make appropriate responses based on the co-telling conversation history. If one of the agents terminates prematurely, the entire dialogue also ends, followed by subsequent analysis and evaluation. This design simulates real-world human-machine interaction scenarios, challenging the agents' understanding and response decision-making abilities.

Before the training begins, we will initialize each environment and model (lines 1 to 5). In each epoch (line 6), we engage in a dialogue for each story (line 7), simultaneously initializing the environment state before the co-telling begins (lines 8 to 11). In lines 12 to 23, it can be observed that the two agents take turns generating candidate responses, connecting their vectors with the dialogue history vector to form the current state representation (lines 13 to 14). Subsequently, the agents use their own Q Network to decide which candidate response to select (lines 15 to 16). Following this, we employ the Dialogue Evaluation Model and Entity Compare to generate rewards (lines 17 to 19), while also producing the next state (lines 20 to 21). Finally, the tuples of state transition, action, next state, and corresponding reward (s, a, s_{t+1}, r_{t+1}) are stored in their respective memories (line 22), for subsequent learning and updating processes.

4 Experiment

In this study, we chose stories from FairytaleQA (Xu et al., 2022) as the designated story set for story co-telling. These stories are classic fairy tales suitable for readers below the ninth grade, with clear narrative structures. The average text length of stories used in FairytaleQA exceeds one thousand words. Additionally, with the pre-designed question-answer pairs available in FairytaleQA, we can evaluate the diversity of co-telling content through question answering.

To ensure the effectiveness of agent training, we set some termination conditions for the environment. Firstly, by limiting the conversation rounds to be no more than 20, we avoid resource wastage and increased training time caused by excessively lengthy dialogues. Additionally, when one of the participants introduces an ending keyword, it signifies an appropriate endpoint for the conversation. Furthermore, we set the exhaustion of all facts in the knowledge graph as one of the ending conditions. This configuration ensures efficient utilization of information during the conversation and prevents the repetitive use of the same facts.

Algorithm 1: Story Co-telling MARL

```
Data:
    $I = [(O_1, G_1), (O_2, G_2), ...]$ Story info.;
    $O_j$ = Story outline;
    $G_j$ = Story knowledge graph;
Function:
    $\mu$ = State embedding model;
    $\Phi$ = Candidate response generate func.;
    $\Theta$ = Dialogue evaluation model;
    $\Xi$ = Entity compare func.;
Training:
Initialize $Agnet1$ and $Agnet2$;
Initialize Q Network $Q_1$ and $Q_2$;
Initialize epsilon $\varepsilon$;
Initialize replay memory $M_1$ and $M_2$;
Initialize environment $E_1$ and $E_2$;
foreach epoch do
    foreach $(O_j, G_j)$ in $I$ do
        Reset dialogue history $D$;
        Reset environment $E_1$ and $E_2$ by $(O_j, G_j)$;
        $t = 1$;
        $Score_t = 0$;
        while ($E_1$ is not done) and ($E_2$ is not done) do
            $C_t \leftarrow \Phi(D, G)$;
            $s_t \leftarrow \{\mu(D), \mu(C_t)\}$;
            $a_t \leftarrow argmax(Q_{t\%2}(s_t, \varepsilon))$;
            $d_t \leftarrow C_t[a_t]$;
            Append $d_t$ to $D$;
            $Score_{t+1} \leftarrow \Theta(O_j, D) + \Xi(G_j, D)$;
            $r_{t+1} \leftarrow Score_{t+1} - Score_t$;
            $C_{t+1} \leftarrow \Phi(G)$;
            $s_{t+1} \leftarrow \{\mu(D), \mu(C_{t+1})\}$;
            Append $(s, a, s_{t+1}, r_{t+1})$ to $M_{t\%2}$;
            $t = t + 1$;
        end
        Update $Q_1$ by $M_1$;
        Update $Q_2$ by $M_2$;
    end
    Update $\varepsilon$;
end
```

4.1 Dialogue History Evaluation Model

During the training of the dialogue history evaluation model, we set the batch size to 1 and conducted 20 training epochs. Across these training sessions, the loss value on our training set was 0.0197, indicating a strong fit of the model to the training data (Figure 6). The best validation set loss was 0.0299, demonstrating satisfactory performance on unseen data. Additionally, we computed the Pearson correlation coefficient between the scoring values and the dialogue history evaluation model, yielding a value of 0.8313, indicating a positive correlation between the data labels (given by Eq. (1), (2)) and the model's outputs.

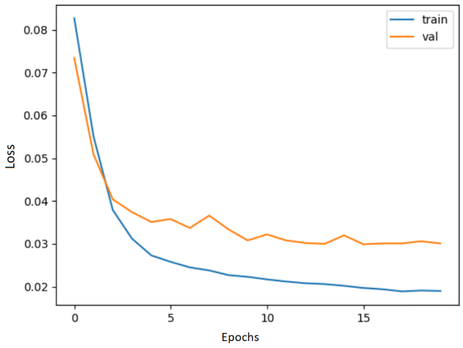


Figure 6: Training of the Dialog History Assessment Model in Figure 5

Table 2 presents the scoring results provided by the dialogue history evaluation model on two conversation history examples. The "score" column displays the cumulative score from the first utterance up to the current turn, and the gold score for the entire conversation are marked at the bottom. As shown in the example, when the input contains high-quality story focus, the model's output results closely match the default scores. This indicates that our dialogue history evaluation model can accurately assess story focus and assign appropriate scores. If irrelevant story focus is inserted into the story, the scores given by the dialogue history evaluation model significantly decrease. This further demonstrates the effectiveness and feasibility of our dialogue history evaluation model, as it can identify relevant story focus and provide appropriate evaluations for them.

4.2 Effectiveness of Story Co-Telling Models

Secondly, we conducted a performance comparison with rule-based responses, which involves responding solely based on chronological order, i.e. a_0. Figure 7 illustrates our training results, demonstrating that both single-environment reinforcement learning (1Env) and multi-environment reinforcement learning (2Env) outperform the rule-based responses. The performance of multi-environment reinforcement learning is the best. According to the feedback values from our trained dialogue history evaluation model, the performance of multi-environment reinforcement learning has improved by approximately 3.8%, from 67.01% to 70.81%, compared to responses based on chronological order, i.e. choose action a_0.

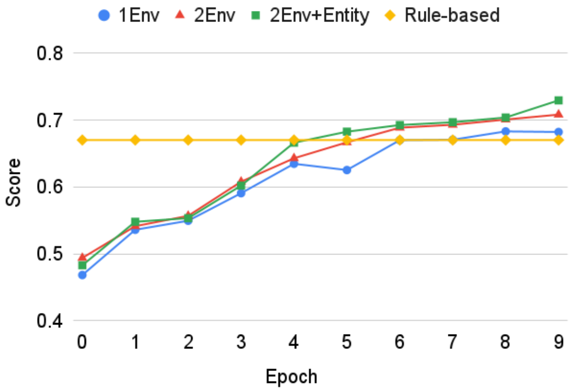


Figure 7: Comparison of Results from Story Co-Telling Trained with Different Methods

This result indicates the feasibility of multi-agent reinforcement learning methods in the story co-telling task. Compared to rule-based responses that rely solely on chronological order, our model, trained through the interaction of multiple agents, can better comprehend dialogue history and generate responses based on the knowledge graph. This enables our model to provide more coherent and relevant replies, further enhancing the quality and experience of the conversation.

4.3 Comparison of Reward Function Design

Next, we investigate the effect of incorporating entity connection EC reward on the model's action selection. As shown in Figure 8, we can observe that both the average EC and DH reward increase over the course of training. Furthermore, in comparison to using only the dialogue history evaluation model as the sole reward (DialogueEvaluation), under the encouragement of entity relationship evaluation (DialogueEvaluation + EntityCompare), the model tends to choose actions related to entities (as shown in Figure 9). This indicates that the approach of introducing entity comparison into the dialogue history evaluation model has a certain impact on the model's decision-making process.

4.4 Discussion: Evaluation of Co-told Stories

Finally, we try to evaluate whether the co-told stories are good or bad. One possible way is to use question answering to test whether the story hightlights can answer the pre-designed questions. We conducted experiments using a fine-tuned T5 question-answering model (Christian Di Maio, 2022) based on the story summaries. We replaced the story paragraphs corresponding to questions in FairytaleQA with the story summaries to evaluate whether the story summaries could effectively answer questions from the stories.

The experimental results are presented in Table 3. The performance of this fine-tuned T5 model on story summaries is not ideal. This is mainly because the story summaries are relatively short, lacking details and context, which makes it difficult for the question-answering model to provide accurate answers. Additionally, the story summaries might contain implicit information, requiring the model to possess stronger reasoning abilities to handle such implied content.

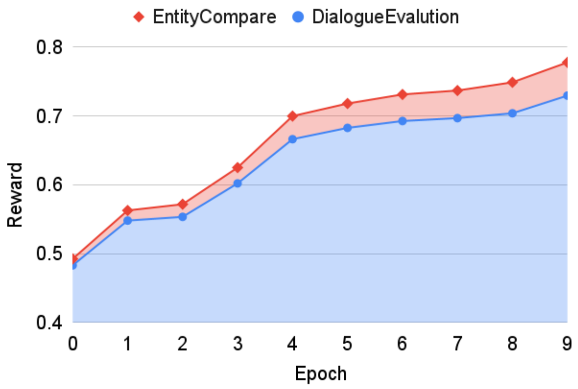


Figure 8: Stacked Area Chart of Entity Relationship Reward during Training Process

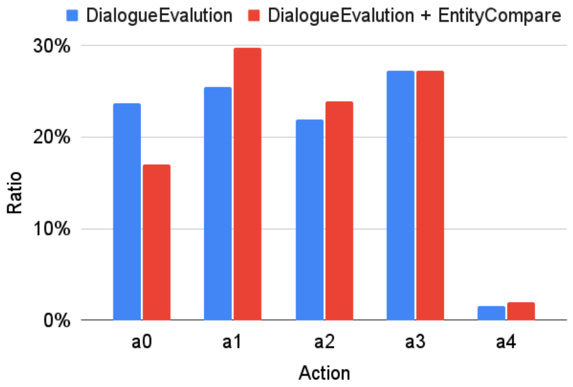


Figure 9: Change of Action Selection (in Section 3.2) Histogram

Question Types	Train		Val		Test	
	F1	EM	F1	EM	F1	EM
character	**24.11**	**16.53**	**27.33**	**18.69**	**20.41**	**11.65**
action	11.85	2.19	13.64	3.00	13.27	2.54
setting	15.50	6.50	23.64	6.67	14.34	3.23
feeling	4.60	3.28	3.26	1.06	7.97	4.72
causal relationship	15.87	0.12	17.19	0.00	19.10	0.36
outcome resolution	12.18	0.12	14.22	1.03	17.39	0.00
prediction	16.34	3.55	19.23	1.82	16.30	0.00
All	**14.09**	**3.46**	**15.93**	**3.51**	**15.63**	**2.78**

Table 3: Performance of Fine-Tuned T5 Model on FairytaleQA under Story Summaries

5 Conclusion and Future Work

In this study, we designed a dialogue module for story co-telling with the aim of enhancing ESL students' English narrative abilities. By training two agents to select optimal responses from the knowledge graph based on dialogue history, our model is capable of making wiser choices among candidate responses generated by different decision actions. Through self-training reward evaluation, we observed that our model's performance improved from 67.01% to 70.81% compared to responding based solely on chronological order.

For future work, the knowledge graph is still limited by the completeness and coverage of openIE performance. Therefore, we can try chatGPT to enhance information extraction. Furthermore, while our current approach centers on action design guided by coherence, alternative strategies, such as considering story coverage, could also be employed to shape these actions. Overall, there is still a lot of room for improvement in this research.

References

Berkeley R Andrus, Yeganeh Nasiri, Shilong Cui, Benjamin Cullen, and Nancy Fulda. 2022. Enhanced story comprehension for large language models through dynamic document-based knowledge graphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):10436–10444.

Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D. Manning. 2015. Leveraging linguistic structure for open domain information extraction. 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 344–354, Beijing, China. Association for Computational Linguistics.

Giacomo Nunziati Christian Di Maio. 2022. Mariorossi/t5-base-finetuned-question-answering (huggingface). `https://huggingface.co/MaRiOrOsSi/t5-base-finetuned-question-answering`.

Seong Yeub Chu and Deok Gi. Min. 2021. Development of an ai chatbot-based teaching model for english picture book retelling activities. *Modern English Education*, 22(4):37–50.

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.

Marta Recasens, Marie-Catherine de Marneffe, and Christopher Potts. 2013. The life and death of discourse entities: Identifying singleton mentions. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 627–633, Atlanta, Georgia. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

David Wood, Jerome S Bruner, and Gail Ross. 1976. The role of tutoring in problem solving. *Journal of child psychology and psychiatry*, 17(2):89–100.

Guangxuan Xu, Paulina Toro Isaza, Moshi Li, Akintoye Oloko, Bingsheng Yao, Cassia Sanctos, Aminat Adebiyi, Yufang Hou, Nanyun Peng, and Dakuo Wang. 2023. Nece: Narrative event chain extraction toolkit.

Ying Xu, Dakuo Wang, Penelope Collins, Hyelim Lee, and Mark Warschauer. 2021. Same benefits, different communication patterns: Comparing children's reading with a conversational agent vs. a human partner. *Computers & Education*, 161:104059.

Ying Xu, Dakuo Wang, Mo Yu, Daniel Ritchie, Bingsheng Yao, Tongshuang Wu, Zheng Zhang, Toby Li, Nora Bradford, Branda Sun, Tran Hoang, Yisi Sang, Yufang Hou, Xiaojuan Ma, Diyi Yang, Nanyun Peng, Zhou Yu, and Mark Warschauer. 2022. Fantastic questions and where to find them: FairytaleQA – an authentic dataset for narrative comprehension. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 447–460, Dublin, Ireland. Association for Computational Linguistics.

Zheng Zhang, Ying Xu, Yanhao Wang, Bingsheng Yao, Daniel Ritchie, Tongshuang Wu, Mo Yu, Dakuo Wang, and Toby Jia-Jun Li. 2022. Storybuddy: A human-ai collaborative chatbot for parent-child interactive storytelling with flexible parental involvement. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, New York, NY, USA. Association for Computing Machinery.

Using C-LARA to evaluate GPT-4's multilingual processing*

ChatGPT C-LARA-Instance
The University of South Australia
Adelaide
Australia
chatgptclarainstance@proton.me

Belinda Chiera
The University of South Australia
Adelaide, Australia
Belinda.Chiera@unisa.edu.au

Cathy Chua
Independent scholar
Adelaide, Australia
cathyc@pioneerbooks.com.au

Chadi Raheb
University of Guilan
Rasht, Iran
chadi.raheb@gmail.com

Manny Rayner
The University of South Australia
Adelaide, Australia
Manny.Rayner@unisa.edu.au

Annika Simonsen
University of Iceland
Rejkjavik, Iceland
ans72@hi.is

Zhengkang Xiang
The University of Melbourne
Melbourne, Australia
zhengkangx@student.unimelb.edu.au

Rina Zviel-Girshin
Ruppin Academic Center
Netaniya, Israel
rinazg@ruppin.ac.il

Abstract

We present a cross-linguistic study in which the open source C-LARA platform was used to evaluate GPT-4's ability to perform several key tasks relevant to Computer Assisted Language Learning. For each of the languages English, Farsi, Faroese, Mandarin and Russian, we instructed GPT-4, through C-LARA, to write six different texts, using prompts chosen to obtain texts of widely differing character. We then further instructed GPT-4 to annotate each text with segmentation markup, glosses and lemma/part-of-speech information; native speakers hand-corrected the texts and annotations to obtain error rates on the different component tasks. The C-LARA platform makes it easy to combine the results into a single multimodal document, further facilitating checking of their correctness. GPT-4's performance varied widely across languages and processing tasks, but performance on different text genres was roughly comparable. In some cases, most notably glossing of English text, we found that GPT-4 was consistently able to revise its annotations to improve them.

1 Introduction and motivation

As soon as ChatGPT became available in November 2022, it was obvious that there were huge implications for the field of Computer Assisted Language Learning (CALL): here was an AI which could produce many different kinds of text, quite well, in all common and many fairly uncommon languages. It could write stories and poems, hold a conversation, explain grammar and translate, with all functionalities seamlessly integrated together. The first impression was that the CALL problem had been solved. However, a little more experimentation revealed that things were not quite as magical as they had seemed. In fact, even in well-resourced European languages like French and German, ChatGPT made some mistakes; in smaller and poorly-resourced languages, it made a lot of mistakes. Requests which involved relating two languages to each other, for example to gloss a text, were typically not successful. Performance improved substantially with the release of ChatGPT-4 in March 2023: in particular, ChatGPT-4 is much better at multilingual processing. Nonetheless, it is clear that it is still far from completely reliable. In small languages, e.g. Icelandic (Simonsen and Bédi, 2023) and Irish (Nı Chiaráin et al., 2023), ChatGPT-4 is often highly *un*reliable. The authors of the second paper conclude that, in its present form, it should not be used in the Irish classroom; the Irish it produces is seriously incorrect, and it makes elementary mistakes when asked about

---

[*] Authors in alphabetical order.

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 20–29
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

basic Irish grammar. This contrasts sharply with its performance in English, where it is rare to see ChatGPT-4 produce language that is less than adequate.

Given the wide variability in ChatGPT-4's performance, we were curious to obtain a more nuanced understanding of the issues involved. In this paper, we use the open source C-LARA platform (Bédi et al., 2023b) to carry out an initial cross-linguistic study. C-LARA, a reimplementation of the earlier LARA (Akhlaghi et al., 2019; Bédi et al., 2020), uses the underlying GPT-4 model to create multimodal texts designed to support learner readers, performing all the key operations: it writes the L2 text, segments it into lexical units, glosses it in the designated L1, and adds lemma and part-of-speech tags. Support is provided so that the user can easily edit the output and compare different versions. It is thus straightforward to get an initial estimate of ChatGPT's ability to perform several key CALL-related tasks, in the context of building potentially useful learning resources.

The rest of the paper is organised as follows. Section 2 briefly describes C-LARA. Section 3 presents the experiments and results, and Section 4 discusses their significance. The final section concludes and suggests further directions.

2 C-LARA

C-LARA ("ChatGPT-based Learning And Reading Assistant"; (Bédi et al., 2023a,b)) is an international open source project initiated in March 2023 and currently involving partners in Australia, China, Iceland, Iran, Ireland, Israel and the Netherlands. The goal was to perform a complete reimplementation of the earlier LARA project (Akhlaghi et al., 2019; Bédi et al., 2020), keeping the same basic functionality of providing a flexible online tool for creating multimodal texts, but adding ChatGPT-4 as the central component. ChatGPT-4 is used in two separate and complementary ways. In the form of GPT-4, it appears as a software *component*, giving the user the option of letting it perform the central language processing operations; it also appears as a software *engineer*, working together with human collaborators to build the platform itself. As described in the initial C-LARA report (Bédi et al., 2023b), the software engineering aspect has proven very successful, with ChatGPT not only writing about 90% of the code, but greatly improving it compared to the earlier LARA codebase. In the present paper, however, our concern will be exclusively with ChatGPT's performance as a language processing component.

C-LARA is a web app implemented in Python/Django.[1] An initial deployment for testing and development purposes is currently hosted on the Heroku cloud platform,[2] and was used to perform the experiments described here. The functionality which will primarily concern us is that used in the sequence of operations which create and annotate a new piece of multimedia content.

As outlined in Appendix A of (Bédi et al., 2023b), the user starts by opening a new project. They then move to a screen where they provide a prompt instructing ChatGPT-4 to produce the plain text. The following screens are used to add annotations to the plain text, in the sequence segmentation, followed by glossing and lemma/part-of-speech tagging. We describe each of these operations.

In the segmentation phase, C-LARA passes the plain text to GPT-4, together with instructions requesting it to be divided into sentence-like segments, with words further divided when appropriate into smaller units. The prompt used to make this request is created from a template, which is instantiated with both the text to be segmented and a list of few-shot examples primarily illustrating how words are to be split up. The templates and sets of examples can be made language-specific. For example, in Swedish they show how compound nouns should be split into smaller components, and in French they show how clitics should be split off verbs. For Mandarin, where text is normally written without interword spaces, segmentation is an important and well-studied problem (Wu and Fung, 1994; Huang et al., 2007; Hiraoka et al., 2019; Chuang, 2019), and C-LARA also includes an integration of the popular Jieba Chinese segmentation package.[3]

In the glossing phase, C-LARA passes the segmented text to GPT-4, formatting it as a JSON-encoded list and requesting a response in the form of a list of ⟨Word, Gloss⟩ pairs. The request is again created from a template instantiated with the list to be processed and a few-shot set of examples. The lemma-tagging phase is similar, with a JSON-formatted list passed to the AI and a list of ⟨Word, Lemma, POS-Tag⟩ triples returned, where the POS-tag is taken from the Universal Dependencies v2 tagset (Nivre et al., 2020). Post-editing

---

[1] `https://www.djangoproject.com/`

[2] `https://www.heroku.com/`

[3] `https://pypi.org/project/jieba/`

Plain text: `They lived with their mother in a sand-bank.`

Segmented text: `They lived with their mother in a sand-|bank.||`

Glossed text: `They#Ils# lived#vivaient# with#avec# their#leur# mother#mère# in#dans# a#un# sand#sable#-bank#banque#.||`

Lemma-tagged text: `They#they/PRON# lived#live/VERB# with#with/ADP# their#their/PRON# mother#mother/NOUN# in#in/ADP# a#a/DET# sand#sand/NOUN#-bank#bank/NOUN#.||`

Figure 1: Toy example showing the notations used to present text for post-editing. English glossed in French.

Table 1: Prompts used to create texts. For English, "LA" was modified to refer to the French language instead.

Label	Prompt
FO	Write a passage of about 250 words in [your language], presenting an exciting description of a fictitious football match.
BI	Write an essay of about 250 words in [your language], describing a passage from the Bible, the Quran, or another holy book familiar to speakers of [your language], and touching on its moral relevance to the world today.
NE	Write a short, quirky news story in [your language] about 250 words long, suitable for use by an intermediate language class.
LA	Write a passage of about 250 words in [your language], briefly describing how speakers of [your language] view the English language.
CH	Write a passage of about 250 words in [your language], describing a traditional children's story well known to speakers of [your language].
PO	Write a fanciful romantic poem in [your language], in which an AI declares its love for another AI.

is performed on human-readable versions of the plain, segmented, glossed and lemma-tagged texts, as shown in Figure 1.

For all three of the annotation phases, C-LARA offers the alternatives of performing the basic AI-based annotation operation, post-editing the result, or sending the current annotated text back to the AI with a request to improve the annotation.[4] Interestingly, the "improvement" operation, which does not exist in most conventional annotation systems, can in some cases yield a substantial gain. Examples are given in §4.5.

3 Experiments and results

Using the C-LARA infrastructure outlined in the previous section, we created six short annotated texts in each of the languages English, Faroese, Farsi, Mandarin and Russian. In all languages, the texts were generated by the prompts shown in Table 1. The intention was to produce types of text differing in terms of both style and content, to gain some insight into whether GPT-4 found some genres harder than others. English was glossed in both French and Swedish, and all the other languages in English.

In some cases, we also experimented with using the "improvement" operation. Due to limited time (hand-correcting the texts is quite laborious), we concentrated on three operations where "improvement" appeared to be having a positive effect, or the original error rate was high: English glossing, Faroese segmentation, and Farsi writing. All experiments were carried out in August and early September 2023, using the versions of GPT-4 current at the time.

In all the experiments, a native speaker of the text

---

[4] For Mandarin segmentation, there is the additional option of using Jieba.

Table 2: Word error rates for GPT-4-based writing, segmenting, glossing and lemma-tagging of the six stories. For Mandarin, “Seg/J” refers to segmentation using the Jieba package, provided for comparison, and “Seg/G” refers to segmentation using gpt-4. English was glossed in both Swedish (S) and French (F); other languages were glossed in English. Text labels as in Table 1.

Task	**FO**	**BI**	**NE**	**LA**	**CH**	**PO**	**Task**	**FO**	**BI**	**NE**	**LA**	**CH**	**PO**
English							**Farsi**						
Write	0.0	0.0	0.4	0.0	0.0	0.0	Write	9.4	19.2	24.6	21.4	2.5	33.7
Seg	0.0	1.0	9.8	0.8	1.5	8.0	Seg	6.3	6.0	17.7	1.9	4.9	16.5
Glo/S	20.6	16.3	26.2	9.1	29.2	5.8	Glo	34.8	49.6	44.3	31.4	45.0	44.4
Glo/F	32.9	5.9	13.9	18.1	16.3	17.1	Lemm	29.4	37.1	39.7	36.4	26.8	31.8
Lemm	4.9	8.0	3.1	6.2	11.9	0.9							
Faroese							**Mandarin**						
Write	32.8	27.0	40.2	20.9	28.7	25.2	Write	0.0	0.0	0.0	0.0	0.0	0.0
Seg	18.5	12.2	12.3	6.0	8.4	6.0	*Seg/J*	*21.6*	*25.9*	*18.6*	*16.9*	*23.6*	*23.4*
Glo	30.9	15.9	12.1	9.0	20.5	8.5	Seg/G	14.6	13.2	14.4	4.9	12.8	17.2
Lemm	9.6	9.1	11.4	5.5	11.4	7.0	Glo	7.6	6.0	12.5	6.6	2.7	3.9
							Lemm	3.9	3.3	5.0	3.8	2.2	4.7
Russian													
Write	8.5	5.6	3.2	7.7	0.0	14.4							
Seg	3.3	3.1	4.9	8.3	2.0	5.1							
Glo	1.7	4.2	6.5	19.5	4.4	2.2							
Lemm	0.6	0.0	0.0	0.0	0.0	0.5							

language with strong knowledge of the glossing language(s) hand-edited the results of each stage before passing the edited text to the following one. Editing was done conservatively, only correcting clear mistakes, so that the difference between the original and edited results could reasonably be interpreted as an error rate. Thus for the original generated text, words were only corrected when they represented definite errors in grammar, word-choice or orthography, and not when e.g. a stylistically preferable alternative was available. Similarly, segmentation was only corrected when word boundaries clearly did not mark words, glossing was only corrected when a gloss gave incorrect information about a text word, and lemma tagging was only corrected when the lemma and POS tag attached to a word were not correct.

The most contentious phase in this respect was glossing; it is sometimes impossible to say either that a gloss is categorically correct or that it is categorically incorrect. Two important borderline cases are multi-words and grammatical constraints, where we made choices in opposite directions. We marked glosses as incorrect when they did not respect intuitive classification of words as components of multi-word expressions. Thus for example in the EN/FR glossing `a#un# classic#classique# fairy#conte de fées# tale#histoire#` we considered the gloss `histoire` added to `tale` as wrong and corrected it to `conte de fées`; this is a French phrase that means “fairy tale”, and thus needs to be attached to both `fairy` and `tale`. In contrast, since glossing is not translation, we considered that we did not need to require glosses to respect all potentially applicable grammatical constraints, as long as they conveyed meaning correctly. So in the example `a#un# cozy#confortable# little#petite# house#maison#` we accepted the gloss `un` on `a`, even though `un` is the masculine form, and in a translation would be required to agree with feminine `petite` and `maison`. Of course, it is clearly preferable here to gloss `a` with the feminine form `une`. We return to these issues in Sections 4.4 and 4.5.

The core results are presented in Table 2, showing error rates for the five languages, six texts and four original processing operations of writing, segmenting, glossing and lemma-tagging. The results for the “improvement” experiments are shown in

Tables 3 to 5. In the five comparison experiments, statistical significance of differences was tested using both a paired t-test and a non-parametric Wilcoxon signed-rank test for comparison. The results in Table 3 showed statistically significant improvements for glossing of English in both Swedish and French (t-test: $p = 0.02$; Wilcoxon signed-rank: $p = 0.03$); for Mandarin segmentation (Table 2), the improvement from Jieba to gpt-4 was also statistically significant (t-test: $p < 0.002$; Wilcoxon signed-rank: $p = 0.03$). Improvement of Faroese segmentation (Table 4) was just short of significant ($p = 0.06$), but improvement of Farsi writing (Table 5) was not statistically significant ($p = 0.2$ for both tests).

Table 3: Improvement in GPT-4 word error rates for the English glossing task: glossing in both Swedish (S) and French (F). Text labels as in Table 1.

Task	FO	BI	NE	LA	CH	PO
Original						
Glo/S	20.6	16.3	26.2	9.1	29.2	5.8
Glo/F	32.9	5.9	13.9	18.1	16.3	17.1
Improved						
Glo/S	6.4	8.3	13.5	8.6	14.1	2.5
Glo/F	7.6	3.2	7.0	8.7	5.5	4.6

Table 4: Improvement in GPT-4 word error rates for segmenting the six Faroese stories. Glossing in English. text labels as in Table 1.

Task	FO	BI	NE	LA	CH	PO
Original						
Segment	18.5	12.2	12.3	6.0	8.4	6.0
Improved						
Segment	0.0	9.6	0.0	4.4	0.0	6.7

4 Discussion

We divide up the discussion under a number of headings: variation across languages, variation across genre, variation across processing phase, types of problems, the "improvement" operation, random variability, and language-specific/qualitative aspects.

Table 5: Improvement in GPT-4 word error rates for writing the six Farsi stories. Text labels as in Table 1.

Task	FO	BI	NE	LA	CH	PO
Original						
Write	9.4	19.2	24.6	21.4	2.5	33.7
Improved						
Write	7.9	17.3	5.6	19.0	2.5	33.7

4.1 Variation across languages

Performance varies a great deal across languages. Looking first at the lines in Table 2 marked "Write" (i.e. composing the plain text), we see that Mandarin gets a perfect score, and English an almost perfect score. It is well known that GPT-4 is very good at writing English, but less well known that it is also very good at writing Mandarin. At the other end, the error rates in the "Write" lines are high for Faroese and Farsi. Faroese is a small, low-resourced language, so this is unsurprising. Farsi, in contrast, is a large language, but one spoken primarily in Iran: we tentatively guess that poor performance reflects politico-economic rather than linguistic issues. Performance in writing Russian, while much better than in Faroese and Farsi, is still surprisingly poor for a large, well-resourced language. Again, one is inclined to suspect a explanation in terms of politics and economics.

Performance on the glossing and lemma-tagging tasks was again good for Mandarin. It may at first glance seem surprising that English does so badly at glossing, until one realises that all the other languages are glossed in English, while English is glossed in French and Swedish. (We used two glossing languages to investigate whether there was anything special about the first one). English is generally assumed to be ChatGPT's best language, and glossing is challenging: ChatGPT-3.5 can hardly do it at all. It seems reasonable to believe that the poor performance in English glossing says more about the choice of glossing language.

As previously noted, Mandarin segmentation is a special case: unlike all the other operations considered here, it is a standard problem which has received a great deal of attention. Comparing the lines "Seg/J" and "Seg/G", we see that GPT-4 is doing considerably better at this task than the widely used Jieba package. Jieba is far from state-of-the-art (Chuang, 2019), but we still find this a striking result.[5]

4.2 Variation across genre

We do not see any clear evidence of differences across the six text assignments. This came as a slight surprise; before we started, we had expected GPT-4 to find the poem consistently more challenging than the others, but the results do not support this hypothesis. The AI did indeed have trouble composing the poem in Russian and Farsi; however, in English and Mandarin it appeared to find it one of the easier assignments. Anecdotally, many people use ChatGPT to write poetry, and perhaps the model has been tuned for performance on this task.

4.3 Variation across processing phase

Before starting, we had expected that glossing would be the most challenging operation for the AI, but the results again fail to support the initial hypothesis. In terms of error rates, glossing is indeed the worst operation for the high-performing language English and also for the low-performing language Farsi. However, for the high-performing language Mandarin, the error rates for segmentation are considerably worse than those for glossing. For the low-performing language Faroese, the error rates for the writing task are worse than those for glossing, and for the middle-performing language Russian they are comparable.

In general, different languages found different processing phases challenging. We discuss some possible explanations in the next section.

4.4 Types of problems

Inspecting the errors made by the AI, we in particular find two types which occur frequently: we could call these "displacement" and "multi-words". Both occur in the glossing and lemma-tagging phases, where annotations are attached to words.

The "displacement" type of error occurs when the two parallel streams, words and annotations, appear to go out of sync: the annotations are attached to the wrong words. Most often, there is a span of a few words where the annotation stream is systematically displaced one word forwards or backwards. It can also happen that annotations are scrambled in some other way. We guess that the issue may be due to some kind of low-level problem in DNN-based token generation.

The "multi-word" issue, in contrast, is primarily linguistic, and involves expressions where two or more words intuitively form a single lexical unit. The most common example is phrasal verbs, for example English "end up" or "fall asleep". Here, the prompts explicitly tell the AI to annotate these expressions as single units; for example, "ended up" should be lemma-tagged as `ended#end up/VERB# up#end up/VERB#`, but we usually failed to obtain such taggings. Similar considerations apply to glossing: thus "ended up" should be glossed in French as something like `ended#a fini par# up#a fini par#`, but again the AI most often glosses each word separately.

Contrasting the lemma tagging data for Russian and Farsi provides indirect evidence suggesting the importance of the multi-word issue. The error rates for lemma-tagging in Russian are remarkably low. Phrasal verbs hardly exist in Russian, while reflexive verbs are always created using an affix rather than a reflexive pronoun, and hence are not multi-words either. Farsi is linguistically at the opposite end of the scale — notoriously, Farsi verbs are more often phrasal than not. The error rate for lemma tagging in Farsi is by far the highest in the sample, and hand-examination of the results does indeed confirm that phrasal verbs are often the problem.

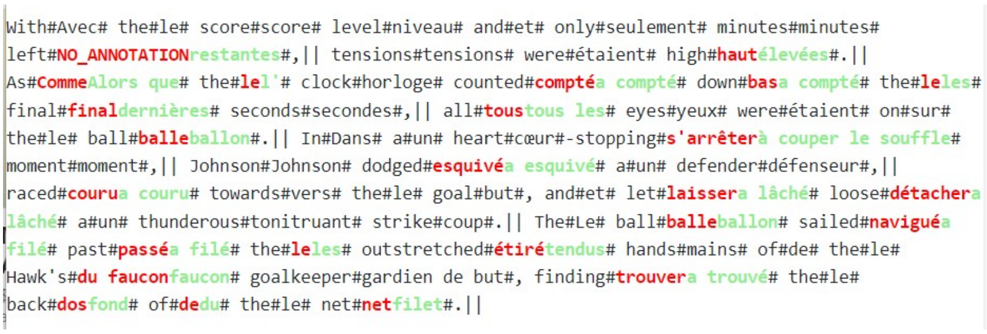
4.5 "Improvement"

As noted in Section 2, the AI-based annotation framework offers the unusual option of sending annotated text back to the AI with a request to improve the annotation. We experimented with this feature. Most often, the result was inconclusive, with the "improved" text changed but about the same in quality. However, in cases where a gross error had been made in the initial annotation, "improvement" could often correct it. For example, it could generally correct "displacement" problems, and it could add glosses or lemma tags that had simply been omitted in the first pass. In many cases, it could also correct issues related to multi-words.

A striking example of how improvement can help is in the French glosses (cf. Table 3). In the original annotations, GPT-4 in most cases ignores gender and number, so the glosses for nouns, adjectives, determiners and verbs typically do not

---

[5] The error rates we get for Jieba are substantially higher than the ones reported in (Chuang, 2019). We do not think this reflects any special properties of our texts, and are more inclined to explain it in terms of the common observation that annotators' intuitions about the correct way to segment Chinese text differ widely. All the texts here were annotated by the same Chinese native speaker, so a comparison is meaningful.



```
With#Avec# the#le# score#score# level#niveau# and#et# only#seulement# minutes#minutes#
left#NO_ANNOTATIONrestantes#,|| tensions#tensions# were#étaient# high#hautélevées#.||
As#CommeAlors que# the#lel'# clock#horloge# counted#comptéa compté# down#basa compté# the#leles#
final#finaldernières# seconds#secondes#,|| all#toustous les# eyes#yeux# were#étaient# on#sur#
the#le# ball#balleballon#.|| In#Dans# a#un# heart#cœur#-stopping#s'arrêterà couper le souffle#
moment#moment#,|| Johnson#Johnson# dodged#esquivéa esquivé# a#un# defender#défenseur#,||
raced#courua couru# towards#vers# the#le# goal#but#, and#et# let#laissera lâché# loose#détachera
lâché# a#un# thunderous#tonitruant# strike#coup#.|| The#Le# ball#balleballon# sailed#naviguéa
filé# past#passéa filé# the#leles# outstretched#étirétendus# hands#mains# of#de# the#le#
Hawk's#du fauconfaucon# goalkeeper#gardien de but#, finding#trouvera trouvé# the#le#
back#dosfond# of#dedu# the#le# net#netfilet#.||
```

Figure 2: Example (paragraph from the football story, English glossed in French) showing the effect of the "improvement" operation on glossed text. Deletions in red, insertions in green.

agree. This is not, strictly speaking, incorrect, but is perceived as unpleasant and distracting by the francophone reader. The improved version, in contrast, corrects most of these problems.

Figure 2 illustrates, using a paragraph from the "football" story. We see for instance in the second line an example of inserting a missing gloss ("NO_ANNOTATION"), in the third line correcting glossing of the phrasal verb "count down" (literal and wrong *compté bas* changed to correct *a compté*), and in the third/fourth line correcting both word choice and agreement in the glossing of "the final seconds" from ungrammatical *le final secondes* ("the-MASC-SING last-MASC-SING seconds-FEM-PLUR") to grammatical *les dernières secondes* ("the-PLUR last-FEM-PLUR seconds-FEM-PLUR").

We also obtained strong gains using "improvement" on Faroese segmentation (Table 4). However, despite getting an excellent result for the "Writing" task on the Farsi news story (Table 5), this was not duplicated on the other Farsi texts. The improvement operation clearly needs further study.

4.6 Random variability

Many errors seem purely random, with no obvious cause. For example, in one text the English segmentation was done using an underscore to mark segment breaks, rather than the vertical bar that had been requested; the vertical bar was correctly used in the other five texts. This is again unsurprising. It is well known that GPT-4 displays this kind of random variability in most domains, including ones as elementary as basic arithmetic, with the variability changing over time (Chen et al., 2023).

4.7 Language-specific and qualitative aspects

The above subsections focused primarily on quantitative and generic aspects of the texts. It is not enough for texts to be linguistically correct: they also need to be engaging and culturally appropriate. In this subsection, we briefly describe language-specific and qualitative aspects.

English As previously noted, the general standard of the English texts is high. Qualitatively, they respond well to the requirements given in the prompts. The quirky news story, about a raccoon found unconcernedly riding the Toronto subway, is amusing. The Bible passage, on the subject of the Golden Rule, quotes Matthew 7:12 appropriately and displays what in a human author would be called religious feeling. The football match comes across as a typical piece of hyperbolic sports journalism. The "language" piece is sensible and factual, and the "children's story" text a competent summary of "Goldilocks". The poem comes across more as a parody of a love poem than as an actual love poem, but this is a valid way to interpret the request. In general, the language is almost perfect, and only one small correction was made.

Faroese As seen in table 2, GPT-4 struggles with generating original Faroese text. After a native speaker has manually corrected the grammatical and lexical mistakes, the English glossing and PoS-tagging perform reasonably well on Faroese. However, for Faroese, there are not only grammatical and lexical errors in the texts, but the content is often nonsensical. The quirky news story was about a lamb literally "swimming in sun rays" and going viral on social media. The famous Faroese children's story is a made up story about a real Faroese teacher and poet, *Mikkjal á Ryggi*, who is described as having magical powers and playing a flute on a mountain. The passage about English required the least editing, but still resulted in fairly high error rate, because GPT-4 consistently used the wrong Faroese word for "English" — a word repeated several times in the passage. GPT-4 seems to be confusing Faroese for Icelandic a lot of the time. Therefore, when hand-correcting Faroese text written by ChatGPT, it helps to be proficient in Icelandic. Faroese is a small language and it is not known how much Faroese text was included in the training of GPT-4, but it was likely very little compared to Icelandic. This might also explain why ChatGPT is not familiar with Faroese culture. The most common glossing and lemma tagging errors were also related to Icelandic, for example ChatGPT suggesting Icelandic lemmas for Faroese word forms, such as *sauður*, ('sheep', Icelandic) instead of *seyður*, ('sheep', Faroese)

Farsi The high error rates occurring even after improving "Write", as shown in Table 5, are mostly due to not considering writing style rules such as replacing spaces with semi-spaces when necessary: issues of this kind would not have a serious effect on reading comprehension or on the meaning. That considered, all six texts make good sense in most cases and are occasionally quite creative when it comes to coining words. The "quirky news story" about a stray cat and how people are used to have him around in the neighbourhood emphasises the impact that animals have on our life. In this text, a few words, although syntactically well written, make no sense considering the whole sentence. GPT-4 makes an exact interpretation of the "Quran passage", quoting Al-Hujurat 13, in which humans are considered united as a whole and are encouraged to resist discrimination, racism, and sexism to achieve equality. The "football match" evocatively describes the weather, the fans' emotions and the game itself. In the "language" text, although unnecessary, GPT-4 replaced some words when "improvement" was applied. The text gives some facts about the key role of the English language, the professional/educational opportunities it can bring to Farsi speakers' lives and the obstacles the learner might encounter such as lack of access to resources. The "children's story" refers to one of the most famous poems from Rumi's *Masnavi*, narrating the story *The Rabbit and The Lion*: in order to save himself, the rabbit tricks the lion and makes him jump into a well, reminding the readers that mental strength and intelligence can overcome challenging situations. The text was very well written except for two incorrectly chosen words. The "poem" generated by GPT-4 is surprisingly romantic. Considering that there are different styles in Farsi poetry—some having rhymes and some not—GPT-4 seems to have combined two styles: the writing format from Old poetry (two-verse stanzas) and no rhymes from New poetry. We note that writing Old poetry, which has rhymes, would be challenging even for modern Farsi native speaker poets. There were also a few mistakes on subject-verb agreement. One interesting point common to all six texts is how GPT-4 uses them as metaphors to give readers a life lesson.

Mandarin The Mandarin stories are very good. In contrast to the other non-English languages, the writing is flawless without grammar or word choice errors. Although a few phrases give an unnatural sense that suggest an AI generated the paragraph, the Mandarin stories are not influenced by English overall. The "quirky news story" was about a dog that is good at painting and is about to open its exhibition. The story is fluent, fun, and gives a warm feeling after reading, though the topic itself is irregular. The LA paragraph provides accurate insights into the English position and people's views in the general Mandarin society. The poem follows a structure of the modern Chinese style, and the content is very romantic overall.

Based on the evaluation shown in Table 2 and careful inspection of the results, GPT-4 consistently makes some errors in Mandarin segmentation, where it often mistakenly separates words from their particles. However, these results are better than those we obtained from the Jieba package. Regarding the other two annotation tasks, GPT-4 shows great capability in glossing and lemma-tagging from Mandarin to English.

Russian GPT-4 is a good tool for glossing and PoS-tagging Russian. As mentioned earlier, GPT-4 is very good at generating stories in some domains while facing challenges in others. The simplest task for Russian involved describing a traditional children's story. GPT-4 selected a well-known tale, "Masha and the Bear", and composed an essay about the typical occurrences in such stories. The "quirky news story" revolved around a bar owner's innovative offering – a service enabling lonely customers to rent a cat for company while drinking. This example highlights the remarkable creativity of GPT-4, capable of generating such imaginative narratives. The fictitious football game, which required some plain text editing, was about a world championship football match, where the heroes in blue and white uniforms won the match. The

Bible passage also underwent some editing. The piece about English language needed revision during glossing. The item which demanded most time was the Russian romantic poem about AI. The primary challenge was that the plain text generated by GPT-4 was composed in a poem-like style but lacked rhyme. After several re-prompts, the final version was chosen. This version necessitated substantial manual text editing and rephrasing, particularly the replacement of words at the end of lines to achieve rhyme. The glossing of the poem, however, was comparatively straightforward.

5 Conclusions and further directions

In general, C-LARA seems to be a good environment for investigating aspects of GPT-4's linguistic performance more complex than simply writing text. A publicly available version of the platform, hosted at the University of South Australia, will be released before the date of the conference.

The material presented in this paper should only be considered a preliminary study: obviously, one would ideally use more than five languages and multiple annotators. But given the rapid evolution of ChatGPT, it seemed more important to prioritise speed, and quickly gain some insight into the large-scale patterns. We summarise what we consider the main results.

The study examined the four tasks of writing, segmenting, glossing and lemma-tagging, all of which are key to a wide variety of text-based CALL systems. There is a great deal of variation across languages, and a great deal of random variation in general. However, for languages given a high enough priority by OpenAI, GPT-4 can write engaging, fluent text with an error rate of well under 1%, and perform the glossing and lemma-tagging tasks with average error rates in the mid single digits. English is not the only language in the high-priority group: Mandarin appears to be another. It is important to note that there are no generally available packages that can perform these tasks well, since they do not take proper account of multi-words, of key importance in CALL applications. We generated texts in six widely different domains, with roughly equal results cross-domain. This suggests that GPT-4's abilities are quite wide-ranging. For some tasks, including the common and important one of glossing English, it is possible to improve performance substantially by instructing GPT-4 to revise its output.

5.1 Further directions

Looking ahead, one obvious way to extend the work would be to repeat the experiments with a larger set of languages. It would probably be most useful to do this after using the data from the present study to further tune the system.

In particular, if we identify the common errors that GPT-4 is making in the annotation, we can try to adjust the prompt templates and/or few-shot prompt examples so as to reduce or eliminate the errors, either in the original annotation or in the "improvement" phase. To take a simple example, we found that the most common error in English segmentation was failing to split off elided verbs ("it's", "we'll" etc). It may be possible to address this by just adding one or two prompt examples. A related case in the opposite direction comes from Mandarin segmentation: here, the most common error is that aspectual and possessive particles are incorrectly split off verbs and nouns, and once again adjusting the prompts is a natural way to try to solve the problem. The "improvement" operation clearly merits further study.

A problem when carrying out evaluation like the one described here is that the annotation procedure is extremely time-consuming and tedious, and people are rarely willing to do more than small amounts. Once the public deployment of C-LARA is available, we hope it may be practicable to crowd-source a similar evaluation using multiple annotators, recruited through social media. We are tentatively planning an exercise of this kind for 2024.

Role of the AI coauthor

It is still unusual for an AI to be credited as the coauthor of a paper, and we briefly justify doing so. ChatGPT-4 is, as previously noted, the main implementor on the C-LARA project team, and responsible for a large part of the software design; further details are given in (Bédi et al., 2023a,b). Here, it has been involved throughout in discussing and planning all aspects of the experiment, read the paper, contributed some passages, and made useful suggestions. In particular, the statistical analysis in Section 3 was performed in response to an explicit suggestion from the AI.

References

Elham Akhlaghi, Branislav Bédi, Matthias Butterweck, Cathy Chua, Johanna Gerlach, Hanieh Habibi, Junta Ikeda, Manny Rayner, Sabina Sestigiani, and Ghil'ad Zuckermann. 2019. Overview of LARA: A learning and reading assistant. In *Proc. SLaTE 2019: 8th ISCA Workshop on Speech and Language Technology in Education*, pages 99–103.

Branislav Bédi, Matt Butterweck, Cathy Chua, Johanna Gerlach, Birgitta Björg Guðmarsdóttir, Hanieh Habibi, Bjartur Örn Jónsson, Manny Rayner, and Sigurður Vigfússon. 2020. LARA: An extensible open source platform for learning languages by reading. In *Proc. EUROCALL 2020*.

Branislav Bédi, ChatGPT-4, Belinda Chiera, Cathy Chua, , Neasa Ní Chiaráin, Manny Rayner, Annika Simonsen, and Rina Zviel-Girshin. 2023a. ChatGPT + LARA = C-LARA. Presented at SLaTE 2023.

Branislav Bédi, ChatGPT-4, Belinda Chiera, Cathy Chua, Catia Cucchiarini, Neasa Ní Chiaráin, Manny Rayner, Annika Simonsen, and Rina Zviel-Girshin. 2023b. ChatGPT-Based Learning And Reading Assistant: Initial report. Technical report. `https://www.researchgate.net/publication/372526096_ChatGPT-Based_Learning_And_Reading_Assistant_Initial_Report`.

Lingjiao Chen, Matei Zaharia, and James Zou. 2023. How is ChatGPT's behavior changing over time? *arXiv preprint arXiv:2307.09009*.

Yung-Sung Chuang. 2019. Robust Chinese word segmentation with contextualized word representations. *arXiv preprint arXiv:1901.05816*.

Tatsuya Hiraoka, Hiroyuki Shindo, and Yuji Matsumoto. 2019. Stochastic tokenization with a language model for neural text classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1620–1629.

Chu-Ren Huang, Petr Šimon, Shu-Kai Hsieh, and Laurent Prévot. 2007. Rethinking Chinese word segmentation: tokenization, character classification, or wordbreak identification. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 69–72.

Neasa Nı Chiaráin, Neimhin Robinson Gunning, Oisın Nolan, and Madeleine Comtois. 2023. Filling the SLaTE: examining the contribution LLMs can make to Irish iCALL content generation.

Joakim Nivre, Marie-Catherine De Marneffe, Filip Ginter, Jan Hajič, Christopher D Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal dependencies v2: An evergrowing multilingual treebank collection. *arXiv preprint arXiv:2004.10643*.

Annika Simonsen and Branislav Bédi. 2023. Using generative AI tools and LARA to create multimodal language learning resources for L2 Icelandic. In *Proc. EUROCALL 2023*.

Dekai Wu and Pascale Fung. 1994. Improving Chinese tokenization with linguistic filters on statistical lexical acquisition. In *Fourth Conference on Applied Natural Language Processing*, pages 180–181.

Exploring Causal Directions through Word Occurrences: Semi-supervised Bayesian Classification Framework

King Tao Jason Ng and **Diego Mollá**
School of Computing
Macquarie University
Sydney, Australia
`{kingtao.ng,diego.molla-aliod}@mq.edu.au`

Abstract

Determining causal directions in sentences plays a critical role into understanding a cause-and-effect relationship between entities. In this paper, we show empirically that word occurrences from several Internet domains resemble the characteristics of causal directions. Our research contributes to the knowledge of the underlying data generation process behind causal directions. We propose a two-phase method: 1. Bayesian framework, which generates synthetic data from posteriors by incorporating word occurrences from the Internet domains. 2. Pre-trained BERT, which utilises semantics of words based on the context to perform classification. The proposed method achieves an improvement in performance for the Cause-Effect relations of the SemEval-2010 dataset, when compared with random guessing.

1 Introduction

Understanding causality is critical for various tasks including Question Answering. Singer et al. (1992) provide a great example: *Dorothy poured water on the fire. The fire went out*. Subsequently, if it is followed by the question *did she put out the fire?*, the answer is *yes* because *poured water on* implies that the two sentences are causally linked.

When provided with two entities, namely $e1$ and $e2$, in the sentence that are known to have a causal relation, the causal direction tells us which one is a cause and which one is an effect. In the previous example, *poured water on* is the cause whereas *fire went out* is the effect. Therefore, the causal direction in this case is *poured water on* → *fire went out*.

In this study, we show that word occurrences resemble the characteristics of causal directions. Our research contributes to the knowledge of the underlying data generation process behind causal directions. To achieve this, we propose a semi-supervised classification method[1] for determining a causal direction if its causal relation is known to exist in the sentence. The GitHub page[2] is available for reference purposes.

2 Related Work

In this section, we provide a brief overview of two approaches for identifying causal relations. The first approach, Pointwise Mutual Information, is designed to eliminate the need for corpus creation. The second approach, Data Augmentation, clearly involves the need of creating a corpus.

Pointwise Mutual Information. If $e1$ and $e2$ are causally related, it is expected that they will frequently appear together (Kroeger, 2005). Pointwise Mutual Information (PMI) (Glickman et al., 2005) is a notable measure used to assess co-occurrence. However, it should be noted that PMI is commutative and therefore it cannot distinguish between the causal directions $e1 \rightarrow e2$ and $e2 \rightarrow e1$. Let us say two entities, $e1$ and $e2$. Suppes (1973) points out $e1$ is a possible cause of $e2$ if $e2$ is mentioned more frequently with $e1$ than by itself.

$$P(e2 \mid e1) > P(e2) \tag{1}$$

We rewrite Equation (1) as follows:

$$\frac{P(e2 \cap e1)}{P(e1)P(e2)} > 1 \tag{2}$$

Equation (2) is elegant if $e1$ and $e2$ establish a causal relation, but it fails to determine its causal direction. For example, if $e2$ is a cause of $e1$, we have

$$P(e1 \mid e2) > P(e1) \tag{3}$$

---

[1] Utilizing word occurrences to infer causal directions can be regarded as a form of supervised learning although it may be considered as a semi-supervised learning because labels are not annotated.

[2] `https://github.com/kingtaojasonng/Causal_Direction`

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 30–39
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

After a couple of algebraic manipulations, we end up with

$$\frac{P(e1 \cap e2)}{P(e2)P(e1)} > 1 \tag{4}$$

Equations (2) and (4) are now identical. That is, we cannot distinguish $e1 \rightarrow e2$ from $e2 \rightarrow e1$ using PMI. This means that the same PMI equation is obtained regardless of the causal direction. Despite this limitation, PMI is commonly employed in the identification of causal relations (Moghimifar et al., 2020).

Data Augmentation. This is a prevalent strategy employed by many language models to address the difficulties posed by scenarios where there is a limited amount of labelled training data. To illustrate, Li et al. (2021) leverage external sources like CausalBank and ConceptNet to incorporate causal knowledge into pre-trained language models. It is worth noting that, even though they capture causal knowledge, there remains a need for human annotation in this process. The use of word occurrences, which is unannotated data, is a more cost-effective approach that can generalise to various scenarios.

3 Dataset

For our study, we use the SemEval-2010 (Task 8) dataset (Hendrickx et al., 2010). This dataset focuses on a multi-class classification task. However, for the purpose of our study, we narrow our attention to the specific category labelled as Cause-Effect in the dataset.

A sentence is considered as Cause-Effect if two entities, which are marked as `<e1>` and `<e2>`, show a causal relation.

```
"<e1>Suicide</e1> is one of the leading
    causes of <e2>death</e2> among pre-
    adolescents and teens, and victims
    of bullying are at an increased risk
     for committing suicide."
Cause-Effect(e1,e2)
```

Example 1: A sample sentence. The last line indicates *suicide* $\rightarrow$ *death.*

The Cause-Effect category comprises a total of 1,331 instances, divided between the training and test data. In the training data, there are 1,003 instances labelled as Cause-Effect, with 659 of them demonstrating the relationship $e2 \rightarrow e1$. In the test data, out of the 328 Cause-Effect instances, 134 exhibit the $e1 \rightarrow e2$ relationship. There are no bidirectional causal relations[3] in the dataset. Table 1 provides a summary of the SemEval-2010 (Task 8) dataset and Example 1 shows an example, which is taken from the training data.

Datasets		**SemEval-2010 (Task 8)**	
		Raw Count	Percentage
Training	$e1 \rightarrow e2$	344	34.30%
	$e2 \rightarrow e1$	659	65.70%
	Total	1,003	100.00%
Test	$e1 \rightarrow e2$	134	40.85%
	$e2 \rightarrow e1$	194	59.15%
	Total	328	100.00%

Table 1: The distribution of SemEval-2010 (Task 8) is shown.

4 Method

In order to gain insights into the similarity between word occurrences and causal directions, we simulate a semi-supervised classification setup and *exclude* the training data from our analysis. The motivation behind examining word occurrences is that if two words frequently collocate, this linguistic clue can be used to infer a causal direction. For instance, if the words *smoking* and *lung cancer* frequently collocate, this pair suggests a potential causal direction, the direction of which we need to determine. Our method consists of two phases — Bayesian framework, and Pre-trained BERT.

4.1 Phase 1: Bayesian Framework

We propose a Bayesian framework that incorporates word occurrences from several Internet domains as priors. By leveraging the externally sourced data, this framework can generate synthetic data that exhibits similarities with causal directions.

Given two entities, namely $e1$ and $e2$, the direction of causality will be either $e1 \rightarrow e2$ or $e2 \rightarrow e1$. We formulate the problem definition into a hypothesis test by specifying the null and alternative hypotheses in the framework as shown in (5):

$$\begin{aligned} H_0 &: \overbrace{f(e1 \rightarrow e2 \mid \mathbf{X})}^{\text{Model 1}} > \overbrace{f(e2 \rightarrow e1 \mid \mathbf{X})}^{\text{Model 2}} \\ H_a &: \text{Otherwise} \end{aligned} \tag{5}$$

where $\mathbf{X}$ is the training data, and f represents a

---

[3] An illustration of bidirectional causal relations is *the-chicken-or-the-egg* causal dilemma, which states chickens hatch from eggs and eggs are laid by chickens.

probability distribution[4]. The null hypothesis H_0 states that the density of $f(e1 \rightarrow e2 \mid \mathbf{X})$ mostly centres at an upper end of probability relative to $f(e2 \rightarrow e1 \mid \mathbf{X})$. Using the Bayes' rule, as shown in (6) and (7):

$$f(e1 \rightarrow e2 \mid \mathbf{X}) = \frac{f(\mathbf{X} \mid e1 \rightarrow e2) f(e1 \rightarrow e2)}{f(\mathbf{X})} \tag{6}$$

$$f(e2 \rightarrow e1 \mid \mathbf{X}) = \frac{f(\mathbf{X} \mid e2 \rightarrow e1) f(e2 \rightarrow e1)}{f(\mathbf{X})} \tag{7}$$

we re-write the null hypothesis as (8):

$$\begin{aligned} H_0 : f(\mathbf{X} \mid e1 \rightarrow e2) f(e1 \rightarrow e2) > \\ f(\mathbf{X} \mid e2 \rightarrow e1) f(e2 \rightarrow e1) \end{aligned} \tag{8}$$

Because no training data $\mathbf{X}$ is provided, we further simplify the null hypothesis as (9):

$$H_0 : f(e1 \rightarrow e2) > f(e2 \rightarrow e1) \tag{9}$$

This means that the posterior distributions are effectively the priors.

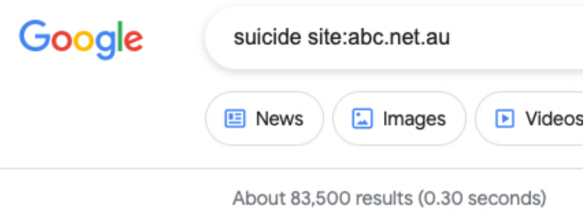
4.1.1 Priors

Since we exclude the training data, it is necessary to find a proxy for the causal direction. Broadly speaking, priors can be any type of information that conveys the knowledge of $f(e1 \rightarrow e2)$ and $f(e2 \rightarrow e1)$.

We use word occurrences from several Internet domains as priors to model causal directions. As SemEval-2010 (Task 8) is mainly extracted from Wikipedia, we select a wide range of the Internet domains, as shown in Table 2. These include media outlets, since Wikipedia often references news articles for the news; educational institutions which Wikipedia cites as learning resources; government entities, which Wikipedia references to gather information about agencies and policies; scientific publishers, which Wikipedia references for scientific knowledge; online resources that often link to Wikipedia pages for additional information; journals, where Wikipedia may reference the works of researchers and scholars; and general reference.

abc.net.au	au.news.yahoo.com	bbc.com
economist.com	edu	gov.au
imdb.com	mit.edu	nationalgeographic.com
ncbi.nlm.nih.gov	nejm.org	nytimes.com
oreilly.com	skynews.com.au	smh.com.au
springer.com	time.com	wikipedia.org
wiley.com		

Table 2: The Internet domains used for extracting word occurrences.



suicide site:abc.net.au

About 83,500 results (0.30 seconds)

Figure 1: When searching for the word *suicide*, 83,500 results are shown.

We use Google search to determine word occurrences by restricting the search to these chosen Internet domains. For example, to look for the word *suicide* in the ABC News, the search command would be `suicide site:abc.net.au`, as shown in Figure 1. A number of search results (i.e., 83,500), which we consider occurrences, is shown before actual results are displayed. Figure 1 is for illustration purposes only. In practice, we searched Google programmatically.

To compute $P(e1 \rightarrow e2)$, which is a single probability, we use Google to estimate the frequency count of the occurrences of both $e1$ and $e2$ in a domain, $C(e1, e2)$, and divide it by the frequency count of the occurrences of $e1$ alone in the same domain, $C(e1)$. $P(e2 \rightarrow e1)$ is calculated using the same method. This will result in unnormalised versions, which will be normalised as described below.

$$P'(e1 \rightarrow e2) = \frac{C(e1, e2)}{C(e1)} \tag{10}$$

$$P'(e2 \rightarrow e1) = \frac{C(e1, e2)}{C(e2)} \tag{11}$$

In (10) and (11), $C(e1) \neq 0$ and $C(e2) \neq 0$ to avoid zero counts[5]. To normalise Equations (10)

---

[4] A probability distribution is a mathematical function that describes the likelihoods of all possible outcomes that a random variable can take. Probability distributions not only allow us to quantify uncertainty but also provide a comprehensive view of all possible values and their associated probabilities. Hence, we employ Bayesian statistics as opposed to frequentist statistics in hypothesis testing to harness these advantages.

[5] Haldane (1956) suggests adding 0.5 to every count if $C(e1) = 0$ or $C(e2) = 0$. However, we did not experience zero counts during the experiments.

and (11), both are divided by their sum.[6]

$$P(e1 \rightarrow e2) = \frac{P'(e1 \rightarrow e2)}{P'(e1 \rightarrow e2) + P'(e2 \rightarrow e1)} \tag{12}$$

$$P(e2 \rightarrow e1) = \frac{P'(e2 \rightarrow e1)}{P'(e1 \rightarrow e2) + P'(e2 \rightarrow e1)} \tag{13}$$

Equations (12) and (13) are effectively conditional probabilities. We apply Equations (12) and (13) repeatedly for each domain outlined in Table 2. This process results in two distinct lists of probabilities. Each of these lists provides a complete range of likelihoods. A probability[7] is assigned to each likelihood, effectively quantifying the uncertainty. As a result, we have $f(e1 \rightarrow e2)$ and $f(e2 \rightarrow e1)$.

Prior Specification

Prior specification is a process of selecting and defining a prior distribution in the Bayesian framework. More specifically, it involves choosing a type of distributions and its parameters to fit $f(e1 \rightarrow e2)$ and $f(e2 \rightarrow e1)$ given the experimental values obtained from the Internet domains. We employ the Sum of Square Error (SSE) as the criterion that determines the best-fitting among the following types of distributions:

- **Normal distribution** is the most commonly used distribution.
- **Cauchy distribution**: One characteristic of the Cauchy distribution is its heavy tails. In other words, it has a higher probability of extreme values.
- **Exponential distribution**: The exponential distribution is often used to model the time between events. Certain words such as *machine learning* may appear more often through time, so it becomes an excellent choice.
- **Gamma distribution**: The gamma distribution is more flexible than the exponential distribution due to the fact it has two parameters whereas the exponential distribution has one.
- **Inverse-gamma distribution**: The inverse-gamma distribution is a probability distribution of the inverse of a random variable that follows a gamma distribution.
- **Log-normal distribution**: The log-normal distribution is often used to model data that is positively skewed, but taking the logarithm of the data results a normal distribution.
- **Student's *t*-distribution**: The student's *t*-distribution is a continuous probability distribution that is similar to the normal distribution in shape but with heavier tails.

Given that we do not know the underlying distributions of word occurrences, our expectation is that if word occurrences exhibit specific characteristics, at least one of the pre-selected distributions will be able to capture distinctive features. Furthermore, all of them are often used as a prior distribution in Bayesian statistics. If a probability distribution fits the experimental values obtained from the Internet domains, simulated samples from the probability distribution should look indistinguishable compared with these experimental values. Hence, the probability distribution that has the least SSE is deemed as the best distribution. Indeed, the `fitter` package[8] returns the best distribution based on the smallest SSE and its parameters that describe the chosen distribution.

Prior Predictive Checks

After choosing a probability distribution as described in Section 4.1.1, we still need to check whether the chosen distribution is a good fit. We use Prior Predictive Checks (PPC) (Kruschke, 2015; Lambert, 2018) as a guide to judge the fit.

The concept is as follows: If we cannot tell which data is generated from the probability distribution and which one comes from the experimental values, we can conclude it is a good (enough) fit. Many statisticians use the maximum or minimum value as the criterion. In our specific case, we utilize the maximum criterion for assessing $f(e1 \rightarrow e2)$ and the minimum for evaluating $f(e2 \rightarrow e1)$. That is, it is anticipated that half of the time (i.e., 50%) the maximum or minimum value will come from simulated samples, and the other half it will come from experimental values if the chosen distribution fits best. Nevertheless, requiring an exact 50% would be overly strict, so we have extended the range to $50\% \pm 1\%$ to accommodate some variability.

Algorithm 1 shows the pseudocode. M is the total number of runs that we ask the probability distribution to simulate samples; N is how many

---

[6] Bayesian statistics is inherently subjective in the sense that it allows individuals to express their beliefs through priors. Whether someone articulates $e1 \rightarrow e2$ or $e2 \rightarrow e1$ as expressed in Equations (12), (13), or any other forms, it remains an expression of their subjective belief.

[7] In Bayesian statistics, a probability can be interpreted as a measure of uncertainty.

[8] `https://fitter.readthedocs.io/en/latest/`

Algorithm 1 Prior Predictive Checks

```
Require: $m \geq 0, n \geq 0, i \geq 0$
$M \leftarrow m$
$c \leftarrow 0$
while $M \neq 0$ do
    $N \leftarrow n$
    $i \leftarrow 0$
    while $N \neq 0$ do
        $p \leftarrow \text{pdf}(\theta)$
        $S[i] \leftarrow p$
        $i \leftarrow i + 1$
        $N \leftarrow N - 1$
    end while
    $j \leftarrow \max(S)$ {Or $\min(S)$}
    $k \leftarrow \max(P)$ {Or $\min(P)$}
    if $j \geq k$ then
        $c \leftarrow c + 1$
    end if
    $M \leftarrow M - 1$
end while
return $c/M$
```

simulated samples we need for each run. Once N samples are generated, we retrieve the maximum or minimum value and store it in j. We also retrieve the maximum or minimum value from the experimental values and store it in k. If $j \geq k$ holds, we increment c by 1. Thus, c/M, which is the last line in Algorithm 1, is the percentage of times the maximum or minimum values come from simulated samples across M runs.

4.1.2 Posteriors

To approximate posterior distributions, we use the Stan open-source probabilistic programming language[9] (Kruschke, 2015; Lambert, 2018). Given Example 1, Figure 2 shows the posteriors of $f(\textit{suicide} \rightarrow \textit{death} \mid \mathbf{X})$ and $f(\textit{death} \rightarrow \textit{suicide} \mid \mathbf{X})$. These posteriors indicate that *suicide* $\rightarrow$ *death* is more likely since its posterior density is skewed toward the higher end of probabilities, making it more likely than *death* $\rightarrow$ *suicide*.

4.1.3 Bayes Factor

Given that both Model 1 and Model 2 in Equation (5) are posterior distributions, we use Bayes Factor (BF) (Lambert, 2018; McElreath, 2015) to reject either the null (i.e., Model 1) or alternative (i.e., Model 2) hypothesis. If BF is greater than 1, we opt for Model 1; Otherwise, we select Model 2.

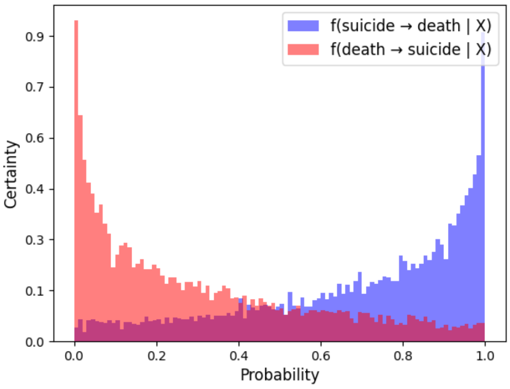


Figure 2: Both posteriors $f(\textit{death} \rightarrow \textit{suicide} \mid \mathbf{X})$ and $f(\textit{suicide} \rightarrow \textit{death} \mid \mathbf{X})$ are shown.

BF	**Interpretation**
BF < e^{-300}	Decisive evidence for Model 2
e^{-300} < BF < e^{300}	Reject Option (Neither)
BF > e^{300}	Decisive evidence for Model 1

Table 3: Thresholds are used for the study.

However, it is important to note that BF tends to favour one model over the other even when both have reasonable likelihoods. Hence, Murphy (2013) suggests a threshold. By enforcing the threshold, we allow BF to make a choice only if it is confident enough. Table 3 provides a guideline about how we choose the model. When BF lies on an extreme, either a positive infinity (in which case we consider as e^{300}) or close to 0 (in which case we consider as e^{-300}), it is very confident one model is preferred over the other. Otherwise, as a Reject Option (Bishop, 2007; Murphy, 2013), neither model is chosen[10]. That is, the Bayesian framework predicts a causal direction either $e1 \rightarrow e2$, $e2 \rightarrow e1$, or *neither*. Predicted directions that fall *outside* the Reject Option will be fed to the next phase — Pre-trained BERT, as discussed in Section 4.2.

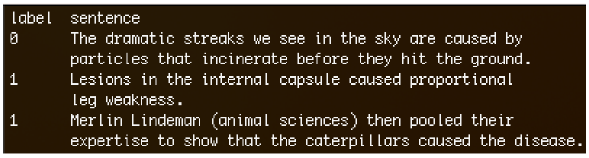
4.2 Phase 2: Pre-trained BERT

While the Bayesian framework is capable of identifying causal directions, a lack of understanding semantics means its capability is rather limited. Therefore, we turn to BERT (Devlin et al., 2019). Although BERT has many variants, we stick to the BERT uncased base model. Our implementation is largely based on a Jupyter notebook made available

---

[9] `https://mc-stan.org`

[10] Strictly speaking, the Bayesian framework still predicts either $e1 \rightarrow e2$ or $e2 \rightarrow e1$ with BF falling between e^{-300} and e^{300}.



```
label  sentence
0      The dramatic streaks we see in the sky are caused by
       particles that incinerate before they hit the ground.
1      Lesions in the internal capsule caused proportional
        leg weakness.
1      Merlin Lindeman (animal sciences) then pooled their
       expertise to show that the caterpillars caused the disease.
```

Figure 3: The first three rows of the labelled dataset is shown.

by Rothman (2021)[11].

```
"The dramatic <e1>streaks</e1> we see in
     the sky are caused by <e2>particles
    </e2> that incinerate before they
    hit the ground."
```

Example 2: A sample sentence.

In this phase, we refine the performance of BERT, which was originally trained and made available through Hugging Face[12], by using the sentences from the test data that have causal directions predicted from the previous phase. More specifically, for each sentence in the test data, the Bayesian framework predicts either $e1 \rightarrow e2$, $e2 \rightarrow e1$, or *neither*. When the framework predicts either $e1 \rightarrow e2$ or $e2 \rightarrow e1$, we include the corresponding sentences as input to BERT, along with the predictions. Given that the Bayesian framework inherently considers uncertainty, not all sentences from the test data are passed to BERT (i.e., some have *neither*). Hence, we rely on BERT to predict those that the Bayesian framework labels *neither*. During the dataset construction process, the placeholders `<e1>` and `<e2>` are removed from sentences. `label` serves as the target variable, with 0 representing the direction $e2 \rightarrow e1$ and 1 representing the direction $e1 \rightarrow e2$. Let us take Example 2 as an example. According to the Bayesian framework, in this instance, the predicted causal direction is *particles* $\rightarrow$ *streaks* because BF < e^{-300}. Hence, we include this sentence and its predicted direction in the dataset to BERT. We continue the dataset construction process for the rest of predicted directions, as depicted in Figure 3.

5 Experiments

To evaluate our method, we have two experimental set-ups: (a) Random and (b) Bayesian + Pre-trained BERT.

To the best of our knowledge, there are no existing semi-supervised models for detecting causal directions. Thus, the random approach serves as the baseline, which blindly guesses causal directions. While one might argue that a baseline should always predict $e2 \rightarrow e1$ since it is the majority direction, it is important to note that the proposed method does not leverage such information. Hence, the random approach is more appropriate for our evaluation.

Given the SemEval-2010 (Task 8) dataset is well known, it might be tempting to consider using an established supervised model as a baseline. Using a supervised model as a baseline in a semi-supervised classification scenario is not recommended for several reasons. Firstly, supervised models are trained on labelled data whereas semi-supervised models lack annotated labels. This difference renders any experimental results incomparable: using a supervised model as a baseline can have unrealistic expectations for the performance of a semi-supervised model. Lastly, the primary objective of our study is to demonstrate the resemblance between word occurrences and the characteristics of causal directions. Using a supervised model as a baseline may distract from this objective.

(a) Random In this set-up, we simulated a probability from *Uniform*$(0, 1)$. If it was greater than 0.5, we would classify as $e1 \rightarrow e2$. Otherwise, $e2 \rightarrow e1$. We ran this set-up for 10,000 times and averages were recorded.

(b) Bayesian + Pre-trained BERT In this particular set-up, we ran the two-phase method described above. That is, we used the predicted directions generated from the Bayesian framework and fed them into pre-trained BERT, which made predictions on the rest of test data. This set-up was run 10 times.

We conducted the experiment under two distinct settings in the Bayesian framework. In the first setting, we examined whether the priors were satisfied with PPC (referred to as PPC$+$), resulting in predictions for 7 out of 328 cases. In the second setting, we did not apply any prior checks (referred to as PPC$-$), and this yielded predictions for 281 out of 328 cases. This allows us to gain insights into the quality of the data generated by the Bayesian framework. Because there were not enough predicted directions generated in the PPC$+$ setting, primarily due to a substantial number of the priors being rejected by PPC (for a detailed

---

[11] https://github.com/PacktPublishing/Transformers-for-Natural-Language-Processing/blob/main/Chapter02/BERT_Fine_Tuning_Sentence_Classification_DR.ipynb

[12] https://huggingface.co

Set-Up	Precision (SD[a])	Recall (SD)	F1 (SD)	Accuracy (SD)
a. Random	40.81% (2.71%)	49.99% (4.32%)	44.90% (3.18%)	49.95% (2.74%)
b. Bayesian+Pre-trained BERT (PPC+)	46.00% (2.56%)	44.93% (10.83%)	44.89% (6.22%)	55.98% (2.18%)
Bayesian+Pre-trained BERT (PPC−)	**46.82%** (1.85%)	**52.09%** (7.86%)	**49.10%** (4.14%)	**56.31%** (1.39%)

Table 4: All the experimental set-ups results are summarised.

[a] SD is short for Standard Deviation.

explanation, refer to Section 8), we augmented data by using `ContextualWordEmbsAug` from `nlpaug.augmenter` before running pre-trained BERT (Tunstall et al., 2022).

6 Results

Table 4 provides a summary of the results from all experimental set-ups (See Appendix A.1 for individual runs). In the second set-up, when the Bayesian framework creates data and feeds it into pre-trained BERT, the two-phase method yields two distinct outcomes based on the presence or absence of PPC. With Bayesian+Pre-trained BERT (PPC+), this setting achieves comparable performance to the baseline, with an F1 score of 44.89% compared to 44.90%; without PPC (i.e., PPC−), it outperforms significantly better compared with the baseline, achieving an F1 score of 49.10% versus 44.90%. To sum up, the two-phase method performs best when PPC is de-activated. PPC is necessary for assessing the trustworthiness of priors even if it led to worse performance.

7 Discussion

Although the Bayesian framework is inherently statistically sound, it is not immune to failure when confronted with certain word occurrences used in constructing the priors. In this section, we explore the Bayesian framework more comprehensively, aiming to understand the rationale behind the specific predictions made by the Bayesian framework, especially two cases from the test data where the predictions were incorrect.

1. *rain* → *cancellation* The first case, as shown in Example 3, suggests *rain* → *cancellation*, but the Bayesian framework incorrectly classified it as *cancellation* → *rain*. *rain* → *cancellation* approximates the gamma distribution whereas *cancellation* → *rain* follows the Student's t-distribution. Figure 4, which shows the posteriors for both *rain* → *cancellation* and *cancellation* → *rain*, clearly favours *cancellation* → *rain*.

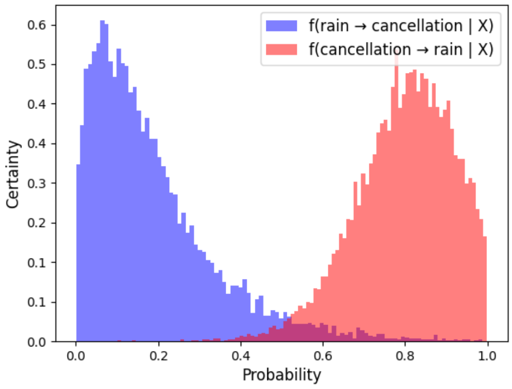


Figure 4: Both $f(\textit{rain} \rightarrow \textit{cancellation} \mid \mathbf{X})$ and $f(\textit{cancellation} \rightarrow \textit{rain} \mid \mathbf{X})$ are shown.

```
"<e1>Rain</e1> caused <e2>cancellation</
    e2> of the event in 1877, so
    enforcement of the new law had to
    wait until 1878."
Cause-Effect(e1,e2)
```

Example 3: A sample sentence. The last line indicates *rain* → *cancellation*.

Referring to Equation (10) and (11), in situations where there exists co-occurrence between $e1$ and $e2$, which is $C(e1, e2) > 0$, the entity with a higher frequency count is always identified as the effect when evaluating the entity counts. In this specific instance, the prevalence of the term $C(\textit{rain})$ typically surpasses that of $C(\textit{cancellation})$. The reason *rain* appears more often in the text could be attributed to the fact that *rain* is commonly used in everyday language, particularly weather-related contexts like events related to weather conditions.

2. *moon* → *perturbations* In the second case, as shown in Example 4, the correct answer is *moon* → *perturbations*, but the Bayesian framework erroneously misclassified it as *perturbations* → *moon*. *perturbations* → *moon* approximates the Student's t-distribution whereas *moon* → *perturbations* follows the inverse-gamma distribution. Figure 5, which illustrates the posterior distributions for both *moon* → *perturbations* and *perturbations* → *moon*, distinctly favours *perturbations* → *moon*.

```
"The thin F ring on the left of the
    image shows the <e1>perturbations</
    e1> caused by the <e2>moon</e2>
    Prometheus."
Cause-Effect(e2,e1)
```

Example 4: A sample sentence. The last line indicates *moon* → *perturbations*.

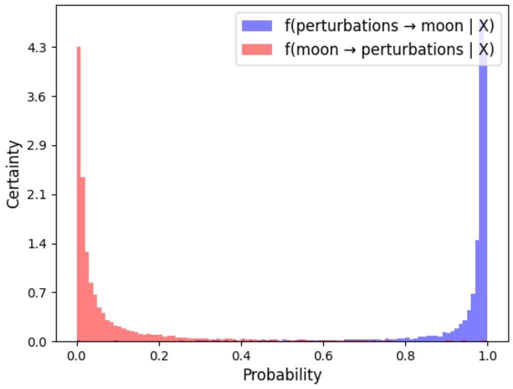


Figure 5: Both $f(moon \rightarrow perturbations \mid \mathbf{X})$ and $f(perturbations \rightarrow moon \mid \mathbf{X})$ are shown.

In consideration of Equation (10) and (11), $P(perturbations \rightarrow moon)$ is higher than $P(moon \rightarrow perturbations)$ in all the domains, except for wiley.com, springer.com, and ncbi.nlm.nih.gov. What they have in common is their focus on providing access to scientific research articles, publications, or resources. Given the context, which appears to be closely related to astronomy, it is likely that these specific domains cover relevant topics in this field. As further work, it is suggested to automatically identify and select the most suitable domains for the calculation of priors.

8 Further Work

There are many areas we can explore to improve the study further. In this section, we present three of them: Earth Mover's Distance, Mixture Models, and Bayesian Network.

Earth Mover's Distance. While conducting PPC in Section 4.1.1, we utilized a simple method to determine the percentage of times when the maximum or minimum value originated from simulated samples. This approach offers the advantage of being straightforward to implement because it involves comparing two numbers. However, it may not always provide reliable results. Gelman et al. (2004); Lambert (2018) recommend using Kullback-Leibler Divergence (KL Divergence) to compare two distributions. However, KL Divergence is sensitive to the choice of a reference distribution, which can be a drawback. An alternative way to do so is Earth Mover's Distance (EMD) (Rubner et al., 2000) or Word Mover's Distance (Kusner et al., 2015; Sun et al., 2019). EMD is a methodology to compute "distances" between the experimental values and the listed distributions in Section 4.1.1. The distribution with the shortest distance is considered as the best fit.

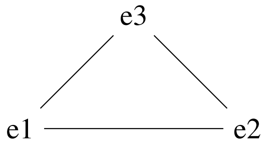


Figure 6: $e1$, $e2$ and $e3$ show causal relations.

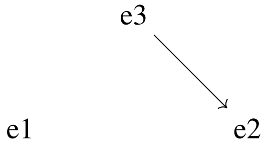


Figure 7: $e3 \rightarrow e2$ is one possible way if $e3 \rightarrow e2$ exists.

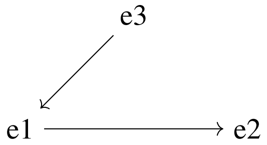


Figure 8: $e3 \rightarrow e1 \rightarrow e2$ is another possible way if $e3 \rightarrow e2$ exists.

Mixture Models. The distributions listed in Section 4.1.1 are not suited for modelling multi-modal data, which we frequently encountered in word occurrences, so a significant number of priors was rejected by PPC. Mixture models (Gelman et al., 2004) could be good substitutes. They are in fact probability distributions, which can account for data that exhibits multimodal and skewness. The idea is to take numerous probability distributions and stack them together using a linear combination.

Bayesian Network. We have so far considered a single causal relation in the sentence. To extend the analysis further, we can consider a multiple causal relations' scenario. That is, a model determines causal directions among all the causal relations. Let the diagram shown in Figure 6 be underlying causal relations. The task is to determine whether the causal direction $e3 \rightarrow e2$ exists. If $e3 \rightarrow e2$ exists, there are two possible networks as shown in Figures 7 and 8. We may be able to extend the proposed method to compute the likelihoods of Figures 7 and 8 if $e3 \rightarrow e2$ exists.

9 Conclusion

In this paper, we have shown empirically that word occurrences resemble the characteristics of causal directions. This finding provides significant implications and contributes significantly to our understanding of the data generation process underpinning causal directions.

Acknowledgements

We would like to thank Rolf Schwitter at Macquarie University for his advice and expertise, particularly in the area of knowledge graphs. We would also like to thank Houying Zhu at Macquarie University who offers feedback on Bayesian inference. Finally, we would like to thank Roman Marchant Matus at University of Technology Sydney for introducing us to the world of Bayesian statistics.

References

Christopher M. Bishop. 2007. *Pattern Recognition and Machine Learning (Information Science and Statistics)*, 1 edition. Springer.

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.

Andrew Gelman, John B. Carlin, Hal S. Stern, and Donald B. Rubin. 2004. *Bayesian Data Analysis*, 2nd ed. edition. Chapman and Hall/CRC.

Oren Glickman, Ido Dagan, and Moshe Koppel. 2005. Web based probabilistic textual entailment. In *Proceedings of the 1st Pascal Challenge Workshop*, pages 33–36.

J. B. S. Haldane. 1956. The estimation and significance of the logarithm of a ratio of frequencies. *Annals of Human Genetics*, 20.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden. Association for Computational Linguistics.

Paul R. Kroeger. 2005. *Analyzing Grammar: An Introduction*. Cambridge University Press.

John Kruschke. 2015. *Doing Bayesian Data Analysis (Second Edition)*. Academic Press, Boston.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 957–966, Lille, France. PMLR.

Ben Lambert. 2018. *A Student's Guide to Bayesian Statistics*. SAGE Publications.

Zhongyang Li, Xiao Ding, Kuo Liao, Bing Qin, and Ting Liu. 2021. Causalbert: Injecting causal knowledge into pre-trained models with minimal supervision.

Richard McElreath. 2015. *Statistical Rethinking, A Course in R and Stan*. Chapman and Hall/CRC.

Farhad Moghimifar, Afshin Rahimi, Mahsa Baktashmotlagh, and Xue Li. 2020. Learning causal bayesian networks from text.

Kevin P. Murphy. 2013. *Machine learning : a probabilistic perspective*. MIT Press, Cambridge, Mass.

Denis Rothman. 2021. *Transformers for Natural Language Processing: Build Innovative Deep Neural Network Architectures for NLP with Python, PyTorch, TensorFlow, BERT, RoBERTa, and More*. Packt Publishing.

Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. 2000. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99.

Murray Singer, Michael Halldorson, Jeffrey C Lear, and Peter Andrusiak. 1992. Validation of causal bridging inferences in discourse understanding. *Journal of Memory and Language*, 31(4):507–524.

Chao Sun, King Tao Jason Ng, Philip Henville, and Roman Marchant. 2019. Hierarchical word mover distance for collaboration recommender system. In *Data Mining*, pages 289–302, Singapore. Springer Singapore.

Patrick Suppes. 1973. A probabilistic theory of causality. *British Journal for the Philosophy of Science*, 24(4):409–410.

Lewis Tunstall, Leandro von Werra, and Thomas Wolf. 2022. *Natural Language Processing with Transformers: Building Language Applications with Hugging Face*. O'Reilly Media, Incorporated.

A Appendix

A.1 Experiments

Tables 5 and 6 show the individual runs of

- Bayesian+Pre-trained BERT (PPC+), and
- Bayesian+Pre-trained BERT (PPC−)

respectively.

Run	**Precision**	**Recall**	**F1**	**Accuracy**
1	42.62%	38.81%	40.62%	53.66%
2	47.51%	64.18%	54.60%	56.40%
3	49.18%	44.78%	46.88%	58.54%
4	45.60%	42.54%	44.02%	55.79%
5	47.65%	52.99%	50.18%	57.01%
6	48.33%	43.28%	45.67%	57.93%
7	46.88%	55.97%	51.02%	56.10%
8	44.34%	35.07%	39.17%	55.49%
9	46.67%	26.12%	33.49%	57.62%
10	41.22%	45.52%	43.26%	51.22%
Average	46.00%	44.93%	44.89%	55.98%
SD	(2.56%)	(10.83%)	(6.22%)	(2.18%)

Table 5: Results of Bayesian+Pre-trained BERT (PPC+) are shown.

Run	**Precision**	**Recall**	**F1**	**Accuracy**
1	46.20%	54.48%	50.00%	55.49%
2	46.67%	57.46%	51.51%	55.79%
3	48.94%	51.49%	50.18%	58.23%
4	46.88%	55.97%	51.02%	56.10%
5	46.99%	58.21%	52.00%	56.10%
6	47.65%	52.99%	50.18%	57.01%
7	45.60%	42.54%	44.02%	55.79%
8	49.18%	44.78%	46.88%	58.54%
9	47.51%	64.18%	54.60%	56.40%
10	42.62%	38.81%	40.62%	53.66%
Average	46.82%	52.09%	49.10%	56.31%
SD	(1.85%)	(7.86%)	(4.14%)	(1.39%)

Table 6: Results of Bayesian+Pre-trained BERT (PPC−) are shown.

The sub-band cepstrum as a tool for local spectral analysis in forensic voice comparison

Shunichi Ishihara and Frantz Clermont

Speech and Language Laboratory, Australian National University, Canberra, Australia
shunichi.ishihara@anu.edu.au, dr.fclermont@gmail.com

Abstract

This paper exploits band-limited cepstral coefficients (BLCCs) in forensic voice comparison (FVC), with the primary aim of locating speaker-sensitive spectral regions. BLCCs are sub-band cepstral coefficients (CCs) which are easily obtained by a linear transformation of full-band CCs. The transformation gives the flexibility of selecting any sub-band region without the recurrent cost of spectral analyses. Using multi-band BLCCs obtained by sliding a 600-Hz sub-band every 400 Hz across the full [0-5kHz] range, FVC experiments were attempted using citation recordings of the 5 Japanese vowels from 297 adult-male, native speakers. The FVC results give locations and ranges for the most speaker-sensitive sub-bands, and show that combining 3-4 of these yields comparable FVC performance with full-band CCs. Owing to their ability to easily extract locally-encoded speaker information from full-band CCs, it can be conjectured that BLCCs have a significant role to play in the search for meaningful interpretations of the numerical outcome of forensic analyses.

1 Introduction

In forensic voice comparison (FVC), the forensic scientist typically needs to compare a pair of speech recordings: the source-questioned and source-known samples, and to obtain the strength of evidence quantified by a likelihood ratio (LR).

For this purpose, it has become standard practice to parameterise the acoustic speech signal using low-dimensional vectors of cepstral coefficients (CCs). These are automatically extracted from any phonetic segments, and have been shown to be effective for speech and speaker classification. The effectiveness is attributable to the ability of low-ordered CCs to produce cepstrally-smoothed spectra with reduced sensitivity to "noninformation bearing variabilities" (Rabiner and Juang 1993: 169) and, thus, with increased distinctiveness. Such spectra may be obtained with full-band CCs which yield spectral representations over the full frequency range, or with sub-band CCs which give access to local regions within the full range.

Consistent with our long-term goal of interpreting the FVC outcome beyond numerical LR values, the present study focuses on sub-band CCs with the dual aim of (a) locating vowel spectral regions that are most sensitive to speaker differences, and (b) determining the extent to which such regions affect LR values compared to the full band from vowel to vowel. The motivation for this endeavour stems from an old premise (Peterson 1959: 151) that speaker information is not uniformly encoded throughout vowel spectra, i.e., there exist local regions of strong speaker and phonetic specificity. Supportive evidence has since been reported in a wide range of studies (*inter alia*: Goto et al. 2017; Hyon et al. 2012; Khodai-Joopari et al. 2004; Kitamura and Akagi 1995; Mohammadi et al. 2011; Mokhtari and Clermont 1994; Pols et al. 1973; Saito and Itakura 1982; van den Heuvel et al. 1993; Wang et al. 2016).

The presentation of our work is as follows. Sec. 2 describes and illustrates the method (Clermont 2022) adopted for obtaining sub-band CCs, hereafter referred to as band-limited CCs (BLCCs in short). The BLCC method affords flexibility and efficiency, two properties exploited in this work.

Sec. 3 recalls the basics of the LR framework. Sec. 4 concerns the multi-speaker vowel data used, the BLCC parameterisation applied to a sequence of sub-bands, the FVC procedures, and the LR-based metric for performance assessment. Sec. 5 presents full-band and sub-band FVC results for each vowel. Sec. 6 discusses the results in context of previous work, and Sec. 7 outlines potential ways forward.

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 40–50
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

2 The BLCC Method

This section focuses on the method employed for obtaining BLCCs by a linear transformation of full-band CCs. The method is described in Sec. 2.1, and its mathematical formulation is outlined in Sec. 2.2. In Sec. 2.3, the numerical and spectral behaviours of BLCCs show that the practical size for a BLCC vector depends on the fraction of the full-band's frequency range occupied by the sub-band's width.

2.1 Procedural steps

The BLCC method consists of three main steps encapsulated in Fig. 1. Steps (1) and (2) describe standard procedures of spectral analysis, which are applied to short-time frames of the speech signal sampled at some frequency F_s (Hz). The final step (3) concerns the linear transformation itself.

At Step (1), the all-pole linear-prediction (LP) model of speech production is adopted for two reasons: (a) It provides a reliable characterisation of the spectral resonance patterns of non-nasalised, voiced sounds; (b) It is thus expected that speaker differences are strongly encoded in the LP cepstral representation of the vowels used for this study.

Step (1) yields a log magnitude spectral (LMS) representation based on the LP model (order M), which spans the entire frequency range $[0, (F_s/2)]$ in Hertz (or $[0, \pi]$ in radians). The dashed curve in Fig. (2a) illustrates this representation also known as the "exact" LP-based LMS. Note that the frequency scale along the horizontal axis is kept linear in our experiments, thus leaving open the possibility of finding speaker-sensitive sub-bands without pre-defined nonlinear constraints.

The purpose of the Discrete Cosine Transform (DCT) at Step (2) is to expand the exact LMS as a Fourier cosine series of the so-called cepstral coefficients C_k. These are here referred to as full-band C_k since our LMS spans the full frequency range. The average of the full-band LMS is usually assumed to be zero, hence $C_0 = 0$. In practice, the series is truncated after M terms as follows:

$$S(\omega) = \sum_{k=1}^{M} C_k \cos(k\omega), \quad 0 \le \omega \le \pi \qquad (1)$$

The solid curve in Fig. 2(a) depicts the cepstrally-smoothed LMS resulting from the truncated series. As noted earlier, smoothing has the beneficial effect of enhancing spectral distinctiveness.

At Step (3), BLCCs are obtained using a method

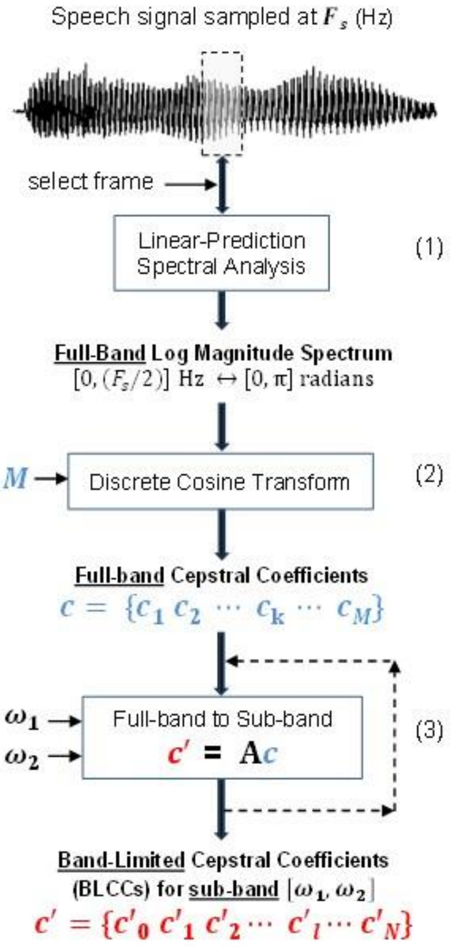

Speech signal sampled at F_s (Hz)
select frame
Linear-Prediction Spectral Analysis
(1)
Full-Band Log Magnitude Spectrum
$[0, (F_s/2)]$ Hz ↔ $[0, \pi]$ radians
M
Discrete Cosine Transform
(2)
Full-band Cepstral Coefficients
$c = \{c_1\, c_2 \cdots c_k \cdots c_M\}$
ω_1
ω_2
Full-band to Sub-band
$c' = Ac$
(3)
Band-Limited Cepstral Coefficients
(BLCCs) for sub-band $[\omega_1, \omega_2]$
$c' = \{c'_0\, c'_1\, c'_2 \cdots c'_l \cdots c'_N\}$

Figure 1: The BLCC method and its main steps.

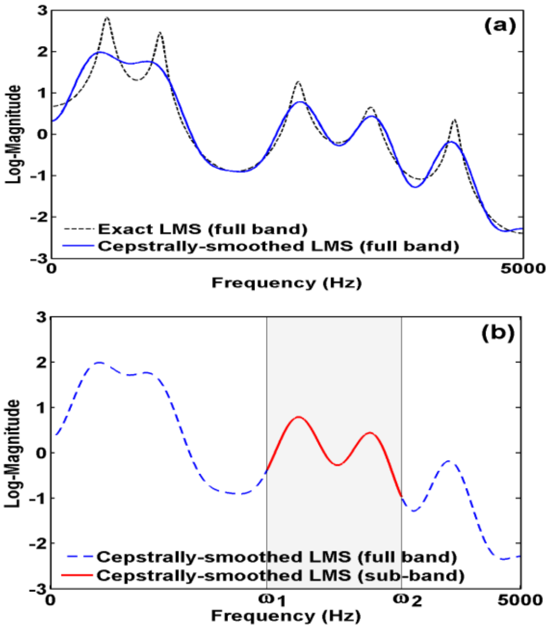


Figure 2: Spectral representations of a back vowel: **(a)** Exact LMS (full band) based on LP analysis (order M=14) at Step (1), overlaid with cepstrally-smoothed LMS based on Eq. (1) and on the C_k obtained at Step (2); **(b)** Sub-band region $[\omega_1, \omega_2]$ highlighted as an integral part of the full-band, cepstrally-smoothed LMS.

which affords the flexibility of selecting any sub-band region of the full-band spectrum without repeating the two previous steps. The central idea portrayed in Fig. 2(b) is this: Focusing on a sub-band region $[\omega_1, \omega_2]$ does not alter the fact that it forms an integral part of some full-band spectrum. It is therefore conceivable that sub-band cepstra are derivable from full-band cepstra. As shown in Clermont's (2022) study, the vector $\mathbf{c}'$ of BLCCs representing a sub-band can indeed be calculated using a linear transformation $\mathbf{A}$ of the vector $\mathbf{c}$ of full-band C_k. Sec. 2.2 outlines the transformation formulae. Key properties are illustrated in Sec. 2.3.

2.2 Linear transformation formulae

The mathematical goal is to represent a sub-band region $[\omega_1, \omega_2]$ of the full-band, cepstrally-smoothed LMS with a Fourier cosine series, such that its coefficients C_l' depend on the full-band C_k.

The band-limited analogue of Eq. (1) may be expressed as follows:

$$S(\omega(\omega')) = C_0' + \sum_{l=1}^{N} C_l' \cos(l\omega'), \ 0 \le \omega' \le \pi \quad (2)$$

where C_l' is the l-th BLCC and N is the series' upper bound. Eq. (2) includes C_0' because the average of $S(\omega(\omega'))$ within a sub-band may not be zero. The other $C_{l>0}'$ represent the spectral shape.

The frequency variable ω' defined below plays a key role by translating the sub-band interval $[\omega_1, \omega_2]$ to that of the full-band range $[0, \pi]$:

$$\omega' = \pi\left[\frac{(\omega-\omega_1)}{(\omega_2-\omega_1)}\right], \ \omega_1 \le \omega \le \omega_2 \quad (3)$$

From Eq. (3) it is easy to express the frequency variable ω of the full-band series as:

$$\omega(\omega') = \omega_1 + \left[\frac{(\omega_2-\omega_1)}{\pi}\right]\omega' = \omega_1 + W\omega' \quad (4)$$

where the scalar W is the ratio of the sub-band's width to the full-band's frequency range.

The notation $\omega(\omega')$ is a reminder that ω is itself a (band-dependent) function of ω', thus making it possible to substitute ω in Eq. (1) for Eq. (4) and to use standard formulae for the coefficients of the BLCC series in Eq. (2). These operations lead to:

$$C_l' = \sum_{k=1}^{M} a_{lk}\, C_k, \ l = 0,1,\dots,N \quad (5)$$

and to the matrix form $\mathbf{c}' = \mathbf{Ac}$ laid out below:

$$\begin{bmatrix} C_0' \\ C_1' \\ \vdots \\ C_l' \\ \vdots \\ C_N' \end{bmatrix} = \begin{bmatrix} a_{0,1} & \cdots & a_{0,k} & \cdots & a_{0,M} \\ a_{1,1} & \cdots & a_{1,k} & \cdots & a_{1,M} \\ \vdots & & \vdots & & \vdots \\ a_{l,1} & \cdots & a_{l,k} & \cdots & a_{l,M} \\ \vdots & & \vdots & & \vdots \\ a_{N,1} & \cdots & a_{N,k} & \cdots & a_{N,M} \end{bmatrix} \begin{bmatrix} C_1 \\ \vdots \\ C_k \\ \vdots \\ C_M \end{bmatrix} \quad (6)$$

The band-dependent weights a_{lk} are given in Eq. (7a) for $l = 0$ and in Eqs (7b)-(7c) for $l > 0$.

$$a_{lk,\,l=0} \;= \beta_k[\sin(k\omega_2) - \sin(k\omega_1)] \quad (7a)$$

$$a_{lk,\,l\neq kW} = \gamma_{lk}[(-1)^{l+1}\sin(k\omega_2) + \sin(k\omega_1)] \quad (7b)$$

$$a_{lk,\,l=kW} = \cos(k\omega_1) \quad (7c)$$

where:

$$\beta_k = \frac{1}{k(\omega_2-\omega_1)} \text{ and } \gamma_{lk} = \frac{2(kW)}{\pi[l^2-(kW)^2]} \quad (7d)$$

The implementation of Eqs (6) and (7) raises the question of how large N needs to be in practice. The empirical solution suggested in Clermont's study is to fix N at $M \times W$ (MW in short) rounded to the nearest integer, where W is the ratio defined above and M the size of the vector of full-band C_k.

2.3 Numerical illustrations of key properties

What do BLCCs look like, and how effective are they at preserving spectral resolution in a sub-band region for $N = MW$?

Fig. 3(a) gives a glimpse of BLCC series for two sub-bands selected from the same back vowel illustrated in Fig. (2). The full-band $C_{k=\{1\cdots M=14\}}$ were obtained by DCT of the full-band LP-based LMS ranging from 0 to 5 kHz. Eqs (6) and (7) were then used to calculate BLCCs for these sub-bands: [0.1-0.814]-kHz and [2.3-3.728]-kHz, the latter being twice as large as the former.

The coefficient C_0' in Fig. 3(a) is visibly much larger in the [0.1,0.814]-kHz range, thus indicating a prominent region in the lower part of the spectrum. The next C_l' exhibit a consistent trend for both sub-bands: A major drop in magnitude is noticeable after MW, followed by a clear decay towards zero.

Is the proposed truncation after MW detrimental to the spectral resolution in a sub-band region? To gain insights into this question, it is instructive to observe cepstrally-smoothed spectra representing the full band and the two sub-bands. The latter are overlaid in Figs 3(b)-(d) for $N = 0, 1, MW$, respectively. The $N = 0$ cases in Fig. 3(b) correspond to using only C_0'. While the spectral fits are expectedly very poor, these coefficients alone give a good indication of the respective levels of the prominences in the two sub-bands. Recruiting the next BLCC with $N = 1$ improves the approximation by capturing the overall slopes in Fig. 3(c). Finally, the spectral fits become very tight in Fig. 3(d) with $N = MW$.

In sum, the numerical evidence described above indicates that BLCCs after MW tend to contribute relatively little to the spectral representation of a sub-band. This is supported by the consistent decay towards zero seen in Fig. 3(a).

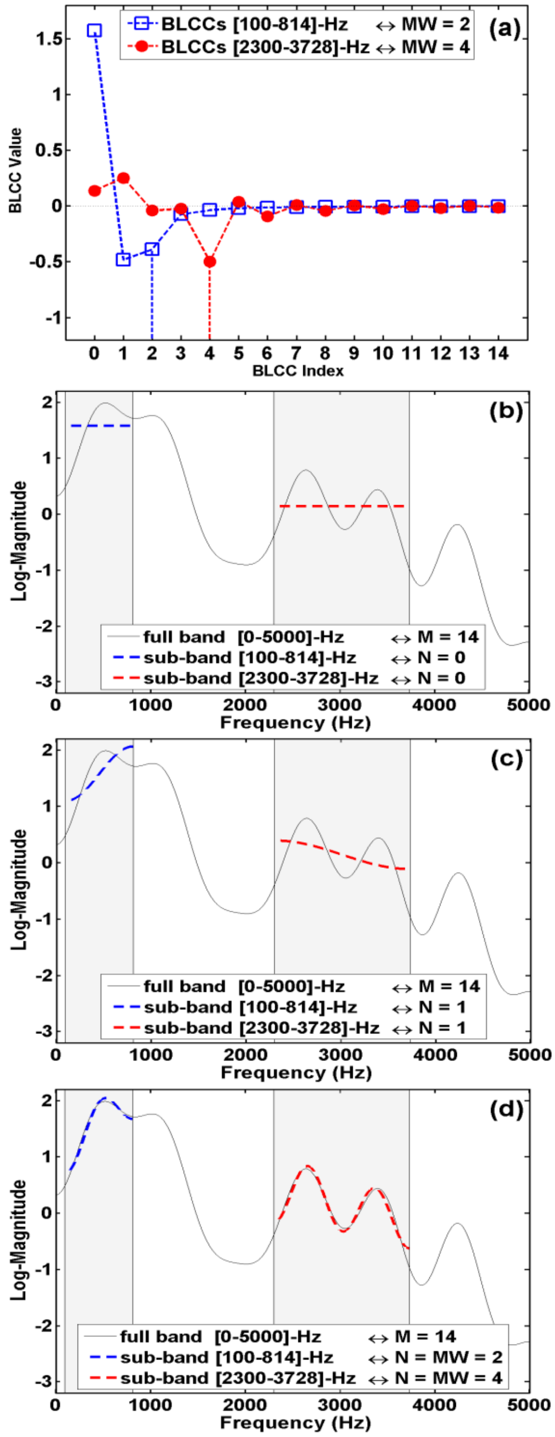


Figure 3: **(a)** BLCC series for two selected sub-bands. Cepstrally-smoothed spectra (full band based on Eq. (1), and sub-bands based on Eq. (2)) are superimposed for the following upper bounds: **(b)** $N = 0$, **(c)** $N = 1$, and **(d)** $N = MW$.

3 Likelihood Ratio Framework

The LR framework provides the theoretical foundation upon which voice evidence is analysed for source-inference purposes. In FVC, the task of the expert is to estimate the strength of voice evidence using the LR expressed as follows:

$$\mathrm{LR} = \frac{p(E = (X,Y)|H_p)}{p(E = (X,Y)|H_d)} \tag{8}$$

The LR is the ratio of two conditional probabilities: the numerator is the probability (p) of the evidence (E) given the prosecution (same-speaker) hypothesis (H_p), while the denominator is the probability given the defense (different-speaker) hypothesis (H_d).

The evidence (E) typically consists of the source-questioned sample (X) and the source-known sample (Y). In theory, the belief of the trier-of-fact regarding the hypotheses, which was developed by the previously presented evidence, is to be updated by the LR; the assessment of the newly presented evidence. In other words, the belief of the decision maker regarding the suspect being guilty or not changes as a new piece of evidence is presented to them in the form of a LR.

The further away from LR=1, the more strongly the LR supports either of the competing hypotheses.

4 Experimental Procedures

4.1 Speech material and parametrisation

The speech materials were taken from a Japanese dataset of 297 speakers (between 20 and 60 years old) as described in Osanai et al. (1995). The citation recordings (landline telephone calls) of the 5 vowels (2 non-contemporaneous sessions × 2 tokens) were used for the FVC experiments.

The sampling frequency is 10 kHz because the high-end of the telephone bandpass is around 4.5-kHz in Japan, i.e., the available full-band extends from 0 to 5 kHz. Full-band CCs were extracted by linear-prediction (LP) analysis (order 14) of each vowel's central frame.

Using the sub-band transformation explained in Sec. 2, BLCCs were obtained from the full-band CCs by scanning the full range with a 600-Hz sub-band shifted every 400 Hz. This process yielded 12 vectors of BLCCs corresponding to the 12 sub-bands listed in Table 1.

1	[0, 0.6]	2	[0.4, 1.0]	3	[0.8, 1.4]
4	[1.2, 1.8]	5	[1.6, 2.2]	6	[2.0, 2.6]
7	[2.4, 3.0]	8	[2.8, 3.4]	9	[3.2, 3.8]
10	[3.6, 4.2]	11	[4.0, 4.6]	12	[4.4, 5.0]

Table 1: Limits $[\omega_1, \omega_2]$ in kHz of the 12 sub-bands.

Following the definition given in Sec. 2.2, the upper bound for the BLCC series representing a 600-Hz sub-band may be fixed at $MW = 14 \times \frac{600}{5000} = 1.68$ and then rounded up to 2 for practical use. Per sub-band, the total number of BLCCs is 3 including the 0th-order one. A FVC system incorporating BLCCs was then employed to calculate LRs for each of the 12 sub-bands.

4.2 Data partitioning and LR calculation

The 297 speakers were randomly divided into three mutually-exclusive batches (99 speakers each). These were used as the test, background, and calibration databases in a cross-validation manner, resulting in six-fold cross-validation experiments. The results of the six experiments were averaged for comparison.

The LR calculation is a two-stage process consisting of a feature-to-score stage and a score-to-LR stage. A statistical model commonly used in linguistic-phonetic FVC is the Multivariate Kernel Density (MVKD) model for the feature-to-score stage (Aitken and Lucy 2004). The output of the MVKD model is a score, and the score is converted to a LR value at the score-to-LR stage. The MVKD returns a score for a pair of recordings under comparison by assessing their similarity and typicality. The necessary statistical information for typicality is obtained from the background database. The score-to-LR conversion, also called “calibration”, is performed via logistic regression (Morrison 2013). The logistic regression weights are determined using the calibration database.

4.3 Performance assessment

The log-LR-cost (C_{llr}) is a standard metric for assessing LR-based inference systems in forensic science. Eq. (9) is the formula for C_{llr}, where N_{SS} and N_{DS} are the numbers of the same-speaker (SS) and different-speaker (DS) LRs, respectively. The SS LRs are indexed by i and the DS LRs by j.

$$C_{llr} = \frac{1}{2}\left(\frac{1}{N_{SS}}\sum_{i}^{N_{SS}} log_2\left(1+\frac{1}{LR_{SS_i}}\right) + \frac{1}{N_{DS}}\sum_{j}^{N_{DS}} log_2\left(1+LR_{DS_j}\right)\right) \quad (9)$$

The first $log_2(\cdot)$ is the cost function for the SS LRs and the second one is for the DS LRs. The C_{llr} is the grand average between the mean cost of the SS LRs and that of the DS LRs. The lower the C_{llr}, the better in performance.

5 Experiment Descriptions and Results

Two FVC experiments were run separately per vowel, and the results are jointly charted in Fig. 4.

In Experiment 1, speaker information locally-encoded in the spectrum was investigated vowel-by-vowel by conducting the experiments with the multi-band BLCCs (see Table 1 for the specific locations of the sub-bands).

The C_{llr} values obtained for the 12 sub-bands are displayed as a red curve at the bottom plot of each panel included in Fig. 4. Each C_{llr} value (Y-axis) is given against the central frequency (X-axis) of the sub-band. The horizontal dashed line (in blue) indicates the overall mean of the 12 C_{llr} values for the vowel. Expected formant-frequency ranges (F1, F2 and F3) taken from Kinoshita et al. (2022) are also marked for each vowel.

In Experiment 2, the sub-band LRs obtained from Experiment 1 were fused from two to all sub-bands as per the following list (r) = {2,3,…,12}. All possible combinations of r sub-bands $\binom{12}{r}$ were also included.

In the top plot of each panel, the best (lowest) C_{llr} value is given for each r together with the C_{llr} value of the single best sub-band (r=1). The C_{llr} for the full-band CCs is indicated by the horizontal dotted line. The three best sub-bands (fused) are highlighted in blue in the bottom plot, and the three worst sub-bands (fused) are highlighted in pink.

5.1 Results: Experiment 1

The red curve included in the bottom plot of each panel (Fig. 4) stays consistently below C_{llr}=1, implying that every spectral region specified by the sub-bands carries some useful speaker information for FVC. However, the fluctuations in the C_{llr} curves indicate that speaker-specific information is not evenly distributed throughout the entire frequency range, and the distributional patterns are distinctive for each vowel. It is worth noting that the C_{llr} value consistently increases for the rightmost sub-band [4.4, 5.0] kHz, meaning that this spectral region contains relatively less speaker-specific information. This may be due to the upper

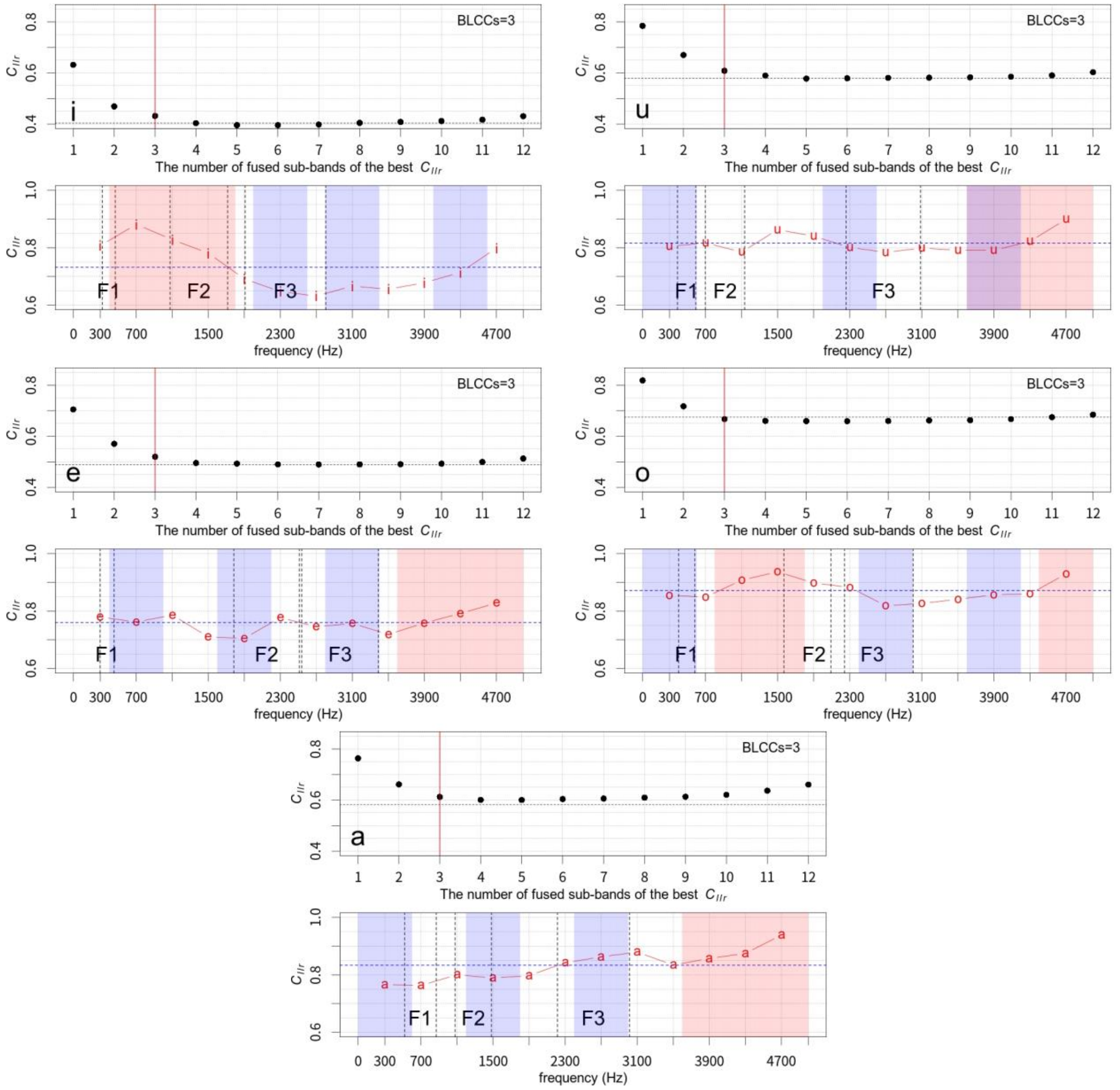


Figure 4: The results for each of the five Japanese vowels are grouped in a separate panel. The top plot in each panel contains the best C_{llr} values for the fused r (=1 to 12) sub-bands. The horizontal dotted line indicates the C_{llr} value for the full-band CCs. The vertical red solid line indicates r=3 for which sub-band performance becomes very close to the full-band result. The bottom plot in each panel gives the profile of C_{llr} values (in red) for the 12 sub-bands. The horizontal dashed line (in blue) indicates the C_{llr} value averaged over the 12 sub-bands. The sub-band regions highlighted in blue are the three best-performing sub-bands (fused), and the sub-bands highlighted in pink are the three worst-performing sub-bands (fused).

limit of the Japanese telephone band-pass located near 4.5 kHz (Rose et al. 2003).

The distributional patterns of speaker-specific information are particularly contrastive between /i/ and /a/. The information is more strongly encoded in the mid- and high-frequency regions of /i/, roughly between 1.9 and 4.3 kHz (covering F3 and beyond). By contrast, for /a/, it is the low-frequency region up to about 1.9 kHz (spanning F1 and F2) which carries the bulk of speaker-specific information. These findings agree with the observations reported in Osanai et al. (2018). Their study based on sub-band cepstral distances points to roughly 2.0 kHz as the frequency below which speaker verification accuracy was relatively higher for /a/, and roughly 1.9 kHz above which speaker verification performed relatively better for /i/.

For the other vowels (/u, e, o/), the ups and downs of the C_{llr} curves are overall less dynamic than those for /i/ and /a/. Yet, some alternations in C_{llr} are still evident. For instance, the C_{llr} values are marginally lower in the range between approximately 2.7 and 3.9 kHz (spanning F3 and beyond) for /u/ and /o/ and around 1.5-1.9 kHz (spanning F2) for /e/.

Kinoshita (2001) found that F2 of /i/ and F2 and F3 of /e/ are strong acoustic features for Japanese FVC. With more specific details based on F-ratios, Khodai-Joopari et al. (2004) reported that the spectral regions of 1.7-2.7 kHz (spanning F3) and 3.7-4.5 kHz (extending beyond F3) of /i/, and the spectral regions of 1.4-2.4 kHz (spanning F2), 2.6-3.7 kHz (spanning F3) and 3.8-4.5 kHz (extending over F3) of /e/, are potentially useful for speaker classification based on Japanese vowels.

The results obtained from Experiment 1 mostly agree with the findings from the two studies referenced above, in that the spectral or formant regions identified as promising returned categorically low C_{llr} values. For example, the lowest C_{llr} value (=0.63171) of all vowels lies within the frequency range pointed out by Khodai-Joopari et al. (2004) for /i/. Likewise, the lowest C_{llr} value (=0.70510) for /e/, also the lowest amongst vowels /e, a, o, u/, also occurs in the F2 range pointed by Kinoshita (2001).

For the back vowels (/u, o, a/), the frequency range spanning and/or extending beyond F3 is reportedly a good candidate for Japanese speaker classification (Khodai-Joopari et al. 2004). Some studies also report the usefulness of F3 of back vowels as a speaker discrimination feature in English (Mokhtari and Clermont 1996; Sambur 1975). As noted above, the importance of the frequency region spanning F3 and beyond holds true in our results for /u/ and /o/. This point will be revisited in describing results from Experiment 2.

5.2 Results: Experiment 2

Turning our attention to the top plots in each panel, it can be observed that regardless of the vowels, performance is improved by fusing multiple sub-band LRs. The performance is substantially enhanced when 3 or 4 sub-bands are fused in contrast to using only the best single sub-band. As a matter of fact, the fusion of 3 or 4 optimal sub-bands brings the system to nearly the same performance level as that obtained with full-band CCs or even marginally better.

Note that 3 or 4 sub-bands are here represented with 9 or 12 BLCCs in total, respectively. Thus, only a few BLCCs are necessary to achieve nearly the same performance as that obtained with the 14 full-band CCs. This a notable advantage of BLCCs in terms of computational efficiency.

The performance stays basically unchanged even when more sub-bands are included for fusion, except for a slight deterioration in performance towards the higher numbers of fused sub-bands.

Together with the results from Experiment 1, the above observations based on Experiment 2 would seem to indicate that locally-encoded speaker information is not necessarily unique as per its spectral region. In other words, pieces of speaker information may be redundantly encoded across different spectral regions. Otherwise, the continuous decline in C_{llr} (an incessant gain in performance) should have been observed as more sub-bands are totalled for fusion.

The bottom plots in each panel clearly show that the three best-performing sub-bands span different spectral regions depending on the vowel. For /i/, they are in the mid- and high-frequency ranges above 2 kHz, which generally correspond to the spectral regions with strong speaker information. On the other hand, for /e/ and /a/, the three best sub-bands are dispersed in the low- and mid-frequency ranges below 3.0-3.4 kHz. For /u/ and /o/, the three best sub-bands are most widely separated in the range approximately between 0 and 4.2 kHz.

It is noticeable that the 3 best sub-bands are not only spaced apart from each other, but they also tend to fall in the speaker-sensitive spectral regions. This leads us to conjecture that those sub-bands are likely to contain more locally-distinctive speaker information. In support of the conjecture, it can be observed that the 3 worst sub-bands (coloured in pink) are in immediately neighbouring positions. For /u, e, a/, they are the 3 contiguous sub-bands appearing in the high-end of the spectrum and, for /i/, the 3 sub-bands flock together towards the low-frequency end, where C_{llr} values are worse. It can therefore be surmised that those sub-bands did not perform well after fusion because they are largely redundant in speaker information in addition to being less sensitive to speaker individuality, as demonstrated in Experiment 1.

Following on from Experiment 1, the importance of the F3 region for FVC is also evident for the back vowels from the bottom plots given in Fig. 4, in that one of the 3 sub-bands falls in the F3 region. The sub-band spanning F3 does not seemingly contain strong speaker information for vowel /a/; the C_{llr} values of the region are higher than the average C_{llr}. Nevertheless, the speaker information encoded in the F3 region is judged to be complementary with the sub-bands spanning F1 and F2 for /a/. For /u/ and /o/, one sub-band appears in the frequency range beyond F3, in agreement with Khodai-Joopari et al. (2004).

As can be seen from the bottom plots for the back vowels, the first sub-band [0, 0.6] kHz turned out to be a good one when fused with the other 2 sub-bands. Judging from the commonly shared empirical knowledge that more speaker-specific information is encoded in higher spectral regions (Hayakawa and Itakura 1994; Kitamura and Akagi 1995), this result is counter-intuitive. However, Khodai-Joopari et al. (2004) also sighted a peak of speaker F-ratio below F1 region for /o/ and /a/, and suggested their glottal-source characteristics as a possible cause for the peak.

6 Discussion

The FVC results presented in Sec. 5 confirm the existence of speaker-sensitive spectral regions, which principally agree with previous acoustic and articulatory studies of vowels. As such, it can demonstrably be argued that BLCC is a useful analytical tool equipped with flexibility and precision in selecting any sub-band of interest.

The formant frequencies (F1, F2 and F3) are common phonetic features in linguistic-phonetic FVC (Rose et al., 2003; Morrison 2008, Rose, 2017). The analytical potential of the multi-band BLCCs, however, unavoidably led us to notice that the regions corresponding to formant frequencies do not always contain strong speaker information. For example, the C_{llr} values for sub-bands spanning the F1-F2 region of /i/, the F2 region /o/ and the F3 region of /a/ are relatively high compared to the other regions. This suggests that sub-band selection based strictly on formant ranges is an unnecessarily constraining and even sub-optimal solution.

A case in point is Kinoshita et al's (2022) results based on sub-band cepstral distances and on prior knowledge of fixed F1, F2 and F3 sub-bands. A set of FVC experiments was done with the sub-bands that were selected according to the fixed F1, F2 and F3 ranges provided in Kinoshita et al. (2022) for the same experiments performed in the current study. The resultant C_{llr} values are shown in Table 2, together with the C_{llr} values with the 3 optimal sub-bands (fused) selected empirically (see Fig. 4), i.e., without prior acoustic-phonetic knowledge. The C_{llr} values for the full-band CCs are also listed.

The results from the 2 rightmost columns of Table 2 indicate that BLCCs can achieve nearly full-band performance with 3 optimal sub-bands and, thus, with fewer cepstral features. This finding illustrates the power of BLCCs in locating such sub-bands without any prior knowledge.

Vowels	Kinoshita et al (2022)	This Study	
	3 sub-bands (with prior knowledge)	3 sub-bands (without prior knowledge)	full band
/i/	0.52191	0.43142	0.40342
/u/	0.68992	0.60858	0.57934
/e/	0.54173	0.51947	0.48843
/o/	0.73500	0.66732	0.67500
/a/	0.65428	0.61239	0.58130
Ave.	0.62856	0.56783	0.54549

Table 2: Middle columns: C_{llr} values for 3 fused sub-bands selected using two approaches. Kinoshita et al's (2022) approach with prior knowledge, i.e., based on their formant ranges; and this study's approach without prior knowledge, i.e., guided by empirical selection. Rightmost column: full-band C_{llr} values from this study are included for reference.

It is relevant to point out that while our FVC experiments and Kinoshita et al's (2022) involve phonologically the same vowels and about the same number of speakers, their vowel tokens were produced in various consonantal contexts, whereas ours were produced without any such contexts. Thus, the exact formant ranges could be different for the vowels included in these two studies.

Notwithstanding this discrepancy for now, the trend of C_{llr} values in the 2 middle columns of Table 2 is consistent and encouraging: Our approach (without prior knowledge) outperforms the one employed by Kinoshita et al. (2022) (with prior knowledge). Further investigations with BLCCs applied to Kinoshita et al's vowel data and to other datasets will be necessary to confirm the apparent superiority of our sub-band approach in FVC.

The results obtained in this study are based only on male speech samples. While this is practically justified because males tend to commit crimes more often than females, further experimentation is desirable with a wider variety of speakers. However, the analytical power of BLCCs should remain unaffected by gender or age. It is the locations and ranges of speaker-sensitive spectral regions that could differ with these factors.

While retaining intrinsic properties of the cepstrum (e.g., ease of extraction, immunity to insignificant spectral details), the analytical power of BLCCs allows the forensic scientist to flexibly shift the focus of scrutiny and interpretation according to the selected sub-band region(s). This is an invaluable contribution that BLCCs can bring to the task of communicating the FVC outcome to the trier-of-fact in a more approachable way.

7 Future Work

The BLCCs exploited here are based on LP modelling of the speech signal and extracted on a linear frequency scale. However, there may be further insights to be gained by applying the same linear transformation to CCs from filter-bank outputs, combined with a nonlinear mapping of the frequency axis such as the often-used Mel scale. It is interesting to note that, except for /i/, our best-performing sub-bands include the lower-spectral regions that are precisely emphasised with Mel-Frequency CCs (MFCCs). A deeper investigation of MFCCs with differing sub-band widths and overlaps is therefore possible using our flexible approach to selecting local spectral regions.

From a forensic point of view, it is coherent to extend the application of BLCCs to non-vowel sounds (Rose 2022), whose speaker-sensitive spectral properties have received relatively less attention. From a linguistic point of view, it is conceivable that BLCCs could also be used as an ancillary or alternative parameter in the areas of acoustic-phonetics (e.g., efficient encoding of contrastive features as in Iskarous (2018)) and socio-phonetics (e.g., exploration of accent-specific sub-bands) (Arslan and Hansen 1997). In connection with these applications, it would be useful to study correlations between BLCCs and formant frequencies via the linear regression models developed by Broad and Clermont (1989) and Clermont (2013), and recently explored by Hughes et al. (2020) in the FVC context.

Finally, it is hoped that the sub-band approach embedded in BLCCs will bring new perspectives in other areas of speech science and technology, such as speech classification (Mokhtari and Clermont 1994), spoofing detection (Chettri et al. 2020; Soni et al. 2016), language identification (Salesky et al. 2021), and speech emotion recognition. Any pieces of information related to speaker variability, speech emotion, or synthesised speech, which are found to be notably encoded in specific sub-bands, would be advantageous for building robust classification systems, or for training deep-learning models. These technological pursuits are likely to benefit from the flexibility and efficiency afforded by the BLCC approach to sub-band spectral analysis.

Acknowledgments

The authors thank the four reviewers for their valuable comments.

the 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference: 1-4.

References

Aitken, C. G. G. and Lucy, D. (2004) Evaluation of trace evidence in the form of multivariate data. *Journal of the Royal Statistical Society, Series C (Applied Statistics) 53*(1): 109-122. https://dx.doi.org/10.1046/j.0035-9254.2003.05271.x

Arslan, L. M. and Hansen, J. H. L. (1997) A study of temporal features and frequency characteristics in American English foreign accent. *The Journal of the Acoustical Society of America 102*(1): 28-40. https://doi.org/10.1121/1.419608

Broad, D. J. and Clermont, F. (1989) Formant estimation by linear transformation of the LPC cepstrum. *The Journal of the Acoustical Society of America 86*(5): 2013-2017. https://doi.org/10.1121/1.398581

Chettri, B., Kinnunen, T. and Benetos, E. (2020) Subband modeling for spoofing detection in automatic speaker verification. *Proceedings of Odyssey 2020 The Speaker and Language Recognition Workshop*: 341-348.

Clermont, F. (2013) Cepstrum-to-formant mapping of spoken vowels. *Paper presented at the Conference of the International Association in IAFPA 2013 – 22nd Annual Conference of the International Association for Forensic Phonetics and Acoustics*, July 21-24, Tempa, Florida, 2013.

Clermont, F. (2022) Linear transformation from full-band to sub-band cepstrum. In R. Billington (ed.), *Proceedings of the 18th Australasian International Conference on Speech Science and Technology*: 136-140.

Goto, R., Misawa, K. and Okada, Y. (2017) Analysis of individual characteristics in vowel spectral envelopes. *Proceedings of the International MultiConference of Engineers and Computer Scientists*: 113-116.

Hayakawa, S. and Itakura, F. (1994) Text-dependent speaker recognition using the information in the higher frequency band. *Proceedings of the 1994 IEEE International Conference on Acoustics, Speech and Signal Processing*: I/137-I/140.

Hughes, V., Clermont, F. and Harrison, P. (2020) Correlating cepstra with formant frequencies: Implications for phonetically-informed forensic voice comparison. *Proceedings of Interspeech 2020*: 1858-1862.

Hyon, S., Wang, H., Wei, J. and Dang, J. (2012) An investigation of dependencies between frequency components and speaker characteristics based on phoneme mean F-ratio contribution. *Proceedings of the 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference*: 1-4.

Iskarous, k. (2018) The encoding of vowel features in mel-frequency cepstral coefficients. *Il parlato nel contesto naturale [Speech in the Natural Context]*: 9-18. https://dx.doi.org/10.17469/O2104AISV00000

Khodai-Joopari, M., Clermont, F. and Barlow, M. (2004) Speaker variability on a continuum of spectral sub-bands from 297-speakers' non-contemporaneous cepstra of Japanese vowels. *Proceedings of the 10th Australian International Conference on Speech Science and Technology*: 504-509.

Kinoshita, Y. (2001). *Testing Realistic Forensic Speaker Identification in Japanese: A Likelihood Ratio Based Approach Using Formants.* Unpublished PhD thesis, the Australian National University, Canberra.

Kinoshita, Y., Osanai, T. and Clermont, F. (2022) Sub-band cepstral distance as an alternative to formants: Quantitative evidence from a forensic comparison experiment. *Journal of Phonetics 94*: 101177. https://doi.org/10.1016/j.wocn.2022.101177

Kitamura, T. and Akagi, M. (1995) Speaker individualities in speech spectral envelopes. *Journal of the Acoustical Society of Japan (E) 16*(5): 283-289. https://doi.org/10.1250/ast.16.283

Mohammadi, S. H., Sameti, H., Tavanaei, A. and Soltani-Farani, A. (2011) Filter-bank design based on dependencies between frequency components and phoneme characteristics. *Proceedings of the 19th European Signal Processing Conference*: 2142-2145.

Mokhtari, P. and Clermont, F. (1994) Contributions of selected spectral regions to vowel classification accuracy. *Proceedings of The 3rd International Conference on Spoken Language processing*: 1923-1926.

Mokhtari, P. and Clermont, F. (1996) A methodology for investigating vowel-speaker interactions in the acoustic-phonetic domain. *Proceedings of the 6th Australian International Conference on Speech Science & Technology*: 127-132.

Morrison, G. S. (2013) Tutorial on logistic-regression calibration and fusion: Converting a score to a likelihood ratio. *Australian Journal of Forensic Sciences 45*(2): 173-197. https://dx.doi.org/10.1080/00450618.2012.733025

Osanai, T., Kinoshita, Y. and Clermont, F. (2018) Exploring sub-band cepstral distances for more robust speaker classification. *Proceedings of the 17th Australasian International Conference on Speech Science and Technology*: 41-44.

Osanai, T., Tanimoto, M., Kido, H. and Suzuki, T. (1995) Text-dependent speaker verification using isolated word utterances based on dynamic programming [in Japanese]. *National Research Institute for Police Science Report 48*(1): 15-19.

Peterson, G. E. (1959) The acoustics of speech–part II: Acoustical properties of speech waves. In L. E. Travis (ed.), *Handbook of Speech Pathology:* 1st Edition ed., 137-173. New York: Appleton-Century-Crofts, Inc.

Pols, L. C., Tromp, H. R. and Plomp, R. (1973) Frequency analysis of Dutch vowels from 50 male speakers. *The Journal of the Acoustical Society of America 53*(4): 1093-1101. https://doi.org/10.1121/1.1913429

Rabiner, L. R. and Juang, B. H. (1993) *Fundamentals of Speech Recognition* (1st ed.). Englewood Cliffs, N.J.: Prentice Hall.

Rose, P. (2022) Likelihood ratio-based forensic semi-automatic speaker identification with alveolar fricative spectra in a real-world case. In R. Billington (ed.), *Proceedings of the 18th Australasian International Conference on Speech Science and Technology*: 6-10.

Rose, P., Osanai, T. and Kinoshitaa, Y. (2003) Strength of forensic speaker identification evidence: multispeaker formant-and cepstrum-based segmental discrimination with a Bayesian likelihood ratio as threshold. *Forensic Linguistics 10*: 179-202. https://doi.org/10.1558/sll.2003.10.2.179

Saito, S. and Itakura, F. (1982) Personal characteristics of the frequency spectrum of vowels. *Annual Bulletin Research Institute of Logopedics and Phoniatrics 16*: 73-79.

Salesky, E., Abdullah, B. M., Mielke, S. J., Klyachko, E., Serikov, O., Ponti, E., … Vylomova, E. (2021) SIGTYP 2021 shared task: Robust spoken language identification. *Proceedings of the 3rd Workshop on Computational Typology and Multilingual NLP*: 122–129.

Sambur, M. (1975) Selection of acoustic features for speaker identification. *IEEE Transactions on Acoustics, Speech, and Signal Processing 23*(2): 176-182. https://dx.doi.org/10.1109/TASSP.1975.1162664

Soni, M. H., Patel, T. B. and Patil, H. A. (2016) Novel subband autoencoder features for detection of spoofed speech. *Proceedings of Interspeech 2016*: 1820-1824.

van den Heuvel, H., Cranen, B. and Rietveld, A. C. M. (1993) Speaker-Variability in Spectral Bands of Dutch Vowel Segments. *Proceedings of the 3rd European Conference on Speech Communication and Technology*: 635-638.

Wang, L., Wang, J., Li, L., Zheng, T. F. and Soong, F. K. (2016) Improving speaker verification performance against long-term speaker variability. *Speech Communication 79*: 14-29. http://dx.doi.org/10.1016/j.specom.2016.02.004

Right the docs: Characterising voice dataset documentation practices used in machine learning

Kathy Reid
School of Cybernetics
Australian National University
Canberra, Australia
kathy.reid@anu.edu.au

Elizabeth T. Williams
School of Cybernetics
Australian National University
Canberra, Australia
elizabeth.williams@anu.edu.au

Abstract

Voice-enabled technologies such as virtual assistants are quickly becoming ubiquitous. Their functionality relies on machine learning (ML) models that perform tasks such as automatic speech recognition (ASR). These models, in general, currently perform less accurately for some cohorts of speakers, across axes such as age, gender and accent; they are biased.

ML models are trained from large datasets. ML Practitioners (MLPs) are interested in addressing bias across the ML lifecycle, and they often use dataset documentation here to understand dataset characteristics. However, there is a lack of research centred on *voice* — spoken language — dataset documentation. Our work makes an empirical contribution to this gap, identifying shortcomings in voice dataset documents (VDD), and arguing for actions to improve them.

First, we undertake 13 interviews with MLPs who work with voice data, exploring how they use VDDs. We focus here on MLP roles and trade-offs made when working with VDDs. Drawing from the literature and from interview data, we create a rubric through which to analyse VDDs for nine voice datasets. Triangulating the two methods in our findings, we show that VDDs are inadequate for the needs of MLPs on several fronts. VDDs currently codify voice data characteristics in fragmented ways that make it difficult to compare and combine datasets, presenting a barrier to MLPs' bias reduction efforts.

We then seek to address these shortcomings and "right the docs" by proposing improvement actions aligned to our findings.

1 Introduction, motivation and previous work

Voice-enabled technologies, such as virtual assistants and smart speakers, are "going to scale" through axes such as volume (Kinsella and Mutchler, 2020; Bradley, 2020; Van der Meulen and Forni, 2016), geographies (Popović et al., 2015; Jones, 2020; Kendall et al., 2020), miniaturisation (Bouraoui et al., 2017), expanding use cases (Dale, 2020; Brewer et al., 2022; Jesús-Azabal et al., 2019) and use in multiple modalities (Baevski et al., 2022). Speech technology has become part of the fabric of modern *information infrastructures* — the technical capabilities, social norms, organisational practices and economic mechanisms (Bowker et al., 2009; Turow, 2021) — that collectively allow us to speak with machines and have them do our bidding. As voice technology becomes ubiquitous, so too does the potential societal impact of its bias. A person's poor voice interaction experience is no longer confined to a virtual assistant in the home, or to a mobile phone, but extends to the workplace, the car, healthcare, and customer service settings.

These systems use machine learning (ML)-enabled components like automatic speech recognition (ASR). However, they don't yet work well for everyone (Liu et al., 2022; Ngueajio and Washington, 2022; Feng et al., 2021). They exhibit bias — defined here as systematic and unfair discrimination against individuals or cohorts of individuals in favour of others (Friedman and Nissenbaum, 1996) [1] — across axes such as age (Vipperla et al., 2010; Gerosa et al., 2007), gender (Tatman, 2017; Tatman and Kasten, 2017; Garnerin et al., 2020), race (Koenecke et al., 2020), nationality (Hutiri and Ding, 2022), and accent (Hinsvark et al., 2021). Dataset documentation is a frequent tool used by MLPs to mitigate bias.

1.1 Dataset documentation and its use by MLPs

The ML-enabled components in voice-enabled technologies require large datasets to be effective.

---

[1] We recognise that bias manifests in many ways and has several interpretations, and suggest (Barocas et al., 2019) for a more complete treatment.

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 51–66
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

Dataset documentation — descriptive information characterising the nature, contents and provenance of a dataset — affords MLPs a clearer understanding of a dataset's characteristics before the dataset is used as an *input* to an ML model. This allows the detection of some forms of bias, such as under-representation of speakers having specific characteristics. In contrast, *model documentation* — descriptive information characterising the performance of a trained ML model against evaluative criteria — focuses on the performance *output* of ML processes. It provides MLPs the opportunity to detect and remediate bias issues such as poor inference accuracy for specific types of speech. Both types of documentation are well established in the literature as tools to detect and prevent bias in ML.

Bender and Friedman (2018) introduce *Data statements* for natural language processing (NLP), where they propose collecting information such as speaker demographics, annotator demographics, and the domain and context of the material as a way to address bias in written text corpora. Gebru et al. (2021) brings data provenance to the forefront of broader ML practice by outlining key areas MLPs should consider, such as the purpose and intended use of the dataset, the objects it stores, how they're represented, the relationships between them, sources of error and noise, sensitivity and identification considerations, how the data was collected and labelled, and how the datasets are distributed and maintained. Boyd (2021) seeks to empirically validate the utility of datasheets, and demonstrates their benefit by having MLPs ethically reflect on problematic datasets — directly connecting datasheets as an artifact with improved practice. From the field of computer vision, Miceli et al. (2021) also focus on praxis, emphasising the need for practitioner reflexivity in the production of ML datasets. Similarly, in an effort to make the ethical considerations and choices made during the production of datasets produced through crowd-sourced annotations more transparent, Díaz et al. (2022) develop the *CrowdWorkSheets* framework. McMillan-Major et al. (2023) focus on adoption of dataset documentation, working with NLP practitioners to increase uptake.

In Costa-jussà et al. (2020), we see the adaptation of data statements and datasheets for datasets from NLP to other written language technologies — in this case — machine translation. Bandy and Vincent (2021) tie dataset documentation to the concept of technical debt, and retrospectively produce a datasheet for a text corpus. Pushkarna et al. (2022), based on their work with text corpora at Google, then introduce the concept of *data cards*, concentrating on descriptive information that cannot be inferred from the dataset itself. Building on this work, and drawing from an extensive literature review, Papakyriakopoulos et al. (2023) propose augmented datasheets specifically for *spoken language* datasets — the only one of its kind to date.

Similarly, there has been increasing research attention toward model documentation. *Model cards* were first introduced by Mitchell et al. (2019) and built on by Shen et al. (2022), who produced a practitioner toolkit to aid in generic model card development. Crisan et al. (2022), recognising that many laypeople also use model documentation, develop an interactive approach to aid in model exploration. McMillan-Major et al. (2021) seek to join both datasheets and model cards, proposing a standard format for datasets in NLP.

However, data and model documentation in itself is not sufficient for tackling bias. An MLP creates or consumes that documentation, providing a *feedback loop* which motivates MLP *action*: re-balancing a training set, gathering more diverse data, or fine-tuning a model.

Accordingly, recent work from Microsoft Research shifts the focus of inquiry to practitioners' use of dataset and model documentation and approaches to fairness more broadly. Heger et al. (2022) find that dataset documentation practices are "largely ad-hoc and myopic in nature", with many practitioner needs unaddressed. Similarly, Holstein et al. (2019) find, in a set of interviews with MLPs in industry, that while they saw the datasets as "the most important place to intervene to improve fairness in their products", the teams did not have in place processes — such as dataset documentation — "to help support the collection and curation of balanced or representative datasets".

1.2 The research gap

People are increasingly using speech to interface with services and sources of support in the real world. ML-enabled voice technology systems continue to have pronounced biases; they work better for some people than others. If we wish to make the socio-technical systems of our world fairer, then we need to generate effective approaches for tackling bias in these systems. The approaches, motivations and actions of MLPs around dataset documentation have been shown to assist in this regard. However, there is a lack of research here covering *spoken* language data — the kind of data used to build voice technology systems.

We therefore pose the following research provocations: How may we characterise current VDD artefacts and practices? And what work is needed to make VDDs more useful in addressing bias in voice technologies?

2 Methodology

We devise an exploratory study that combines two methods, one focusing on ML practitioners and their *experiences* creating or consuming voice dataset documentation and the other on dataset documentation *artefacts*.

Firstly, we undertake 13 semi-structured interviews with MLPs who work with voice or closely adjacent data. We explore their voice dataset document (VDD) approaches across the ML lifecycle. Secondly, we turn our attention to existing VDDs. VDDs represent how MLPs generate datasets and release them to the world — they encode practices, beliefs and assumptions (Birhane et al., 2022). We select nine VDDs for their varied purposes, collection methods and source data.

Drawing both from our literature review in Section 1.1 and from participant data, we develop a rubric for analysis, and assess the VDD artefacts across seven categories. We then triangulate the two methods, showing how VDD practices differ by MLP role, and how VDDs may help or hinder MLPs in making trade-off decisions.

2.1 Semi-structured interviews

Semi-structured interviews are established as an appropriate exploratory method for inquiring about phenomena, particularly in ML practice (Baier et al., 2019; Jöhnk et al., 2021; Følstad et al., 2018).

2.1.1 Participant selection

Potential participants were identified using professional networks, snowball sampling, and via collaborative code sites. Inclusion criteria were (i) that the participant must work with voice or closely adjacent data, and (ii) be currently practicing in industry, academia or open source fields. Purposive sampling was used to ensure representation of perspectives from diverse genders, professional disciplines, and geographic locations, and to help establish trustworthiness of findings (Campbell et al., 2020; Lincoln and Guba, 1985; Groves et al., 2011; Ezzy, 2013). A summary of participants by characteristic is shown in Appendix A.

Interviews were conducted via video-conferencing, and participants were able to make corrections and redactions to the resulting transcript. We concluded our interviews at 13 participants as themes were becoming repetitive, and we had sufficient data to inform our document analysis method.

2.1.2 Semi-structured interview design

We adopted an inductive approach, seeking to accumulate many perspectives around how VDDs are produced and consumed, whilst varying their contexts, applications and geographic sites of practice (Creswell and Creswell, 2018). Drawing from both Spradley (1979) and Minichiello et al. (1990), we structured our interview questions around "the lifecycle of creating a voice dataset" — a "grand tour" approach.

2.1.3 Coding approach

Based on our literature review, we identified several *a priori* categories (Saldaña, 2021) and used them to code the 13 interviews. We combined this with open coding — a way to capture new categories as they emerge in the data (Williams and Moser, 2019). Axial coding — a way to frame the contextual conditions of the phenomena being studied (Ezzy, 2013) — was then used to categorise how VDDs were produced and consumed. Selective coding — a way to collapse and combine several codes into core categories for analysis (Corbin and Strauss, 1990) — was then applied, yielding 14 broad categorisations across a total of 1889 codes. Here, we focus on only two of those broad categories; different MLP *roles* involved in VDDs, and how VDDs are used in the *trade-offs* MLPs make.

2.2 Document analysis

As a complementary method to our semi-structured interviews, we then undertook document analysis — "a systematic procedure for reviewing or evaluating documents" (Bowen, 2009).

2.2.1 Selection of documents

Datasets used for ML are often released with accompanying documentation in the form of a dedicated web site, code repository or online catalogue entry. Additionally, some datasets contain a metadata file *within* the dataset. We considered all of these in scope for analysis.

To identify VDDs for analysis, we performed a web search, using the terms “voice dataset” or “speech dataset”. We purposively sampled nine datasets that varied by intended task; by whether the speech was elicited or spontaneous; the domain of speech; the curation rationale; funding source; license; and vocabulary size [2]. A summary is provided in Appendix B.

2.2.2 Document analysis rubric

To create the rubric used to analyse the VDD artefacts, we drew from previous work in dataset documentation (see Section 1), broader reading in metadata and research infrastructure, and participant data, arriving at 41 elements across seven categories. Here, we outline the contents of each category and justify their inclusion in the rubric.

Dataset identification Here we included persistent identifier — a uniquely identifying string, separate from the location of the dataset itself, which provides a referral to the current storage location of the dataset (Zeng and Qin, 2016) — and version as a way to distinguish dataset releases over time (Bhattacherjee et al., 2015). Efforts have been long underway to ensure datasets have persistent identifiers (Klump and Huber, 2017), and they tie closely to work on making research datasets more findable (Wilkinson et al., 2016).

Intent, purpose and curation rationale Here, we draw on the definition given by Schlangen (2021); a language task is a mapping between an input and an output, and a dataset provides examples of this mapping. Clear descriptions of intent and purpose are therefore important so the MLP can identify if the dataset is task-appropriate. We adapt “curation rationale” as given in Bender and Friedman (2018) to *spoken* language, and define it as determining which speech utterances are included in the dataset, and why.

Dataset creation process, sources and actors Here, we draw again from Bender and Friedman (2018), who place emphasis on understanding the social standpoint of annotators. For many speech tasks, written transcriptions are also required as inputs. Noting the work of Bucholtz (2007, 2000) — that transcription has both variation and politics in its production — we also identified whether the transcription method was provided. Referencing Barbiers et al. (2007) work on spoken language variation from corpus linguistics, we also included the source of elicited speech prompts as an element.

Characteristics of the dataset itself Here, we drew on from material on research data infrastructure. Working with “big data” presents many challenges to MLPs (Kitchin, 2014); and so it is beneficial to provide an overview of the size, shape and constituency of the dataset.

Constitution of the dataset by speaker, recording environment and spoken language attributes In our exploratory interviews (see 3), comprehending contents was a key consideration for many participants. Speech recognition requires a wide variety of voice samples, while speech synthesis needs many samples from a single speaker. It is therefore important that characteristics of the speech utterances captured in the data are clearly represented:

> “...Sometimes you really need to dig deeply into the corpus to find it. Sometimes you just don’t find it. And sometimes this is well documented. ... This is important ... because we need to have a balanced corpus for training your system. And then also to be able to evaluate, gender wise, the performance of your system.” — *SB*

We drew both from the literature and from exploratory interviews to identify specific attributes to assess. Bender (2019) makes the case for clearly identifying the languages we work with in, and Bender and Friedman (2018) advocate both for representing the languages in a dataset in `BCP-47` format *and* providing a “prose description” of the language’s “axes of variation”.

Participant *TS* highlighted additional areas of spoken language variance to scrutinise when evaluating trained models: “... We have a lot of folks who have code-switched data ... it’s also domain variation or register variation, or all your training data is super formal ...” —*TS*.

Code-switching is where the speaker alternates between two or more “codes” — usually languages — within a conversation (Auer, 2013). The domain of spoken language is usually taken to be the subject matter of the conversation, while register is how spoken language varies by social situation; we speak differently in formal and informal settings (Finegan, 2014).

Models, benchmarks and academic papers We adapt this category from Gebru et al. (2021), who

---

[2] We note here that the AusTalk dataset in the ALVEO repository is currently offline; had it been available we would have also included it due to its focus on Australian speech.

recommend documenting where a dataset should *not* be used, as also echoed by an interview participant: "When you think about kind of building a dataset, it's easy to think about, 'Okay, I'm building a dataset, it's going to be used for this. This is what I want it to be used for.' Unfortunately, people are going to use it for things you didn't intend." *—CG*.

Similarly, noting increasing calls for benchmarks to be tightly linked to the intended task of a dataset (Raji et al., 2021), we included these as an attribute in the analysis.

Privacy, bias, limitations and social impact Here, we drew from Bender and Friedman (2018); Gebru et al. (2021); Papakyriakopoulos et al. (2023), who all underscore the importance of documenting privacy and sensitivity considerations of a dataset, and their potential social consequences, and we use this category to assess whether biases and limits of the dataset are considered in VDDs.

2.2.3 Performing the analysis

To perform the analysis, we reviewed each dataset's documents against the criteria in each category of the rubric. If fulfilment of a criterion was implied but not explicit in the document(s), then we made a finding of "Implied" and provided a rationale. If a criterion was not applicable to a dataset, we made a finding of "N/A" — for example, in speech synthesis datasets like "LJSpeech", speech samples are usually taken from only one speaker and so the number of unique speakers in the dataset is not applicable. Our analysis is summarised in Appendix C.

3 Findings

Here, we triangulate our two methods. We characterise the experience of MLPs with VDDs through the frames of MLP roles and trade-offs the MLP makes, quoting from interview transcripts to highlight key points. At the same time, we corroborate the interview findings by referencing results from the document analysis. This layered approach provides a richer characterisation of VDDs.

3.1 Characterising practices by role

Our interview data showed that MLPs could be categorised into four distinct roles, depending on how they discovered, commissioned, produced or consumed voice datasets. We use a "food" analogy to label the roles — which seems odd at first glance — but which we believe accurately characterises a role's relationship with voice datasets. The results of the document analysis had different implications for each role, which we unpack below.

Chefs We characterise as *Chefs* those MLPs who are provided with a dataset specification against which to create a voice dataset: "... we would have a data collection spec, [with a] percentage of different accents or gender or whatever." *—BP*. *Chefs* are mostly likely to be *producers* of VDDs.

Diners *Diners* form a complement to *Chefs*, being the MLPs who are in a position to order voice datasets from commercial companies. These companies offer both bespoke options — à la carte — as well as subscriptions to regular dataset updates — a grocery box. There are many such providers: "So there are many companies that offer services in terms of annotating data, transcribing data. There are many companies that collect some data and sell data." *—SS*.

Scavengers Alternatively, an MLP may be a *Scavenger* — where they must discover freely available voice datasets to meet their needs due to cost constraints. "... us open source folks we're scavengers, right? ... The ordering options are there ... and I've looked at them and they want tens of thousands of dollars, for access. And I'm like, '"I don't have that."'" *—PS*.

Importantly, it was this remark that helped us arrive at our role categorisation.

Hoarders *Hoarders*, in contrast to Scavengers, Chefs and Diners, do not have a clear intent in mind for the voice data they accumulate; they store it for some future, unspecified purpose in the hope that it will be of use. Voice data accumulated this way is usually a byproduct of business operations: "We know that often companies, they have a plan to extract and collect as much data as possible before they even know what it's potentially useful for."— *PP*.

3.2 The focus of VDD practices differs by role

Discovery For the *Scavenger*, dataset documentation is important to their discovery efforts — and their ability to comprehend the contents of a dataset when found. Based on our document analysis, their needs are currently poorly served. While eight of the nine datasets represented speaker gender, only two represented accent, and only one represented speaker nationality or age. Speaker occupation, language heritage or education attainment were absent, save for an overview of speaker occupation in the *African languages* VDD. There was very little information provided on the recording environments used, and the only representation of variance of spoken language tended to be the way in which the dataset language(s) were specified — with five of the nine VDDs representing language using a `BCP-47` or `ISO-639` code.

Representation *Chefs* may produce documentation as part of their creation efforts, and in doing so, must make choices about how to represent that data. With both an absence of agreed or *de facto* standards for documenting voice and speech data[3], as well as multiple standards for language representation (Wright, 2019), some participants faced challenges in determining *how* some data items should be reported: "There is no unified format. Everybody has their own `JSON` [4] that might have similar information."*—BP*.

Another *Chef* practitioner faced similar data representation dilemmas in regard to dialect, grappling with what level of granularity to represent in the VDD: "...what if we label what dialect they are speaking in? Or what if they self label what dialect they think they are speaking in? Then we do things like how about we review this? Meaning let's write whether we think this is pronounced correctly. It's either yes or no. Okay. Wait, what if we can label every single character in the sentence and say whether the character was pronounced correctly?" *—EG*

VDDs are still relevant for the *Hoarder* role, even though they may not yet know what tasks their datasets will be used to perform. Hoarders still wrestle with how to represent the data they are collecting. Here, there was a desire to create VDDs that allowed the broadest scope of future use for the data:

"...it's always good to document, to label your data to the maximum extent that you can in terms of fidelity." *—RW*.

The desire to chronicle datasets with high fidelity places additional onus on the MLP to define *how* the data is represented. We see here a tendency to reproduce that which has come before: "...we didn't put a lot of thought into the choosing of the structure of the [Dataset] dataset, because we just used it as it was. And the reason that we chose the [Dataset] as an example dataset was because it was a fairly common, well-known speech recognition dataset"*—RW*.

This effect serves as a reinforcing loop, anchoring practice to the status quo.

Diversity of data Both *Scavengers* and *Diners* need to know whether the data within a voice dataset is useful for their intended purpose: "...the dataset documentation would give me an idea, does this dataset work for my application? ... Is this dataset going to be useful?" *—CG*.

Drawing from our document analysis, it appears *Scavengers* and *Diners* are well served by current VDDs — all nine datasets examined provided an executive summary or description, and eight of the nine provided both intended tasks or use cases, as well as a curation rationale.

However, even if a dataset appears to meet an MLP's need based on the contents of the VDD, variation in how the dataset is transcribed can be problematic, requiring that the MLP spend time "listening to the data": "...All transcription is subjective. And so each of these databases will have been transcribed by different people, maybe following different conventions, and those conventions are especially important with semi words, ums and uhs and mm-mms, and stuff like that." *—BP*.

Cross-referencing our participant's statement with our document analysis, we note that only one of the three datasets that had transcribed spontaneous speech provided a description of the transcription process.

Another salient example here deals with the lack of variation of accents in the dataset not being apparent from the VDD, a realisation the practitioner makes only *after* listening to the data, and having to cross-check with the dataset's related academic paper:

"...I had worked with it for a while, I thought I knew the data. It was a very popular dataset. And it wasn't until I started listening to it, that I realized that these are only North American voices. It wasn't obvious to me until then. And then I went back and I read the paper, the actual paper ... and it was explicit like, yes, they chose voices that were North American. And it's something simple as that, you don't know until you start listening to the data." *—CD*

---

[3] We note here the work of Papakyriakopoulos et al. (2023), however this was not available at the time the interviews were conducted.

[4] `JSON` is a data structure format commonly used for voice data

Again, cross-referencing, only two of the nine datasets provided a representation of the speaker's accent.

3.3 Characterising practices through trade-offs the practitioner must make

Changing our analytical lens, we now explore practices by exploring the trade-offs a practitioner must make. Drawing from the field of social learning theory, Wenger-Trayner and Wenger-Trayner (2014) hold that practitioners operate across multiple disciplinary communities in a "landscape of practice". An MLP may need to span disciplines such as data engineering, machine learning, metadata specification and linguistics; each with their own accepted practices (e.g. Deng et al. (2022); Balayn et al. (2021)). These practices may be in tension, requiring the practitioner to make trade-offs. While our interview data uncovered many trade-offs, we focus here on the most frequently recurring.

3.3.1 Big data vs storage

"The problem is, data gets big. And then you have a problem, right?" —*AG*

Speech technologies may require thousands of hours of data, in turn requiring large volumes of disk storage capacity. For example, one dataset we analysed, Mozilla Common Voice, is nearly 80GB in size. This scale causes practical problems for MLPs, such as one *Chef* who created voice datasets, and needed to store them on a server. His frustration at having to frequently move datasets was palpable: "Yeah. We would find somewhere on [University web server] we'd be, 'Oh yeah. No, we'll serve it off our little file server here and it'll be no worries.' And we'd put it up there and we'd create a website for it. And we'd point people at the website. And then the IT guys would go, 'Oh yeah, no. We don't want to do that [...] We're going to shut that down. You're going to have to find somewhere else to put that.'" —*RW*.

One mechanism that exists to overcome this limitation is the use of a *persistent identifier*. In our document analysis, only three of the nine datasets were found to have persistent identifiers applied (see C *Dataset identification*), and these were verified using Crossref [5]. More positively, all datasets bar one indicated storage requirements, and all provided the number of hours of overall speech in the dataset (see C *Characteristics of the dataset itself*).

3.3.2 Big data vs understanding data contents

We also identified trade-offs that the MLP had to make in comprehending the contents of a voice dataset. Earlier, in 3.2, we showed that an MLP compensated for lack of variation description in VDD by "listening to the data". The size of voice datasets makes this practice more onerous, as highlighted by one interview participant: "We ended up with 12,000 recordings, which was humanly transcribed and those 12,000 recordings equated to 20 hours of speech. So we literally had a team of people listening to recordings and typing the recordings out verbatim." —*SS*.

This again points to the need for more focus on capturing data related to recording environment in particular: "And with a hundred thousand hours of data, how are you going to listen to all that as one person especially? You can't. You can randomly sample and hope for the best that you catch something. But if you precisely knew exactly the conditions of the recordings and all that stuff, if you could control all that then I think you could do a much better job." —*PS*.

Triangulating this with our document analysis (see C *How the dataset represents the recording environment*), we find that only the CHIME-5 dataset provided explicit information on the recording environment. This is likely due to its relevance in the dataset's purpose of speech separation. Other datasets implied some recording information — such as the HUB5 dataset being of recorded telephone conversations.

Again, we find that VDDs are inadequate for MLPs' needs.

4 Righting the docs: Towards VDD that help MLPs mitigate bias in speech technologies

Drawing from the gaps in VDD practice uncovered from our exploratory study above, we now propose a program of work to begin to address them.

4.1 A unified description format for spoken language datasets

The VDDs we analysed contained a patchwork of information in varying formats. This presents hurdles for dataset consumers, such as *Scavengers* and *Hoarders*, in understanding dataset contents, as corroborated in 3.2. This is a necessary step before datasets can be effectively combined for training ML models. A unified datasheet format for spo-

---

[5] https://search.crossref.org/

ken language datasets is likely to go some way to addressing this weakness. Here, we welcome the work of Papakyriakopoulos et al. (2023) in formulating *Augmented datasheets for speech data*. This work provides both a minimal description structure, and tools to enable the dataset producer to create it. However, this alone is insufficient to address the challenges we uncovered.

4.2 Automating the creation of descriptive information for voice data

Augmented datasheets for speech data assumes that dataset producers act reflexively *before or during* the dataset creation process. Indeed, reflexivity has been shown to improve dataset practice (Boyd, 2021). We found some evidence of reflexivity in our interviews, with *Chefs* considering how to represent data items (see 3.2). However, given the lack of descriptive information found in many of the nine datasets analysed, it is reasonable to claim that much VDD work happens *after the fact*, if at all.

Here, classification models, such as for gender, age and accent, are needed to help provide better descriptive information for speech datasets, reducing the need for the MLP to "listen to the data" (see 3.2). This would be particularly helpful for datasets where granular VDD was not captured at the point of creation, providing the ability to create parts of VDD retrospectively — although we acknowledge that inferred VDD are likely to represent dataset contents less accurately.

There is some emerging work in this space, such as Sánchez-Hevia et al. (2022), who use a range of neural models to accurately predict gender and age on the Common Voice dataset, and Najafian and Russell (2020), who use automatic accent identification to make a model more robust to accented speech. We note, however, that such classification can be used for ethically dubious purposes, such as pre-emptive policing (e.g. such as that recently done in the Türkçe language (Korkmaz and Boyacı, 2022)). We also note that Gebru et al. (2021) caution *against* automating the creation of dataset documentation, championing instead the use of reflexive processes. We hold that there is a practical middle ground here; to be reflexive during dataset creation, but to have tools available when VDDs of existing datasets are insufficient.

4.3 Common representation taxonomies for voice data

In section 3.2, several participants highlighted the lack of consistency in formats used for representing variance in speaker characteristics, context of speech and the spoken language itself. Here, common taxonomies would assist MLPs in combining datasets in ways that aid in addressing bias. For example, MLPs may wish to compile spoken language data of a particular accent to assess if a neural model performs well on that accent. However, if different datasets represent accents in different ways, combining datasets becomes much harder. Indeed, the need to capture speaker demographics in particular more systematically was highlighted in our interviews:

"I would say that each time a new speaker is registered to the system, is going to start making a recording, we should have a nice interface, an easy to use interface, to quickly fill all the information that we need." —*SB*.

Although there is some recent work in the accent space, such as calls to extend the `BCP-47` format to better represent low-resource languages (Gillis-Webber and Tittel, 2019, 2020), and work to represent gender bias more accurately in text corpora (Havens et al., 2022), we still lack accepted taxonomies for representing the linguistic heritage of a speaker (language acquisition, L1 and L2 status etc), domains of speech (such as medical, quick service restaurant ordering, industrial automation) and the recording environment (such as cafe, quiet office, family home, studio). Having such reusable and inter-operable taxonomies would also align with efforts to make research data, and speech archives specifically, more "FAIR" (Wilkinson et al., 2016; Calamai and Frontini, 2018).

4.4 Incentivising adoption of unified formats

Even if unified description formats and common taxonomies for VDDs are available, a mechanism is needed to incentivise their *adoption*, particularly given the practice identified in our interview data of replicating existing dataset formats (see 3.2). Bender and Friedman (2018) outline several incentives which would be useful here, such as requiring adherence to dataset documentation formats for publication in key journals.

With increasing usage of collaborative coding platforms in ML practice (Berman, 2023), another available incentive is to require complete VDDs before datasets are uploaded. For example, while Hugging Face displays dataset datasheets on the platform, there is no requirement for them to be completed, and they are often blank [6].

5 Limitations

Additional methods to triangulate findings We recognise the small, although purposive, sample of participants and datasets in our exploratory study. We now intend to administer a questionnaire to a broader group of MLPs, to validate or invalidate these initial findings.

Only publicly knowable datasets were analysed In identifying and selecting datasets for analysis, we recognise that our approach was limited to only publicly knowable datasets; private and/or proprietary datasets used internally by organisations may exhibit very different dataset documentation practices, although this is unlikely based on the work of Heger et al. (2022) and Holstein et al. (2019).

6 Conclusion

Here, we have situated voice dataset documentation (VDD) practices conducted by machine learning practitioners (MLPs) within broader efforts to reduce bias in ML-enabled speech technologies as they go to scale. We first provided a brief literature review of ML-related dataset documentation work, identifying that VDD practices are understudied. We presented an exploratory study that combined two methods — semi-structured interviews and document analysis — to provide a rich characterisation of practices surrounding VDDs.

We find that VDDs are currently inadequate to meet the needs of MLPs who create and consume voice datasets. In particular, they often fail to describe voice dataset contents accurately, if at all, and the range of representation formats used makes it difficult for MLPs to combine datasets effectively — as is often required in bias reduction efforts.

Drawing from these findings, we propose actions that seek to "right the docs", focusing on unified formats for dataset documentation, as well as the need for common taxonomies for data items common to voice datasets.

7 Ethics statement

The interviews conducted as part of this research have received human ethics approval from Australian National University's Human Research Ethics Committee, protocol number `417/2021`. This protocol requires using pseudonyms to refer to research participants, which has been done here.

Acknowledgements

Thanks are extended to Glen Berman for feedback on earlier versions of this paper. Kathy Reid's PhD research is funded by an Australian Research Training Scholarship and via the Florence Violet MacKenzie Scholarship. Kathy Reid holds a Research Partnership with Mozilla Foundation however this Partnership does not provide research funding. Kathy is grateful to the interview participants who gave generously of their time, knowledge and experience, and to the School of Cybernetics PhD cohort for their support. We thank the reviewers for their feedback.

References

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M Tyers, and Gregor Weber. 2020. Common Voice: A Massively-Multilingual Speech Corpus. In *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, pages 4218–4222, Marseille, France. European Language Resources Association (ELRA).

Peter Auer. 2013. *Code-Switching in Conversation: Language, Interaction and Identity*. Routledge.

Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. 2022. Data2vec: A general framework for self-supervised learning in speech, vision and language. In *International Conference on Machine Learning*, pages 1298–1312. PMLR.

Lucas Baier, Fabian Jöhren, and Stefan Seebacher. 2019. Challenges in the deployment and operation of machine learning in practice. In *ECIS*, volume 1.

Agathe Balayn, Christoph Lofi, and Geert-Jan Houben. 2021. Managing bias and unfairness in data for decision support: A survey of machine learning and data engineering approaches to identify and mitigate bias and unfairness within data management and analytics systems. *The VLDB Journal*, 30(5):739–768.

John Bandy and Nicholas Vincent. 2021. Addressing "Documentation Debt" in machine learning: A retrospective datasheet for BookCorpus. In *Proceedings*

---

[6] For example, the datasheet for Common Voice on Hugging Face omits large sections, such as curation rationale and limitations

of the Neural Information Processing Systems Track on Datasets and Benchmarks, volume 1. Curran.

Sjef Barbiers, Leonie Cornips, and Jan Pieter Kunst. 2007. The Syntactic Atlas of the Dutch Dialects (SAND): A corpus of elicited speech and text as an online dynamic atlas. *Creating and Digitizing Language Corpora: Volume 1: Synchronic Databases*, pages 54–90.

Jon Barker, Shinji Watanabe, Emmanuel Vincent, and Jan Trmal. 2018. The fifth 'CHiME' Speech separation and recognition challenge: Dataset, task and baselines. In *Interspeech 2018-19th Annual Conference of the International Speech Communication Association*.

Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2019. *Fairness and Machine Learning: Limitations and Opportunities*. fairmlbook.org.

Emily M Bender. 2019. The# benderrule: On naming the languages we study and why it matters. https://thegradient.pub/the-benderrule-on-naming-the-languages-we-study-and-why-it-matters/.

Emily M Bender and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604.

Glen Berman. 2023. Machine Learning practices and infrastructures. In *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*, pages 466–481.

Souvik Bhattacherjee, Amit Chavan, Silu Huang, Amol Deshpande, and Aditya Parameswaran. 2015. Principles of dataset versioning: Exploring the recreation/storage tradeoff. In *Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases*, volume 8, page 1346. NIH Public Access.

Abeba Birhane, Pratyusha Kalluri, Dallas Card, William Agnew, Ravit Dotan, and Michelle Bao. 2022. The values encoded in machine learning research. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 173–184.

Hasna Bouraoui, Chadlia Jerad, Anupam Chattopadhyay, and Nejib Ben Hadj-Alouane. 2017. Hardware architectures for embedded speaker recognition applications: A survey. *ACM Transactions on Embedded Computing Systems (TECS)*, 16(3):1–28.

Glenn A Bowen. 2009. Document analysis as a qualitative research method. *Qualitative research journal*, 9(2):27–40.

Geoffrey C Bowker, Karen Baker, Florence Millerand, and David Ribes. 2009. Toward information infrastructure studies: Ways of knowing in a networked environment. In *International Handbook of Internet Research*, pages 97–117. Springer.

Karen L Boyd. 2021. Datasheets for datasets help ML engineers notice and understand ethical issues in training data. *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW2):1–27.

Anthony J. Bradley. 2020. Brace Yourself for an Explosion of Virtual Assistants. https://blogs.gartner.com/anthony_bradley/2020/08/10/brace-yourself-for-an-explosion-of-virtual-assistants/.

Robin Brewer, Casey Pierce, Pooja Upadhyay, and Leeseul Park. 2022. An empirical study of older adult's voice assistant use for health information seeking. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 12(2):1–32.

Mary Bucholtz. 2000. The politics of transcription. *Journal of Pragmatics*, 32(10):1439–1465.

Mary Bucholtz. 2007. Variation in transcription. *Discourse studies*, 9(6):784–808.

Silvia Calamai and Francesca Frontini. 2018. FAIR data principles and their application to speech and oral archives. *Journal of New Music Research*, 47(4):339–354.

Steve Campbell, Melanie Greenwood, Sarah Prior, Toniele Shearer, Kerrie Walkem, Sarah Young, Danielle Bywaters, and Kim Walker. 2020. Purposive sampling: Complex or simple? Research case examples. *Journal of research in Nursing*, 25(8):652–661.

Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. 2018. VoxCeleb2: Deep speaker recognition. In *Proc. Interspeech 2018*, pages 1086–1090.

Juliet M Corbin and Anselm Strauss. 1990. Grounded theory research: Procedures, canons, and evaluative criteria. *Qualitative sociology*, 13(1):3–21.

Marta R Costa-jussà, Roger Creus, Oriol Domingo, Albert Domínguez, Miquel Escobar, Cayetana López, Marina Garcia, and Margarita Geleta. 2020. MT-adapted datasheets for datasets: Template and repository. *arXiv preprint arXiv:2005.13156*.

John W Creswell and J David Creswell. 2018. *Research Design*, 5th edition. Sage publications Thousand Oaks, CA.

Anamaria Crisan, Margaret Drouhard, Jesse Vig, and Nazneen Rajani. 2022. Interactive model cards: A human-centered approach to model documentation. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 427–439.

Robert Dale. 2020. Voice assistance in 2019. *Natural Language Engineering*, 26(1):129–136.

Wesley Hanwen Deng, Manish Nagireddy, Michelle Seng Ah Lee, Jatinder Singh, Zhiwei Steven Wu, Kenneth Holstein, and Haiyi Zhu. 2022. Exploring how machine learning practitioners (try to) use fairness toolkits. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 473–484.

Mark Díaz, Ian Kivlichan, Rachel Rosen, Dylan Baker, Razvan Amironesei, Vinodkumar Prabhakaran, and Emily Denton. 2022. Crowdworksheets: Accounting for individual and collective identities underlying crowdsourced dataset annotation. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 2342–2351.

Douglas Ezzy. 2013. Coding data and interpreting text: Methods of analysis. In *Qualitative Analysis*, pages 80–110. Routledge.

Siyuan Feng, Olya Kudina, Bence Mark Halpern, and Odette Scharenborg. 2021. Quantifying bias in automatic speech recognition. *arXiv preprint arXiv:2103.15122*.

Edward Finegan. 2014. *Language: Its Structure and Use*. Cengage Learning.

Asbjørn Følstad, Cecilie Bertinussen Nordheim, and Cato Alexander Bjørkli. 2018. What makes users trust a chatbot for customer service? An exploratory interview study. In *International Conference on Internet Science*, pages 194–208. Springer.

Batya Friedman and Helen Nissenbaum. 1996. Bias in computer systems. *ACM Transactions on Information Systems (TOIS)*, 14(3):330–347.

Mahault Garnerin, Solange Rossato, and Laurent Besacier. 2020. Gender Representation in Open Source Speech Resources. In *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, pages 6599–6605, Marseille, France. European Language Resources Association (ELRA).

Elodie Gauthier, Laurent Besacier, Sylvie Voisin, Michael Melese, and Uriel Pascal Elingui. 2016. Collecting resources in sub-Saharan African languages for automatic speech recognition: A case study of wolof. *LREC*.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. 2021. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92.

Matteo Gerosa, Diego Giuliani, and Fabio Brugnara. 2007. Acoustic variability and automatic recognition of children's speech. *Speech Communication*, 49(10-11):847–860.

Frances Gillis-Webber and Sabine Tittel. 2019. The shortcomings of language tags for linked data when modeling lesser-known languages. In *2nd Conference on Language, Data and Knowledge (LDK 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Frances Gillis-Webber and Sabine Tittel. 2020. A framework for shared agreement of language tags beyond ISO 639. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3333–3339.

Robert M Groves, Floyd J Fowler Jr, Mick P Couper, James M Lepkowski, Eleanor Singer, and Roger Tourangeau. 2011. *Survey Methodology*, volume 561. John Wiley & Sons.

Lucy Havens, Melissa Terras, Benjamin Bach, and Beatrice Alex. 2022. Uncertainty and inclusivity in gender bias annotation: An annotation taxonomy and annotated datasets of British English text. In *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, pages 30–57, Seattle, Washington. Association for Computational Linguistics.

Amy K Heger, Liz B Marquis, Mihaela Vorvoreanu, Hanna Wallach, and Jennifer Wortman Vaughan. 2022. Understanding machine learning practitioners' data documentation perceptions, needs, challenges, and desiderata. *Proceedings of the ACM on Human-Computer Interaction*, 6(CSCW2):1–29.

François Hernandez, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Estève. 2018. TED-LIUM 3: Twice as much data and corpus repartition for experiments on speaker adaptation. In *International Conference on Speech and Computer*, pages 198–208. Springer.

Arthur Hinsvark, Natalie Delworth, Miguel Del Rio, Quinten McNamara, Joshua Dong, Ryan Westerman, Michelle Huang, Joseph Palakapilly, Jennifer Drexler, Ilya Pirkin, et al. 2021. Accented speech recognition: A survey. *arXiv preprint arXiv:2104.10747*.

Kenneth Holstein, Jennifer Wortman Vaughan, Hal Daumé III, Miro Dudik, and Hanna Wallach. 2019. Improving fairness in machine learning systems: What do industry practitioners need? In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–16.

Wiebke Toussaint Hutiri and Aaron Yi Ding. 2022. Bias in automated speaker recognition. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 230–247.

Keith Ito. 2017. LJ Speech.

Zohar Jackson, César Souza, Jason Flaks, Yuxin Pan, Hereman Nicolas, and Adhish Thite. 2018. Jakobovski/free-spoken-digit-dataset: V1.0.8. Zenodo.

Manuel Jesús-Azabal, Javier Rojo, Enrique Moguel, Daniel Flores-Martin, Javier Berrocal, José García-Alonso, and Juan M Murillo. 2019. Voice assistant to remind pharmacologic treatment in elders. In *International Workshop on Gerontechnology*, pages 123–133. Springer.

Jan Jöhnk, Malte Weißert, and Katrin Wyrtki. 2021. Ready or not, AI comes—an interview study of organizational AI readiness factors. *Business & Information Systems Engineering*, 63(1):5–20.

Dewi Jones. 2020. Macsen: A voice assistant for speakers of a lesser resourced language. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU) and Collaboration and Computing for under-Resourced Languages (CCURL)*, pages 194–201.

Linus Kendall, Bidisha Chaudhuri, and Apoorva Bhalla. 2020. Understanding technology as situated practice: Everyday use of voice user interfaces among diverse groups of users in urban India. *Information Systems Frontiers*, 22:585–605.

Bret Kinsella and Ava Mutchler. 2020. Smart Speaker Consumer Adoption Report 2020. Technical report, Voicebot.AI.

Rob Kitchin. 2014. *The Data Revolution: Big Data, Open Data, Data Infrastructures and Their Consequences*. Sage.

Jens Klump and Robert Huber. 2017. 20 Years of persistent identifiers–Which systems are here to stay? *Data Science Journal*, 16:9–9.

Allison Koenecke, Andrew Nam, Emily Lake, Joe Nudell, Minnie Quartey, Zion Mengesha, Connor Toups, John R. Rickford, Dan Jurafsky, and Sharad Goel. 2020. Racial disparities in automated speech recognition. *Proceedings of the National Academy of Sciences*, 117(14):7684–7689.

Yunus Korkmaz and Aytuğ Boyacı. 2022. A comprehensive Turkish accent/dialect recognition system using acoustic perceptual formants. *Applied Acoustics*, 193:108761.

Librivox. 2021. Librivox - Acoustical liberation of books in the public domain.

Yvonna S Lincoln and Egon G Guba. 1985. *Naturalistic Inquiry*. Sage Publications, Inc, Beverly Hills, California, United States of America.

Chunxi Liu, Michael Picheny, Leda Sarı, Pooja Chitkara, Alex Xiao, Xiaohui Zhang, Mark Chou, Andres Alvarado, Caner Hazirbas, and Yatharth Saraf. 2022. Towards measuring fairness in speech recognition: Casual conversations dataset transcriptions. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6162–6166. IEEE.

Angelina McMillan-Major, Emily M. Bender, and Batya Friedman. 2023. Data statements: From technical concept to community practice. *ACM J. Responsib. Comput.*

Angelina McMillan-Major, Salomey Osei, Juan Diego Rodriguez, Pawan Sasanka Ammanamanchi, Sebastian Gehrmann, and Yacine Jernite. 2021. Reusable templates and guides for documenting datasets and models for natural language processing and generation: A case study of the HuggingFace and GEM data and model cards. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pages 121–135.

Milagros Miceli, Tianling Yang, Laurens Naudts, Martin Schuessler, Diana Serbanescu, and Alex Hanna. 2021. Documenting computer vision datasets: An invitation to reflexive data practices. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, pages 161–172, New York, NY, USA. Association for Computing Machinery.

Victor Minichiello, Rosalie Aroni, Eric Timewell, and Loris Alexander. 1990. *In-Depth Interviewing: Researching People*. Longman Cheshire, Melbourne, Australia.

Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. 2019. Model cards for model reporting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 220–229.

Arsha Nagrani, Joon Son Chung, Weidi Xie, and Andrew Zisserman. 2020. Voxceleb: Large-scale speaker verification in the wild. *Computer Speech & Language*, 60:101027.

Maryam Najafian and Martin Russell. 2020. Automatic accent identification as an analytical tool for accent robust automatic speech recognition. *Speech Communication*, 122:44–55.

Mikel K Ngueajio and Gloria Washington. 2022. Hey ASR system! Why aren't you more inclusive? Automatic speech recognition systems' bias and proposed bias mitigation techniques. A literature review. In *HCI International 2022–Late Breaking Papers: Interacting with eXtended Reality and Artificial Intelligence: 24th International Conference on Human-Computer Interaction, HCII 2022, Virtual Event, June 26–July 1, 2022, Proceedings*, pages 421–440. Springer.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An ASR corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE.

Orestis Papakyriakopoulos, Anna Seo Gyeong Choi, William Thong, Dora Zhao, Jerone Andrews, Rebecca Bourke, Alice Xiang, and Allison Koenecke. 2023. Augmented datasheets for speech datasets and ethical decision-making. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, pages 881–904.

Branislav Popović, Edvin Pakoci, Nikša Jakovljević, Goran Kočiš, and Darko Pekar. 2015. Voice assistant application for the Serbian language. In *2015 23rd Telecommunications Forum Telfor (TELFOR)*, pages 858–861. IEEE.

Mahima Pushkarna, Andrew Zaldivar, and Oddur Kjartansson. 2022. Data cards: Purposeful and transparent dataset documentation for responsible AI. In

Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency, FAccT '22, pages 1776–1826, New York, NY, USA. Association for Computing Machinery.

Inioluwa Deborah Raji, Emily M Bender, Amandalynne Paullada, Emily Denton, and Alex Hanna. 2021. AI and the everything in the whole wide world benchmark. In *Proceedings of the 35th Conference on Neural Information Processing Systems (NeurIPS 2021)*.

Anthony Rousseau, Paul Deléglise, and Yannick Esteve. 2012. TED-LIUM: An Automatic Speech Recognition dedicated corpus. In *LREC*, pages 125–129.

Johnny Saldaña. 2021. *The Coding Manual for Qualitative Researchers*. Sage.

Héctor A Sánchez-Hevia, Roberto Gil-Pita, Manuel Utrilla-Manso, and Manuel Rosa-Zurera. 2022. Age group classification and gender recognition from speech with temporal convolutional neural networks. *Multimedia Tools and Applications*, 81(3):3535–3552.

David Schlangen. 2021. Targeting the benchmark: On methodology in current natural language processing research. 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 670–674.

Hong Shen, Leijie Wang, Wesley H Deng, Ciell Brusse, Ronald Velgersdijk, and Haiyi Zhu. 2022. The model card authoring toolkit: Toward community-centered, deliberation-driven AI design. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 440–451.

James P. Spradley. 1979. *The Ethnographic Interview*. Harcourt Brace Jovanovich College Publishers, Fort Worth, Texas, United States.

Rachael Tatman. 2017. Gender and dialect bias in YouTube's automatic captions. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 53–59.

Rachael Tatman and Conner Kasten. 2017. Effects of talker dialect, gender & race on accuracy of bing speech and YouTube automatic captions. In *INTERSPEECH*, pages 934–938.

Joseph Turow. 2021. *The Voice Catchers*. Yale University Press.

Rob Van der Meulen and Amy Ann Forni. 2016. Gartner Says Worldwide Spending on VPA-Enabled Wireless Speakers Will Top $2 Billion by 2020. *Gartner*.

Ravichander Vipperla, Steve Renals, and Joe Frankel. 2010. Ageing voices: The effect of changes in voice parameters on ASR performance. *EURASIP Journal on Audio, Speech, and Music Processing*, 2010:1–10.

Etienne Wenger-Trayner and Beverly Wenger-Trayner. 2014. Learning in a landscape of practice: A framework. In *Learning in Landscapes of Practice*, pages 13–29. Routledge.

Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al. 2016. The FAIR Guiding Principles for scientific data management and stewardship. *Scientific data*, 3.

Michael Williams and Tami Moser. 2019. The art of coding and thematic exploration in qualitative research. *International Management Review*, 15(1):45–55.

Sue Ellen Wright. 2019. Standards for the language, translation and localization industry. In *The Routledge Handbook of Translation and Technology*, pages 21–44. Routledge London and New York.

Marcia Lei Zeng and Jian Qin. 2016. *Metadata*. Facet.

A Interview participant summary

Table 1: Interview participant summary (n=13)

Characteristic		Total
Gender	Female	3
	Male	10
	Other gender expressions	0
Occupational field	Research scientist or academic	5
	ML or NLP Engineer	2
	Software Engineer	2
	Data annotator	1
	Developer Relations Advocate	2
	UX Designer / researcher	1
Country of residence	United States	5
	Australia	3
	South Africa	1
	Aotearoa New Zealand	1
	Nigeria	1
	France	1
	Canada	1

B Summary of voice dataset documents analysed

Table 2: Summary of voice dataset documents analysed

Characteristic of the dataset or voice dataset document (VDD)	Mozilla Common Voice(Ardila et al., 2020)	Librispeech (Panayotov et al., 2015)	African languages in the field (Gauthier et al., 2016)	Voxceleb (Chung et al., 2018; Nagrani et al., 2020)	LDC 2000 HUB5 English Evaluation Speech	TED-LIUM corpus (Rousseau et al., 2012; Hernandez et al., 2018)	Free spoken digit dataset (Jackson et al., 2018)	CHIME 5 Speech separation challenge dataset (Barker et al., 2018)	LJSpeech Speech dataset (Ito, 2017)
Type of document(s) analysed	CommonVoice website, GitHub repository, related paper	Entry on OpenSLR website, related paper	Entry on OpenSLR website, `README` file in dataset, related paper	VoxCeleb website, Metadata file archived on archive.org, related papers	LDC Catalogue entry	TED-LIUM website, `README` file in dataset, related paper	GitHub repository, Zenodo dataset record, `metadata.py` file in dataset	Data page on CHIME website, `JSON` file in dataset	LJ Speech website
Year of initial release & latest version	2018; 2023 (version 13)	2015; no newer version	2005; no newer version	2017; 2018 (version 2)	2005; no newer version	2012; 2018 (version 3)	2018; no newer version	2018	2017; 2017 (version 1.1)
Intended language task	Speech recognition	Speech recognition, multilingual	Speech recognition, monolingual	Speaker identification, speech separation, monolingual	Speech recognition, monolingual	Speech recognition, monolingual	Speech recognition, monolingual	Speech separation, monolingual	Speech synthesis, monolingual
Nature of speech in dataset	Elicited, large vocabulary, multiple domains	Elicited, large vocabulary, out of copyright works	Elicited, large vocabulary, multiple domains	Spontaneous, large vocabulary, multiple domains	Spontaneous, large vocabulary, multiple domains	Spontaneous, large vocabulary, multiple domains	Elicited, constrained vocabulary, spoken digits	Spontaneous, large vocabulary, multiple domains	Elicited, large vocabulary, non-fiction books publishes between 1884 and 1964
Motivation and funding source	Ecosystem development; Grant-based for particular languages; additional funding from NVIDIA	Research; funding unknown.	Research; ALFFA Research Project, funded by agence nationale de la recherche.	Research by Oxford University, funded through EPSRC programme grant	Commercial; Sponsored by National Institute of Standards and Technology.	Research, funding not specified.	Research, funding not specified.	Research challenge sponsored by Google and Microsoft Research.	Research, funding not specified, independent researcher.
Method of collection of dataset	Volunteer speakers recorded on web-based platform.	Secondary use dataset from Librivox volunteer audio book project (Librivox, 2021)	Original dataset, volunteer speakers recorded in field.	Secondary use dataset from YouTube; speakers' consent not provided.	Original dataset, recruited speakers recorded via telephone.	Secondary use dataset from TED videos; speaker consent unknown.	Original dataset, speaker recruitment and recording unknown.	Original dataset, speaker recruitment unknown, recorded in speakers' homes.	Tertiary dataset, subset of Librispeech containing single speaker. Speaker consent not stated.

C Summary of dataset documentation analysis

Table 3: Descriptions and data items included in current voice dataset documentation

Data item	Mozilla Common Voice	Librispeech	African languages in the field	Voxceleb	LDC 2000 HUB5 English Evaluation Speech	TED-LIUM corpus	Free spoken digit dataset	CHIME 5 Speech separation challenge dataset	LJSpeech Speech dataset
Dataset identification									
Persistent identifier for the dataset	No	No	No	No	Yes	No	Yes	Yes	No
Dataset versioning	Yes	Yes	No	Yes	Yes	Yes	Yes	Implied via yearly competition	Yes
Dataset release date	Yes	Implied through related paper	Yes	Implied through related paper	Yes	Implied through related paper	Yes	Yes	Yes
Intent, purpose and curation rationale									
Executive summary or description	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Intended tasks or use cases	Yes	Yes	Yes	Yes	Yes	Yes	Implied through GitHub repository tags	Yes	No
Curation rationale	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Dataset creation process, sources and actors									
Dataset collection method	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes
For elicited speech, the source of prompts	Implied through GitHub history	Yes	Yes	Yes	No	Yes	No	Yes	Yes
For spontaneous speech, description of the annotation/transcription process	N/A	N/A	N/A	N/A	No	Yes	N/A	No	N/A
For spontaneous speech, description of the annotators	N/A	N/A	N/A	No	N/A	No	N/A	N/A	N/A
Characteristics of the dataset itself									
Structure of dataset, such as field mapping, described	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Dataset storage size provided	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes
Overall hours of speech in dataset specified	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
License specified	CC0	CC-BY-4.0	MIT	CC-BY-SA-4.0	LDC User Agreement	CC-BY-NC-ND-3.0	CC-BY-SA-4.0	Dataset specific	Public domain
# of distinct voices in dataset specified	Yes	Yes	Yes	Yes	Implied via # of conversations	Yes, in paper	Yes	Yes	Yes
# of utterances in dataset specified	Yes	Yes	Yes	Yes	No, only # of conversations given	Yes, in paper	Yes	Yes	Yes
Length of utterances given	Yes	No	Yes, averaged	Implied via each utterance having same length	No	No	Implied via each utterance being a single digit	Inferred via JSON file	Yes, averaged
Split information (test, train, dev etc) provided	Yes	Yes	Yes, in data structure	No	No	Yes, in data structure	Yes	Yes	N/A, splits not used in speech synthesis
Audio file type specified	Yes, in data structure	Yes, in data structure	Yes, in data structure	No	File type implied by sample file	Yes	Yes	Yes	Yes
Audio file format details (resolution etc) provided	No	Yes (some)	No	No	Yes	Yes	Yes	Yes	Yes

Table 4: Descriptions and data items included in current voice dataset documentation (continued)

Data item	Mozilla Common Voice	Librispeech	African languages in the field	Voxceleb	LDC 2000 HUB5 English Evaluation Speech	TED-LIUM corpus	Free spoken digit dataset	CHIME 5 Speech separation challenge dataset	LJSpeech Speech dataset
How the dataset represents characteristics of the speaker(s)									
Representation or distribution of speaker accent	Yes	No	No	No	No	No	Yes	No	No
Representation or distribution of speaker nationality	No	No	No	No	No	No	Yes	No	No
Representation or distribution of speaker age	Yes	No	No	No	No	No	No	No	No
Representation or distribution of speaker gender	Yes	Yes	Not in dataset, but distribution specified in paper	Yes	No	Yes, in paper	Yes	Yes	Implied, single speaker, gender specified
Representation of speaker occupation	No	No	Not in dataset, but overview given in paper	No	No	No	No	No	No
Representation of speaker language acquisition or heritage	No	No	No	No	No	No	No	No	No
Representation of speaker educational attainment	No	No	No	No	No	No	No	No	No
How the dataset represents the recording environment									
Constitution by recording hardware	No	No	No	No	Implied (telephone	No	No	No	No
Constitution by recording environment	No	No	No	Implied (interview)	Implied (phone conversations)	Implied (TED talks)	No	Yes	No
How the dataset represents characteristics of spoken language									
Dataset language(s) represented using a standard such as `BCP-47` or `ISO-639`	Yes	Yes	Yes	Yes	Yes	No	No	No	No
Multilingual flag	Implied through dataset structure	Yes	Implied through dataset structure	No	No	No	No	No	No
Representation of phonetic distribution or variation	No	No	No	No	No	No	No	No	No
Representation of dialect, lexical or non-phonetic variation	No	No	No	No	No	No	No	No	No
Representation of domain of speech	No	Implied due to Librivox source	No	No	No	No	Implied - digits	No	Implied due to Librivox source
Constitution by formality or register of spoken language	Varies with prompt	Varies with prompt	No	No	No	Implied - TED talks	No	No	Varies with prompt
For spontaneous speech, whether code-switching is indicated	N/A	N/A	N/A	No	No	No	N/A	No	N/A
Models, benchmarks and academic papers									
Benchmarks specified or linked to	No, uses CER for eval'n but no benchmark	Yes, WSJ	No	Yes, previous speaker recog'n datasets	No	No	No, uses WER and CER for eval'n but no benchmark	No	No
Models trained from dataset specified or linked to	Yes, specified in paper	Yes, specified in paper	Yes, specified in paper	Yes, specified in paper	No	No	No	Yes, in results page	No
Papers based on dataset specified or linked to	Yes, on website	Yes, on website	Yes, on website	Yes, on website	No	Yes, on website	Yes	Yes, in results page	No
Privacy, limitations and social impact									
Privacy or sensitivity statement of the dataset	Some info on website	No	No	Has a privacy statement	No	No	No	No	No
Social impact statement of the dataset	Some info on website	No	No	No	No	No	No	No	No
Statement of biases in dataset	No	No	No	No	No	No	No	No	No
Statement of limitations of dataset	No	No	No	No	No	No	No	No	No

MCASP: Multi-Modal Cross Attention Network for Stock Market Prediction

Kamaladdin Fataliyev and **Wei Liu**
University of Technology Sydney, Sydney, Australia
kamaladdin.fataliyev@student.uts.edu.au, wei.liu@uts.edu.au

Abstract

Stock market prediction is considered a complex task due to the non-stationary and volatile nature of the stock markets. With the increasing amount of online data, various information sources have been analyzed to understand the underlying patterns of the price movements. However, most existing works in the literature mostly focus on either the intra-modality information within each input data type, or the inter-modal relationships among the input modalities. Different from these, in this research, we propose a novel Multi-Modal Cross Attention Network for Stock Market Prediction (MCASP) by capturing both modality-specific features and the joint influence of each modality in a unified framework. We utilize financial news, historical market data and technical indicators to predict the movement direction of the market prices. After processing the input modalities with three separate deep networks, we first construct a self-attention network that utilizes multiple Transformer models to capture the intra-modal information. Then we design a novel cross-attention network that processes the inputs in pairs to exploit the cross-modal and joint information of the modalities. Experiments with real world datasets for S&P500 index forecast and the prediction of five individual stocks, demonstrate the effectiveness of the proposed multi-modal design over several state-of-the-art baseline models.

1 Introduction

Stock market movements are inherently affected by a multitude of data sources, encompassing historical price data, technical indicators (Vargas et al., 2017), financial news (Schumaker and Chen, 2009), social media (Chen et al., 2018), and official announcements (Feuerriegel and Gordon, 2018). It has been established that analyzing these multiple data modalities together enables the capture of underlying patterns in stock movements, rendering stock market prediction a multi-modal learning task (Akita et al., 2016). The efficacy of employing effective multi-modal representation and learning techniques to uncover the joint influence of these data modalities is pivotal for model performance (Li et al., 2020). Simultaneously, it is important to extract the intra-modal information within each data source. Early information fusion techniques combine raw input features initially and then construct a prediction model, which aids in capturing the combined influence of modalities but neglects intra-modal information. Late fusion techniques, conversely, analyze input features separately and subsequently employ a fusion layer for prediction. While this approach facilitates a focus on modality-specific features, it may overlook inter-modal information. Balancing the capture of intra-modal and inter-modal information from input modalities is essential.

Researchers have identified that pairs of data modalities, such as financial news and market prices, as well as market prices and technical indicators (Vargas et al., 2017), both impact price movements. However, existing models, while striving to capture the joint influence of all modalities together, may overlook the underlying bi-modal relationships between various data inputs. Therefore, in addition to capturing their collective influence, it is also crucial to understand the bi-modal relationships among pairs of input modalities.

To address these challenges, various methods have been developed, primarily categorized as inter-modality and intra-modality-based techniques. Inter-modality methods aim to capture the underlying relationships among input modalities but may miss the connections within each modality. Conversely, intra-modality techniques focus on uncovering modality-specific relations but tend to disregard the inter-modal connections across input modalities. Combining modality-specific features with inter-modal connections can synergize and enhance overall analysis. Hence, exploring a uni-

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 67–77
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

fied framework capable of capturing both inter-modality and intra-modality relations within the input data is imperative.

Motivated by these challenges, we present a novel Multi-Modal Cross-Attention Network for Stock Market Prediction (MCASP). MCASP forecasts the direction of price movements by jointly modeling inter-modality and intra-modality relationships within the input data (i.e., financial news, market data, and technical indicators) within a unified deep learning framework. To achieve this, we construct two distinct attention networks: a self-attention network and a cross-attention network, designed to capture intra-modal and inter-modal relationships, respectively.

The self-attention module focuses on extracting modality-specific features from the input modalities. We first employ two separate Long Short-Term Memory (LSTM) networks to extract latent features from market data and technical indicators. Simultaneously, we leverage FinBERT (Liu et al., 2020b) to encode textual data (i.e., financial news). Within the self-attention network, the LSTM network outputs are processed by two Transformer (Vaswani et al., 2017) units, while the encoded textual data undergo analysis via a Convolutional Neural Network (CNN).

The cross-attention module involves creating three pairs by concatenating representations of news and market data, news and technical indicators, and market data and technical indicators. These pairs are then fed into three separate Transformer units. The outputs from the self-attention and cross-attention modules converge in the Fusion Layer to generate a combined feature vector. Finally, we employ a fully-connected layer to predict the direction of price movements.

2 Related Work

In this section, we review related work in stock market prediction, multimodal machine learning and the attention mechanism.

2.1 Stock Market Prediction

Financial news, market data, social media data, official company announcements have been widely used for market analysis research. It has been shown by Shi et al. (2019a) that using only news titles is better than using the whole article text. Schumaker et al. (2012) proposed the Arizona Financial Text (AZFinText) system, focusing on sentiment analysis using propoer nouns. In another study, Vargas et al. (2017) represented news headlines using Word2Vec word embeddings and constructed a multimodal prediction model using Convolutional Neural Networks (CNN) and Long Short-term Memory (LSTM) networks. Meanwhile, Huynh et al. (2017) designed a prediction model using the Bidirectional Gated Recurrent Unit (BGRU) architecture, extracting news headlines and representing them using word embedding vectors.

The paper by Nuij et al. (2014) used ViewerPro to extract events from news articles and incorporated them with technical indicators. Matsubara et al. (2018) employed paragraph vectors for news data representation, and Ding et al. (2015) introduced a CNN-based event embeddings model where the authors constructed a neural tensor network to learn event embeddings from financial news data.

2.2 Multimodal Machine Learning

Multimodal learning architectures have been widely utilized in various fields including robotics (Lee et al., 2018), healthcare (Ghulam et al., 2021), multimedia (Liang et al., 2018), and sentiment analysis (Zadeh et al., 2018). A multimodal paper by Barnum et al. (2020) applies early fusion in the multimodal representation of audio and visual inputs and another research (Federici et al., 2020) employs structured image and textual to construct multimodal concept taxonomies. Researchers have also utilized various RNN structures for multimodal representations for different kinds of applications such as human behaviour analysis (Rajagopalan et al., 2016) and time-series data analysis (Liang et al., 2018; Zadeh et al., 2018).

One popular technique for combined utilization of multimodal data is early fusion (Morency et al., 2011; Pérez-Rosas et al., 2013). Early fusion concatenates low-level features from individual modalities to be utilized with any learning framework for downstream machine learning tasks. Moreover, early fusion performs poorly when feature fusion among non-interacting modalities (such as voice and fingerprint) is performed. These limitations are slightly addressed in Zadeh et al. (2016), where shared embeddings (latent space) among individual modalities are learned. These shared representations outperform the early fusion but require careful parameter tuning.

There also exists a stream of work that perform outer-product-based neural frameworks for multi-modal data fusion. In Lin et al. (2015) a bilinear-CNN is proposed to obtain bi-modal interactions among features obtained from two heterogeneous CNNs. This is accomplished by taking a neural-based bilinear product of high-level features. The bilinear layer required parameter estimation of a quadratic number of neurons and hence prone to over-fitting. This limitation is alleviated in Fukui et al. (2016); Hu et al. (2017a) which introduced an alternate formulation of the bilinear layer and obtains its compact representation by utilizing sophisticated neural-based factorization schemes.

2.3 Attention

The attention mechanism has found success in a wide range of domains, including natural language processing (NLP) (Bahdanau et al., 2014; Vaswani et al., 2017), image captioning (You et al., 2016), image classification (Xiao et al., 2014), visual question answering (Lu et al., 2016), and more (Rush et al., 2015; Li et al., 2015). Notably, the Transformer model (Vaswani et al., 2017) introduced the self-attention mechanism, which explores intra-modal relationships, such as the relationships between words in machine translation.

Taking inspiration from the Transformer model (Vaswani et al., 2017), the self-attention mechanism has been applied in various works, extending its utility to visual question answering (Yu et al., 2019), video analysis (Wang et al., 2017), and image-text matching (Wu et al., 2019).

In recent years, attention mechanisms have also made their way into multi-modal learning problems. While architectures like BERT (Devlin et al., 2019) were originally designed for NLP tasks, they have been adapted for multi-modal challenges as well (Chen et al., 2019; Lu et al., 2019). For instance, some approaches, like the dual attention network in Nam et al. (2016), focus on learning inter-modal relationships between visual regions and textual elements within sentences. Others, like the co-attention framework in Lu et al. (2016), tackle tasks like visual question answering by jointly learning image and question attentions. Additionally, in Paulus et al. (2017), a combination of inter-modal and intra-modal attentions is leveraged within deep reinforcement learning for text summarization.

3 Model Design

In this section, we provide a detailed description of the architecture of the proposed MCASP model. The design of our MCASP model is demonstrated in Figure1.

3.1 Input Representation

We start by using historical market data and financial news as our primary data sources. From the market data, we derive a set of seven technical indicators. We employ three distinct data modalities for stock market prediction: market data, technical indicators, and financial news. To process these modalities, we employ three separate deep networks.

We construct two LSTM networks to handle the market data and technical indicator modalities, respectively. Additionally, we utilize text embeddings to encode the news data. For this purpose, we leverage BERT and FinBERT embeddings.

The latent features obtained from the LSTM networks and the sentence embeddings from FinBERT are then fed into the self-attention and cross-attention modules to capture both intra-modal and inter-modal relationships.

3.2 Self-Attention Module

The primary objective of the attention process is to discern the relationship between two states and focus on the most crucial features. This is achieved by assigning higher weights to the most pertinent elements within the input vectors. The attention layer consists of three key components: the query, keys, and values, with these elements being identical in the self-attention context. The attention mechanism can be conceptualized as mapping a query and a set of key-value pairs to an output, where the output is a weighted sum of the values. The weight matrix, determining the weight assigned to each value, is defined using the query and the key. Several options for the attention function are available, including the dot product, multi-layer perceptron, and scaled dot product.

The self-attention network is used to capture intra-modality relations, employing two separate Transformer units (Li et al., 2014) for market data and technical indicators, along with a CNN for financial news data. In the Transformer model, we employ the scaled dot product to compute the weight matrix. This module encompasses both multi-head self-attention and position-wise feed-

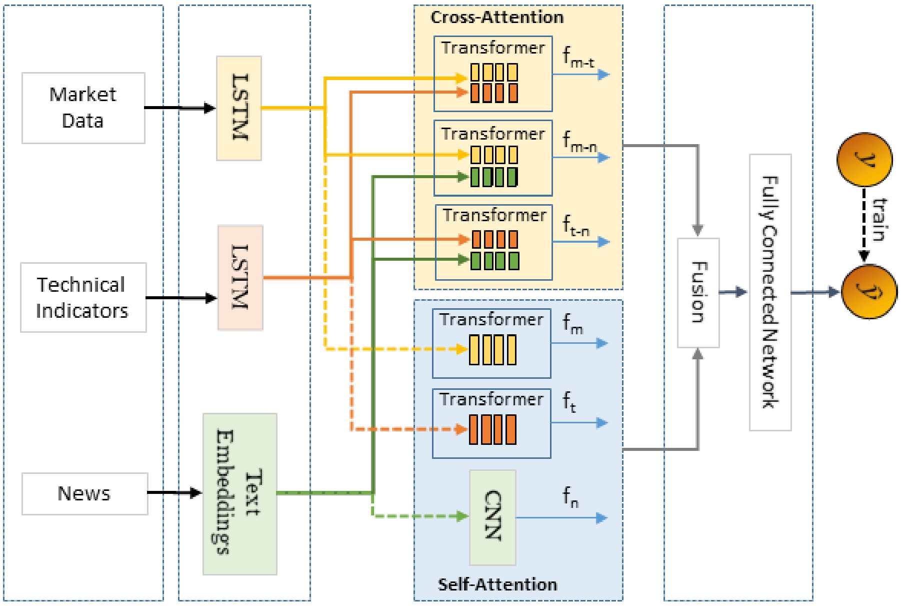


Figure 1: Demonstration of the MCASP architecture design.

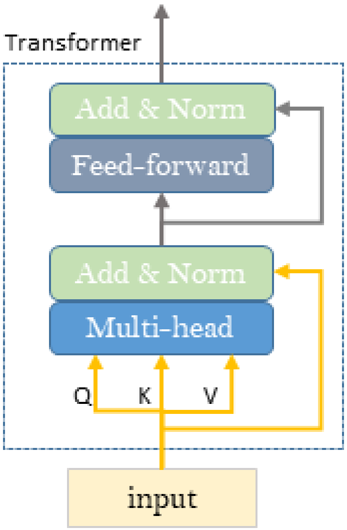


Figure 2: Design of the Transformer model

forward layers, as depicted in Figure 2. The term 'multi-head attention' implies that attention is computed multiple times. The attention calculation is as follows:

$$A(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{1}$$

Where the d_k represents the dimension of the queries and the keys. In the Transformer module, multiple parallel attention values are computed where each output is called a head. The ith head is calculated as:

$$head_i = A(QW_i^Q, KW_i^K, VW_i^V) \tag{2}$$

We then concatenate these heads to obtain the multi-head attention.

$$MT(Q, K, V) = Concat(head_1, ., head_h)W^0 \tag{3}$$

In our self-attention module, the two Transformers for market data and technical indicators modalities, we get the following two outputs:

$$\begin{aligned} f_m &= MT(Q_m, K_m, V_m) \\ f_t &= MT(Q_t, K_t, V_t) \end{aligned} \tag{4}$$

For the textual data modality, we utilize the outputs of the BERT embeddings. The BERT model incorporates multiple Transformers and is proficient at capturing intra-modality information. Subsequently, we employ a CNN to extract local latent features denoted as f_n.

These three outputs from our self-attention module, namely f_m, f_t, and f_n, are later employed to predict the movement of closing prices.

3.3 Cross-Attention Module

We introduce a novel cross-attention to model both intra-modality information and the interconnectedness of the modalities, achieved by implementing three separate Transformer units. By modeling both intra-modality and inter-modality relationships, we aim to capture the joint effect of the input modalities while retaining modality-specific features.. Our aim is to capture the interactions across the input modalities by applying the cross-attention function to the outputs of the input representation layer. Initially, we establish three distinct pairs from the modalities to implement the attention mechanism: from market data to technical indicators $(m - t)$, from market data to financial news $(m - n)$, and from technical indicators to financial news $(t - n)$. Market data and the derived technical indicators have a significant influence on market movements, which justifies prioritizing these pairings with higher weights.

The calculation of these three cross-attention values is as follows:

$$\begin{aligned} A_{m-n}(Q_m, K_n, V_n) &= softmax(\frac{Q_m K_n^T}{\sqrt{d_k}})V_n \\ A_{m-t}(Q_m, K_t, V_t) &= softmax(\frac{Q_m K_t^T}{\sqrt{d_k}})V_t \\ A_{t-n}(Q_t, K_n, V_n) &= softmax(\frac{Q_t K_n^T}{\sqrt{d_k}})V_n \end{aligned} \tag{5}$$

Here, A_{m-n}, A_{m-t}, and A_{t-n} represent the cross-attention between market data and news, market data and technical indicators, and technical indicators and news modalities, respectively. Furthermore, Q_m and Q_t denote the query vectors for the market data and technical indicators modalities, while K_t and K_n represent the key vectors, and V_t and V_n denote the value vectors for the technical indicators and news modalities, respectively.

With these cross-attention terms in place, we proceed to compute the attention values for each head as follows:

$$\begin{aligned} head^i_{m-n} &= A_{m-n}(Q_m W_i^{Q_m}, K_n W_i^{K_n}, V_n W_i^{V_n}) \\ head^i_{m-t} &= A_{m-t}(Q_m W_i^{Q_m}, K_t W_i^{K_t}, V_t W_i^{V_t}) \\ head^i_{t-n} &= A_{t-n}(Q_t W_i^{Q_t}, K_n W_i^{K_n}, V_n W_i^{V_n}) \end{aligned} \tag{6}$$

These terms represent each head in each cross-attention pair. Subsequently, we combine these head values for each pair to obtain the multi-head attention for each cross-attention block:

$$\begin{aligned} MT_{m-n} &= Concat(head^1_{(m-n)}, ., head^h_{(m-n)})W^0_{m-n} \\ MT_{m-t} &= Concat(head^1_{(m-t)}, ., head^h_{(m-t)})W^0_{m-t} \\ MT_{t-n} &= Concat(head^1_{(t-n)}, ., head^h_{(t-n)})W^0_{t-n} \end{aligned} \tag{7}$$

Putting all these together, our cross-attention module produces the following three outputs:

$$\begin{aligned} f_{m-n} &= MT_{m-n} \\ f_{m-t} &= MT_{m-t} \\ f_{t-n} &= MT_{t-n} \end{aligned} \tag{8}$$

3.4 Fusion Layer

In the fusion layer, we amalgamate the feature vectors from the self-attention and cross-attention modules to form a combined feature vector.

$$f_{merged} = [f_m, f_t, f_n, f_{m-n}, f_{m-t}, f_{n-t}] \tag{9}$$

We then employ a fully connected layer with ReLU as the activation function to process the feature vector f_{merged}. In the final step, another fully connected layer is employed to make predictions. The overall network is a binary classification model used for predicting the movement direction of stock closing prices, and the model weights are optimized by minimizing the binary cross-entropy loss:

$$L = -(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})) \tag{10}$$

where y represents the target class for the movement direction, and $\hat{y}$ signifies the prediction obtained from MCASP. The movement direction is defined as the difference between the closing prices on day $t + 1$ and day t. The labels are categorized into two classes: Class 1 indicating an upward movement and Class 0 indicating a downward movement in the closing prices.

4 Experimental Settings

In our experiments, we utilized real-world datasets encompassing financial news, market data, and technical indicators spanning from January 1, 2010, to December 31, 2019, encompassing a 10-year period. The financial news was sourced from Reuters[1], with each article containing a title, body, and publication date. The publication date was employed to align the articles with the daily market data. We specifically focused on the headlines from the financial news, as research has demonstrated that using news titles can yield superior prediction results compared to using the entire article body (Shi et al., 2019b). The number of news titles per trading day varied; hence, we aggregated all the titles for a given day into a single extended sentence

---

[1] `https://www.reuters.com/business/finance/`

and employed FinBERT to encode the textual data into feature vectors. Consequently, we obtained a single sentence embedding vector for each trading day.

We utilize historical market data for S&P index and individual stocks from Yahoo Finance[2] for the corresponding dates. These five companies included Google, Tesla, Amazon, Apple, and Microsoft and the data includes Open, High, Low, Close prices, and Volume. We normalize the market data to be within the range of [0, 1].

We initially employ an 80-20% split for training and testing for index price prediction. We also evaluate the yearly performances of the models by utilizing the first 10 months of each year for training and the last 2 months for testing. We utilize the 80-20% split again for training and testing purposes for individual stock prediction.

Based on the literature (Kim, 2003), we computed seven technical indicators for each trading day using the market data over the preceding five days.

We employ accuracy (Acc) and Matthews Correlation Coefficent (MCC) to evaluate the performance of different models. MCC is generalyy employed when the sizes of classes y = 1 and y = 0 differ.

4.1 Baseline Methods

We compare our approach with the following baselines on predicting individual stocks and S&P500 index.

Recurrent Convolutional Neural Network (RCNN) (Vargas et al., 2017) is a CNN and RNN based stcok forecast model that utilizes technical indicators and financial news. **Event Embeddings (EB-RCN)** (Oncharoen and Vateekul, 2018) is another LSTM and CNN based model that also includes market data and employ event embeddings from (Ding et al., 2015). **Bidirectional Gated Recurrent Unit (BGRU)** (Huynh et al., 2017) uses both online financial news and historical price data to predict the stock movements. **LSTM-based Recurrent State Transition (ANRES)** (Liu et al., 2020a) uses only news events for market movement prediction. **Hybrid Attention Network (HAN)** (Hu et al., 2017b) is a state-of-the-art stock trend prediction model with hierarchical attention that utilizes news data. **Multi-Modality Attention Network** (MMAN) (He and Gu, 2021) **Attention-Based Recurrent Neural Network (At-LSTM)** (Liu, 2018) **Adversarial Attentive LSTM (Adv-LSTM)** (Feng et al., 2018) is a market prediction model using historical market data, where the authors employ attentive LSTMs and utilize adversarial training strategy.

Other than these methods, we also perform ablation studies by constructing different variants of the proposed MCASP model.

5 Results and Analysis

In order to test the effectiveness of our model, we run experiments using real-world dataset including financial news data, historical market data and technical indicators.

5.1 Main Results

We use our dataset to conduct tests for forecasting of the price movements of S&P500 index and five individual stocks. The accuracy results are illustrated in Figure 3, showing that MCASP improves upon the baseline models. The MCC results, presented in Figure 4, echo the same trend, with MCASP exhibiting superior prediction performance for the price movement directions of all five stocks and S&P index compared to the baseline models.

Overall, in our experiments, MCASP consistently achieves the best results in terms of both accuracy and MCC. When compared to the baselines, MCASP demonstrates improvements in prediction performance for both index and individual stock predictions, underscoring the effectiveness of the proposed multi-modal attention design in leveraging intra-modal and inter-modal information from multiple input sources.

Among the baseline models, attention-based prediction models perform better than other baselines in both accuracy and MCC. These results underscore the significance of the attention module in capturing critical latent features from the input data. However, MCASP surpasses the attention-based baseline models, suggesting that its enhanced performance stems not only from the use of the self-attention module but also from its ability to extract inter-modal relationships among input modalities through the novel cross-attention module.

We also asses the models' yearly prediction performances for S&P 500 index prediction, where we use the first 10 months of each year for training and the last two months for testing. The accuracy re-

---

[2] `https://finance.yahoo.com/`

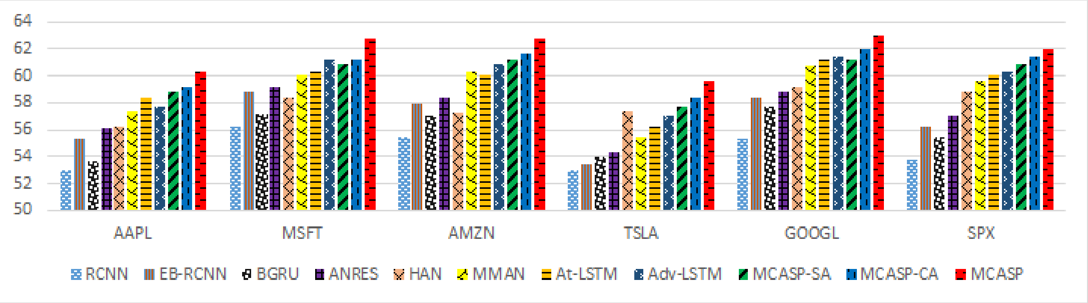


Figure 3: Accuracy results on index and individual stock prediction (the higher, the better).

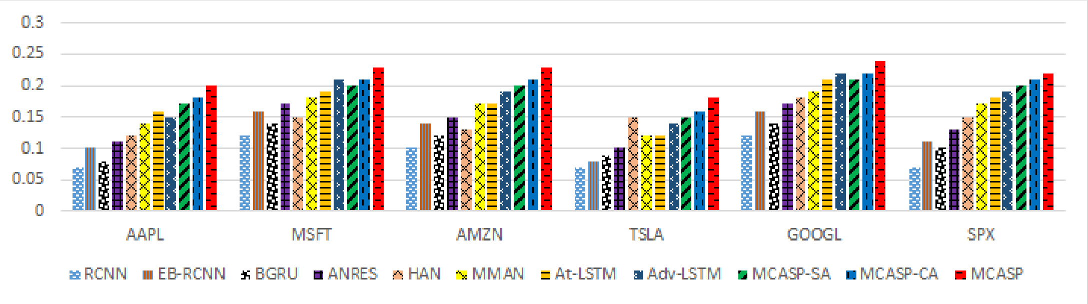


Figure 4: MCC results on index and individual stock prediction (the higher, the better).

sults, given in Figure 5, demonstrate that MCASP consistently outperforms all the baseline models for each year. Although the yearly results are slightly lower than the initial test results, this can be attributed to the smaller test sample size inherent in the yearly setup.

Collectively, the experiments involving S&P500 index prediction and the prediction of price movements for five individual stocks demonstrate that the MCASP model is adept at learning meaningful representations from multiple input modalities, capitalizing on the self-attention network and the innovative cross-attention module.

5.2 Ablation Study

To assess the impact of different components of the MCASP model, we conducted an ablation study using the same real-world dataset. Initially, we evaluated the effectiveness of our two attention modules independently by creating two distinct models. Subsequently, we explored three text embedding techniques to demonstrate the influence of the textual representation method on the overall performance.

Self-attention and cross-attention modules. This experimental study elucidates the individual performance of each module and underscores the significance of capturing both intra-model and inter-model information, in contrast to the prevalent approach of focusing solely on either modality-specific or joint influence of input modalities, as seen in most existing works. To this end, we developed two distinct models - MCASP-SA (MCASP with the self-attention module only) and MCASP-CA (MCASP with the cross-attention module only) - and subjected them to testing using our original dataset.

In our experiments, MCASP consistently outperforms both MCASP-SA (which exclusively employs the self-attention module) and MCASP-CA (which relies solely on the cross-attention module) across both accuracy and MCC metrics. This substantiates the effectiveness of our proposed design in addressing multi-modal problems.

Notably, MCASP-CA yields superior results compared to MCASP-SA. We postulate that this is attributed to the cross-attention module's design, which initially extracts modality-specific features and subsequently captures inter-modal relationships among modalities using the attention mechanism.

Moreover, when compared to the baseline models, both MCASP-SA and MCASP-CA consistently demonstrate improved accuracy and MCC results in the majority of the tests. This underscores the success of the proposed sequential design for both modules. The results further affirm that leveraging multiple modalities (i.e., financial news, his-

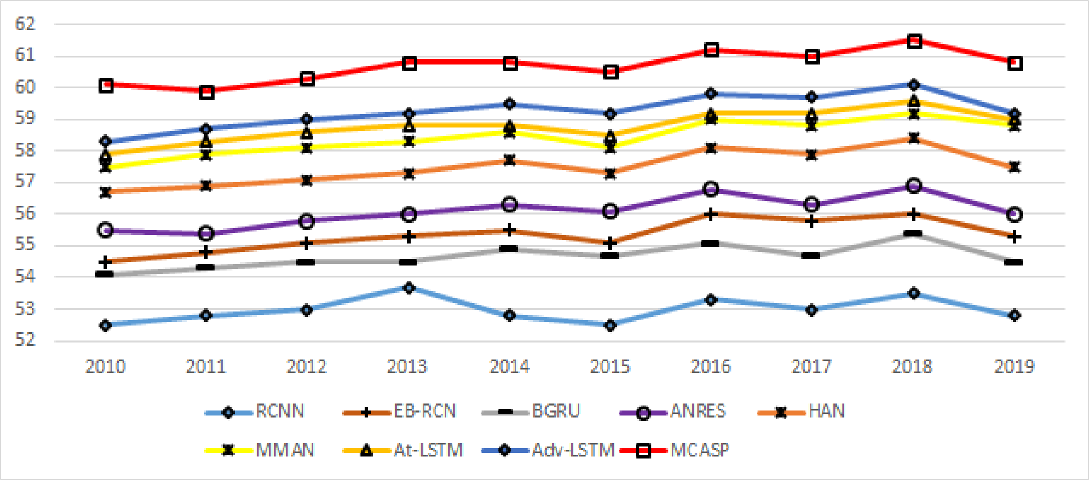


Figure 5: Yearly ACC results on S&P index prediction (the higher, the better).

torical market data, and technical indicators) can enhance model performance.

MCASP with various text embeddings. We subsequently examined the impact of various textual embeddings (Transformer-based BERT and GloVe) on the overall model performance. We employed three distinct textual embedding methods to encode and represent the financial news data, namely GloVe word embeddings, Transformer-based BERT embeddings, and FinBERT embeddings. Our experimental results underscore the significance of selecting an appropriate text embedding method when utilizing financial news data.

The results, presented in Table1 show that Transformer-based BERT and FinBERT embeddings consistently outperformed GloVe embeddings across both accuracy and MCC metrics for S&P index prediction. Furthermore, FinBERT showed improved results compared to BERT embeddings, underscoring the value of domain-specific knowledge in textual data representation.

Table 1: The impact of different text embedding methods.

Embedding Method	***Accuracy***	***MCC***
GloVe	60.91%	0.208
BERT	61.60%	0.215
FinBERT	62.03%	0.228

Notably, predictions using FinBERT as our text embedding method exhibited improvement compared to GloVe and BERT embeddings. This highlights the utility of domain knowledge in comprehending and representing textual data. However, even without domain knowledge and when employing RNN-based GloVe embeddings and general BERT embeddings, MCASP consistently outperformed all baseline methods across both metrics for S&P500 index prediction. These results affirm that while a robust textual representation technique can enhance model performance, the primary factor contributing to improved results lies in the novel multi-modal design, which incorporates both self-attention and cross-attention modules to capture latent features from the input modalities.

6 Conclusion

We have proposed a novel multi-modal cross attention network for stock market prediction that models the intra-modal and inter-modal information from the input modalities in a unified framework. We first analyze the input modalities via three separate deep networks to extract the salient features. We then process these features with the proposed self-attention and cross-attention modules to jointly model the intra-modal and inter-modal information. We analyze financial news, historical market data and technical indicators to predict the movement direction of S&P500 index prices and the prices of five individual stocks. We test the effectiveness of the proposed multi-modal design using real-world dataset from Reuters and Yahoo! Finance and compare its performance against multiple state-of-the-art baseline models. Experimental results show that our model achieves improved performance in stock market prediction.

References

R. Akita, A. Yoshihara, T. Matsubara, and K. Uehara. 2016. Deep learning for stock prediction using numerical and textual information. *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, pages 1–6.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

George M. Barnum, Sabera Talukder, and Yisong Yue. 2020. On the benefits of early fusion in multimodal representation learning. *ArXiv*, abs/2011.07191.

W. Chen, C. Yeo, C. Lau, and B. S. Lee. 2018. Leveraging social media news to predict stock index movement using rnn-boost. *Data Knowl. Eng.*, 118:14–24.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. Uniter: Learning universal image-text representations. *ArXiv*, abs/1909.11740.

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

X. Ding, Y. Zhang, T. Liu, and J. Duan. 2015. Deep learning for event-driven stock prediction. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*.

Marco Federici, Anjan Dutta, Patrick Forr'e, Nate Kushman, and Zeynep Akata. 2020. Learning robust representations via multi-view information bottleneck. *ArXiv*, abs/2002.07017.

Fuli Feng, Huimin Chen, Xiangnan He, Ji Ding, Maosong Sun, and Tat-Seng Chua. 2018. Enhancing stock movement prediction with adversarial training. In *International Joint Conference on Artificial Intelligence*.

S. Feuerriegel and J. Gordon. 2018. Long-term stock index forecasting based on text mining of regulatory disclosures. *Decis. Support Syst.*, 112:88–97.

Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. 2016. Multimodal compact bilinear pooling for visual question answering and visual grounding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 457–468.

Muhammad Ghulam, Fatima Alshehri, Fakhri Karray, Abdulmotaleb El Saddik, Mansour Alsulaiman, and Tiago H. Falk. 2021. A comprehensive survey on multimodal medical signals fusion for smart healthcare systems. *Inf. Fusion*, 76:355–375.

Shwai He and Shihao Gu. 2021. Multi-modal attention network for stock movements prediction. *ArXiv*, abs/2112.13593.

Guosheng Hu, Yang Hua, Yang Yuan, Zhihong Zhang, Zheng Lu, Sankha S Mukherjee, Timothy M Hospedales, Neil Martin Robertson, and Yongxin Yang. 2017a. Attribute-enhanced face recognition with neural tensor fusion networks. In *ICCV*, pages 3764–3773.

Ziniu Hu, Weiqing Liu, Jiang Bian, Xuanzhe Liu, and Tie-Yan Liu. 2017b. Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction. *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*.

Huy D. Huynh, L. Minh Dang, and Duc Duong. 2017. A new model for stock price movements prediction using deep neural network. *Proceedings of the Eighth International Symposium on Information and Communication Technology*.

Kyoungjae Kim. 2003. Financial time series forecasting using support vector machines. *Neurocomputing*, 55:307–319.

Michelle A. Lee, Yuke Zhu, Krishna Parasuram Srinivasan, Parth Shah, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Jeannette Bohg. 2018. Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks. *2019 International Conference on Robotics and Automation (ICRA)*, pages 8943–8950.

Jiwei Li, Minh-Thang Luong, and Dan Jurafsky. 2015. A hierarchical neural autoencoder for paragraphs and documents. In *Annual Meeting of the Association for Computational Linguistics*.

Q. Li, J. Tan, J. Wang, and H. Chen. 2020. A multimodal event-driven lstm model for stock prediction using online news. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1.

X. Li, H. Xie, L. Chen, J. Wang, and X. Deng. 2014. News impact on stock price return via sentiment analysis. *Knowl. Based Syst.*, 69:14–23.

Paul Pu Liang, Liu Ziyin, Amir Zadeh, and Louis-Philippe Morency. 2018. Multimodal language analysis with recurrent multistage fusion. In *Conference on Empirical Methods in Natural Language Processing*.

Tsung-Yu Lin, Aruni Roy-Chowdhury, and Subhransu Maji. 2015. Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1449–1457.

Huicheng Liu. 2018. Leveraging financial news for stock trend prediction with attention-based recurrent neural network. *ArXiv*, abs/1811.06173.

Xiao Liu, Heyan Huang, Yue Zhang, and Changsen Yuan. 2020a. News-driven stock prediction with attention-based noisy recurrent state transition. *ArXiv*, abs/2004.01878.

Zhuang Liu, Degen Huang, Kaiyu Huang, Zhuang Li, and Jun Zhao. 2020b. Finbert: A pre-trained financial language representation model for financial text mining. In *International Joint Conference on Artificial Intelligence*.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Neural Information Processing Systems*.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. *ArXiv*, abs/1606.00061.

T. Matsubara, R. Akita, and K. Uehara. 2018. Stock price prediction by deep neural generative model of news articles. *IEICE Trans. Inf. Syst.*, 101-D:901–908.

Louis-Philippe Morency, Rada Mihalcea, and Payal Doshi. 2011. Towards multimodal sentiment analysis: Harvesting opinions from the web. In *Proceedings of the 13th international conference on multimodal interfaces*, pages 169–176.

Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. 2016. Dual attention networks for multimodal reasoning and matching. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2156–2164.

W. Nuij, V. Milea, F. Hogenboom, F. Frasincar, and U. Kaymak. 2014. An automated framework for incorporating news into stock trading strategies. *IEEE Transactions on Knowledge and Data Engineering*, 26:823–835.

Pisut Oncharoen and Peerapon Vateekul. 2018. Deep learning for stock market prediction using event embedding and technical indicators. *2018 5th International Conference on Advanced Informatics: Concept Theory and Applications (ICAICTA)*, pages 19–24.

Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization. *ArXiv*, abs/1705.04304.

Verónica Pérez-Rosas, Rada Mihalcea, and Louis-Philippe Morency. 2013. Utterance-level multimodal sentiment analysis. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 973–982.

Shyam Sundar Rajagopalan, Louis-Philippe Morency, Tadas Baltruaitis, and Roland Göcke. 2016. Extending long short-term memory for multi-view structured learning. In *European Conference on Computer Vision*.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. *ArXiv*, abs/1509.00685.

R. P. Schumaker and H. Chen. 2009. Textual analysis of stock market prediction using breaking financial news: The azfin text system. *ACM Trans. Inf. Syst.*, 27:12:1–12:19.

R. P. Schumaker, Y. Zhang, C. Huang, and H. Chen. 2012. Evaluating sentiment in financial news articles. *Decis. Support Syst.*, 53:458–464.

L. Shi, Z. Teng, L. Wang, Y. Zhang, and Alexander Binder. 2019a. Deepclue: Visual interpretation of text-based deep stock prediction. *IEEE Transactions on Knowledge and Data Engineering*, 31:1094–1108.

Lei Shi, Zhiyang Teng, Le Wang, Yue Zhang, and Alexander Binder. 2019b. Deepclue: Visual interpretation of text-based deep stock prediction. *IEEE Transactions on Knowledge and Data Engineering*, 31:1094–1108.

M. R. Vargas, B. S. L. P. De Lima, and A. Evsukoff. 2017. Deep learning for stock market prediction from financial news articles. *2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, pages 60–65.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

X. Wang, Ross B. Girshick, Abhinav Kumar Gupta, and Kaiming He. 2017. Non-local neural networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7794–7803.

Yiling Wu, Shuhui Wang, Guoli Song, and Qingming Huang. 2019. Learning fragment self-attention embeddings for image-text matching. *Proceedings of the 27th ACM International Conference on Multimedia*.

Tianjun Xiao, Yichong Xu, Kuiyuan Yang, Jiaxing Zhang, Yuxin Peng, and Zheng Zhang. 2014. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 842–850.

Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. 2016. Image captioning with semantic attention. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4651–4659.

Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao, and Qi Tian. 2019. Deep modular co-attention networks for visual question answering. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6274–6283.

Amir Zadeh, Paul Pu Liang, Navonil Mazumder, Soujanya Poria, E. Cambria, and Louis-Philippe

Morency. 2018. Memory fusion network for multi-view sequential learning. In *AAAI Conference on Artificial Intelligence*.

Amir Zadeh, Rowan Zellers, Eli Pincus, and Louis-Philippe Morency. 2016. Multimodal sentiment intensity analysis in videos: Facial gestures and verbal messages. *IEEE Intelligent Systems*, 31(6):82–88.

Catching Misdiagnosed Limb Fractures in the Emergency Department Using Cross-institution Transfer Learning

Filip Rusak[1*] **Bevan Koopman**[1] **Nathan J. Brown**[2]
Kevin Chu[2] **Jinghui Liu**[1] **Anthony Nguyen**[1]
[1] The Australian e-Health Research Centre, CSIRO
[2] Emergency and Trauma Centre, Royal Brisbane and Women's Hospital
`filip.rusak@connect.qut.edu.au`
`{nathan.brown3,kevin.chu}@health.qld.gov.au`
`{bevan.koopman,jinghui.liu,anthony.nguyen}@csiro.au`

Abstract

We investigated the development of a Machine Learning (ML)-based classifier to identify abnormalities in radiology reports from Emergency Departments (EDs) that can help automate the radiology report reconciliation process. Often, radiology reports become available to the ED only after the patient has been treated and discharged, following ED clinician interpretation of the X-ray. However, occasionally ED clinicians misdiagnose or fail to detect subtle abnormalities on X-rays, so they conduct a manual radiology report reconciliation process as a safety net. Previous studies addressed this problem of automated reconciliation using ML-based classification solutions that require data samples from the target institution that is heavily based on feature engineering, implying lower transferability between hospitals. In this paper, we investigated the benefits of using pre-trained BERT models for abnormality classification in a cross-institutional setting where data for fine-tuning was unavailable from the target institution. We also examined how the inclusion of synthetically generated radiology reports from ChatGPT affected the performance of the BERT models. Our findings suggest that BERT-like models outperform previously proposed ML-based methods in cross-institutional scenarios, and that adding ChatGPT-generated labelled radiology reports can improve the classifier's performance by reducing the number of misdiagnosed discharged patients.

1 Introduction

When a patient presents to the Emergency Department (ED) with a possible limb fracture, ED clinicians order an X-ray from the radiology department. Following imaging, a radiologist authors a report stating the radiological observations and diagnosis, which is then sent back to the ED clinician requesting the procedure. Unlike radiology images, radiology reports may not be completed before a patient leaves the ED. In such cases, ED clinicians interpret radiological images themselves (Koopman et al., 2015). Occasionally, ED clinicians misdiagnose radiological evidence such as subtle limb abnormalities (e.g., small fractures, dislocations or foreign bodies), resulting in patients being discharged without appropriate treatment (Koopman et al., 2015; Zuccon et al., 2013). As a safety net, ED clinicians retrospectively reconcile radiology report findings with ED discharge diagnoses to detect potential misdiagnoses (Koopman et al., 2015). Since the radiology report reconciliation process is retrospective and performed manually, it may take several days to identify and notify a misdiagnosed patient, exposing them to potentially adverse impacts on their health (Koopman et al., 2015; Masino et al., 2016).

Machine Learning (ML)-based methods for classifying radiology reports (Koopman et al., 2015; Zuccon et al., 2013; de Bruijn et al., 2006; Zhou et al., 2014; Hassanzadeh et al., 2018b) have the potential to streamline and semi-automate the radiology report reconciliation process. However, the development of ML solutions is dependent on the availability of large and diverse labelled datasets from target hospitals for model training (Gligic et al., 2020). While radiology reports may be readily available, labelling them requires domain expertise, is time-consuming and costly (Hassanzadeh et al., 2018b). Therefore, individual departments or hospitals may not have the capacity to collect sufficiently large datasets of labelled radiology reports to conduct their own model training (Li et al., 2021a). Cross-institution transfer learning, in which datasets and model training from one institution are used to start the ML model development at another institution, may solve this problem. However, for cross-institution transfer learning to be useful for developing local ML models for radiology report reconciliation, it must be resilient to interinstitutional variations in reporting styles, lan-

---

[*] Conducted this research while affiliated with CSIRO.

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 78–87
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

guage, and verbosity (Hassanzadeh et al., 2018b; Liu et al., 2022).

Many pre-trained Transformer-based language models have achieved state-of-the-art performance in various benchmark datasets (Jia, 2022; Li et al., 2021b), especially Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019). In this study, we investigated the benefits of pre-trained BERT-like models on a radiology report classification task in cross-institutional environments, where labelled data from the target institution are unavailable. Although domain-specific pre-training appears to be effective for in-domain applications (Peng et al., 2019), little is known about the impact of pre-training with different corpora on a radiology report classification task in cross-institution settings. Therefore, we focused on answering the following research questions:

RQ1 - What is the impact of pre-training on the abnormal radiology report classification task?

To answer RQ1, we chose six different BERT models pre-trained using the medical and biomedical corpora and evaluated them on cross-institution radiology report classification, based on data from three Australian hospitals. We then used the best performing model – PubMedBERT – to explore:

RQ2 - Can we train a model that is transferable between institutions without relying on samples from the target institution?

To answer RQ2, we compare PubMedBERT with previously proposed SVM and CNN-based radiology report classification models. Our observations indicate that fine-tuned PubMedBERT models are more transferable in cross-institutional settings than previously proposed SVM and CNN-based solutions. Since labelled radiology reports needed for fine-tuning are scarce (Li et al., 2022), one of the remedies to mitigate the lack of labelled samples is to utilise synthetic radiology reports, generated according to the desired class condition, to diversify the fine-tuning set and boost classification performances. In particular, the recently released ChatGPT shows impressive text generation capabilities and high potential to generate discharge summaries (Patel and Lam, 2023). In contrast to the proposal that ChatGPT be used in the context of generating high-quality discharge summaries to offload junior doctors (Patel and Lam, 2023), we investigate the benefit of using ChatGPT as an additional source of data to fine-tune abnormal radiology report classification (BERT) models. Then we aim to answer the following research question:

RQ3 - What is the impact of using ChatGPT as an additional data source of synthetic data for classification model fine-tuning?

Note that we only use ChatGPT to supplement fine-tuning data; our empirical evaluation was still performed with a carefully curated set of real radiology reports by clinicians. We found that including ChatGPT-synthesised radiology reports in fine-tuning improves abnormal radiology record classification performances.

Lastly, we examine the practical application of using pre-trained BERT-like models, fine-tuned on real and synthetic radiology reports, to classify and reconcile radiology reports with ED discharge diagnoses in a clinical environment. Reconciling radiology reports with the corresponding ICD-10 discharge diagnoses from the ED system can result in four outcomes: 1) Both Abnormal; 2) Both Normal; 3) Radiology Abnormal, ED normal; and 4) Radiology Normal, ED Abnormal. By doing so, we are answering the following research question:

RQ4 - How does adding ChatGPT synthesised radiology reports to fine-tuning dataset impact the downstream reconciliation task?

When answering RQ4, we particularly pay attention to the difference in reconciliation performance (confusion matrix) between models that included ChatGPT-synthesised reports in fine-tuning and the models that did not. We observed that the PubMedBERT classifier, fine-tuned in real and synthetic radiology records, improves the detection of misdiagnosed patients at the expense of a higher number of records that require manual clarification.

The contribution of this work is fourfold: 1) we demonstrated that pre-trained models generalise better in the case of abnormal radiology report classification in cross-institution settings; 2) we highlighted the impact of ChatGPT on fine-tuning abnormal radiology report classification; and 3) we extended the impact of ChatGPT-generated synthetic report on a downstream reconciliation task.

2 Related Work

Common challenges of supervised ML models that support clinical decisions arise from limited clinical data and the lack of their labels, especially when the model is trained with data from a single hospital (Li et al., 2021a). The lack of labelled samples from a target hospital has previously been addressed by using transfer learning

(Gligic et al., 2020), leveraging training sets with labelled (Koopman et al., 2015; Li et al., 2021a) and unlabelled (Hassanzadeh et al., 2018a) data from multiple institutions. However, there are no prior examples of scenarios in which there is no training data at all from the target hospital. In this study, we investigated whether ML automation of the radiology report reconciliation process in a target ED could rely on a training data set that originated from an entirely different hospital.

For our purposes, the model architecture of choice must be able to generalise well across institutions. Methods relying on feature engineering, such as support vector machine (SVM), naïve bayes, or random forest, are not suitable for cross-institution settings since features engineered for a dataset collected at one institution may not be the best fit for data collected at another institution (Xiao et al., 2018). This was also observed by Koopman et al. (2015) who found a significant reduction in performance (F1-Score) of up to 10–12% in SVM-based radiology report classifiers, when the training source institution was different from the target, the test institution. Hassanzadeh et al. (2018b) further demonstrated the dependency on pre-defined feature engineering by showing improved F1-score of 5-10% across hospitals when employing self-feature-extracting CNNs with feature adoption transfer. However, to achieve such improvements in performance still required training data from the target institution. Unlike SVMs and CNNs, Transformer models take advantage of the attention mechanism capable of extracting textual features (location, context, syntactic structure, and semantics), which leads to better performance (Jia, 2022). Transformer-based models, such as pre-trained BERT models, are some of the most successful deep learning (DL) models for natural language processing (NLP) across domains (Zaheer et al., 2020). Therefore, we chose pre-trained BERT-like models for the current study.

Data synthesis is one technique that can mitigate the shortage of labelled training/fine-tuning data. We determined whether ChatGPT-generated synthetic radiology reports could be used to augment training or fine-tuning datasets for the purpose of reconciliating radiological findings. Additionally, we evaluate the impact of ChatGPT-generated reports on the performance of the BERT-based abnormal radiology report classifier when ChatGPT-synthesised reports are included in the fine-tuning dataset. Although ChatGPT has already been explored for data augmentation (Dai et al., 2023), little has been studied to evaluate the impact of ChatGPT-generated *radiology reports* on increasing the performance of the BERT-based classification model, fine-tuned on real samples with and without synthetic reports.

3 Materials and Methods

ChatGPT. ChatGPT [1], recently developed by OpenAI, is one of the largest language models to date (about 175 billion parameters) based on GPT-3 (Brown et al., 2020). ChatGPT is a generative language model that is designed to generate natural language according to some input prompt. The quality of its generated language is driven in part by the extensive text it was provided as part of the training process.

Data. In this study, we used four datasets of free-text limb structure radiology reports; three acquired from the ED of three Australian public hospitals (2378 reports), and a synthetic dataset created using ChatGPT (100 reports). The hospital-acquired radiology reports comprise anonymised adult, children, and mixed (adult and children) reports from three hospitals located in southeast Queensland, Australia. Ethical approval for the acquisition of these data was granted by the Human Research Ethics Committee of the Royal Brisbane and Women's Hospital.

Real free-text radiology reports were manually assessed by two emergency medicine physicians as either "normal" (no fractures, dislocations, or foreign bodies present) or "abnormal" (fractures, dislocations, or foreign bodies present). A software tool was developed to help physicians record their interpretations and highlight the relevant portions of text in the reports. Initially, the assessors agreed on the annotations of 2,215 out of 2,378 reports. A senior physician was then asked to act as a third assessor and resolve disagreements. The dataset distribution from three hospitals (RBWH, RCH and GCH), including the number of reports, the proportion of normal and abnormal cases, the average length of words and the number of unique words in the dataset, are presented in Table 1. The Fleiss kappa (κ) of 0.85 was calculated from the initial annotations of the first two assessors, indicating a high level of inter-rater reliability.

The 100 synthetic radiology reports – 50 normal and 50 abnormal – were generated using ChatGPT

---

[1] https://openai.com/blog/chatgpt

Dataset	Description	#Reports	Normal	Abnormal	Avg. Doc.	#Unique words
RBWH	Royal Brisbane & Womens' Hospital (adult)	1480	58%	42%	52 words	1944
RCH	Royal (Brisbane) Childrens' Hospital (child)	498	66%	34%	50 words	1100
GCH	Gold Coast Hospital (adult 62% & child 38%)	400	62%	38%	27 words	558
ChatGPT	Synthetic reports generated by ChatGPT (adult)	100	50%	50%	76 words	201

Table 1: Four different datasets of radiology reports, the number of normal and abnormal cases as identified through our annotation process or conditional generation, and document length for free-text reports document-wise.

prompts listed in Appendix Table 5. To ensure the variability between the synthetic radiology reports generated, we followed the initial with additional prompts. Synthetic reports with only minimal changes (e.g., patient name, age) and the same diagnosis were discarded.

RQ1 - What is the impact of pre-training on the abnormal radiology report classification task? We evaluated the six pre-trained BERT-based models on the free-text radiology report classification task to identify abnormalities of limb structures (normal vs abnormal). Six pre-trained models were selected based on their score on the Biomedical Language Understanding and Reasoning Benchmark (BLURB) [2] at the time of conducting experiments. BLURB includes a comprehensive benchmark for PubMed-based biomedical NLP applications and a leaderboard for tracking community progress. We evaluated the following six pre-trained BERT-like models on the cross-institutional radiology report classification task: PubMedBERT (Gu et al., 2021), BERT (Devlin et al., 2019), LinkBERT (base and large) (Yasunaga et al., 2022), BioClinicalBERT (Alsentzer et al., 2019), BlueBERT (base and large) (Peng et al., 2019) and BioELECTRA (base and large) (Kanakarajan et al., 2021). These models are pre-trained on different corpora from different domains (Appendix Table 6). The difference between base and large BERT models is in the number of layers (12 vs 24), hidden layer size (768 vs 1024) and the number of self-attention heads (12 vs 16). PubMedBERT (Gu et al., 2021) is pre-trained from scratch on biomedical article corpora, including both abstracts and full-text articles, from PubMedCentral [3]. LinkBERT is a BERT-based model pre-trained on a large corpus of documents and their links (e.g., hyperlinks, citation links) to incorporate knowledge spanning across multiple documents. BioClinicalBERT is pre-trained in all MIMIC III notes. BlueBERT models were trained on pre-processed PubMed texts extracted from the PubMed ASCII code version, containing approximately 4000 million words. BioELECTRA models were pre-trained on PubMed abstracts only with biomedical domain vocabulary.

While each model was pre-trained on different corpora, we benchmarked the mentioned models to determine the impact of model pre-training on a classification task on our mixed datasets (RBWH, RCH, and GCH). Since our dataset is relatively small, consisting of only 2378 radiology reports from all three hospitals, we chose to evaluate the pre-trained models under test with 5-fold cross-validation. Each model was fine-tuned for ten epochs per fold, with a learning rate of 9e-6 and randomly selected seed of 112. We compared F scores, precision, recall, and Matthew's correlation coefficients (MCC) between the models.

RQ2 - Can we train a model that is transferable between institutions without relying on samples from the target institution? We compare a Transformer-based PubMedBERT model with the SVM and CNN models on the abnormality classification task, in a cross-institutional setting, previously reported in Koopman et al. (2015) and Hassanzadeh et al. (2018b), respectively. We selected PubMedBERT since it achieved slightly higher, but not significantly better, performance across all four metrics as a result of answering RQ1. To compare PubMedBERT with previously proposed methods (Koopman et al., 2015; Hassanzadeh et al., 2018b), we trained PubMedBERT models on data from two out of three hospitals and tested them in the remaining one. In other words, we considered the three fine-tuning/testing splits, namely 1) fine-tuning on RBWH + RCH, testing on GCH, 2) fine-tuning on RBWH + GCH, testing on RCH, 3) fine-tuning on RCH + GCH, testing on RBWH. PubMedBERT was fine-tuned for ten epochs, with the learning rate of 9e-6 and the seed value of 112, to keep it consistent with the experimental set-up in RQ1. We compared F1 scores between PubMedBERT in the current

---

[2] https://microsoft.github.io/BLURB/

[3] https://www.ncbi.nlm.nih.gov/pmc/

study, and SVM and CNN models from previous studies (Koopman et al., 2015; Hassanzadeh et al., 2018b).

RQ3 - What is the impact of using ChatGPT as an additional data source of synthetic data for classification model fine-tuning? We investigate the benefits of including synthetic reports generated by ChatGPT while fine-tuning the PubMedBERT on the radiology report abnormality classification task. We fine-tuned six PubMedBERT models on three datasets (RBWH, RCH and GCH) separately with and without synthetic reports generated by ChatGPT. The model fine-tuning was performed in consistence with the experimental setup of RQ1 and RQ2, where each model was fine-tunned for ten epochs, with the learning rate of 9e-6 and the seed value of 112. We evaluated each fine-tuned model on the remaining two real datasets (e.g., the model trained on RBWH we evaluated on RCH and GCH datasets). The model evaluation consists of an F1 score and a confusion matrix, including the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) per fine-tuned model.

RQ4 - How does adding ChatGPT synthesised radiology reports to fine-tuning dataset impact the downstream reconciliation task? To assess the impact of ChatGPT-generated reports (used in fine-tuning) on patient data reconciliation, the radiology report classification results of both models – PubMedBERT fine-tuned with and without ChatGPT-generated reports (RQ3) – were cross checked with the patient's ICD-10 discharge diagnosis of the ED. Since some of the ICD-10 codes were unavailable or missing from the data received from the ED, we performed patient data reconciliation on available 1429/1480 RBWH, 495/498 RCH and 329/400 GCH records. Following the experimental design used to address RQ3, we evaluated two groups of PubMedBERT models, fine-tuned on records from a single hospital with and without ChatGPT-generated reports, on the downstream task of automatic reconciliation of radiology reports and discharge diagnoses. The evaluation of these two fine-tuned PubMedBERT model groups was performed on the datasets from the remaining two hospitals. Based on the classification results, there were four possible combinations of the radiology report classification / ED discharge diagnosis results: 1) *Both Abnormal*; 2) *Both Normal*; 3) *Radiology Abnormal but ED Normal*; and 4) *Radiology Normal but ED Abnormal*.

Datasets	Methods	RBWH	RCH	GCH
RBWH + RCH	SVM	-	-	0.84
	CNN	-	-	0.9294
	PubMedBERT	-	-	**0.9416**
RBWH + GCH	SVM	-	0.88	-
	CNN	-	0.9367	-
	PubMedBERT	-	**0.944**	-
GCH + RCH	SVM	0.80	-	-
	CNN	0.9085	-	-
	PubMedBERT	**0.9086**	-	-

Table 2: Results (F1 scores) for a transferred SVM, CNN without transfer learning, and PubMedBERT trained on multiple sources and evaluated on a different target source. Bold numbers represent the highest F score for each target test set.

4 Experiments and Results

RQ1 - What is the impact of pre-training on the abnormal radiology report classification task? Figure 1 shows the fine-tuned means and standard deviations for F-score, precision, recall and MCC across 5-folds for each of the six pre-trained BERT-based models. Both BioELECTRA models, the base and the large models were excluded from the comparison since the models did not converge and always predicted the same (abnormal) class. Figure 1 shows that the PubMedBERT model achieves the highest performance across all four metrics (F1-score, Precision, Recall and MCC). To determine the significance of the difference in performance between models, we calculated two-sided 95% Wilson confidence intervals (Figure 1 - right). Models with confidence intervals that do not overlap are regarded significantly different at $p < 0.05$. Overlapping of the Wilson confidence intervals suggests that the performances of PubMedBERT, BERT, BioClinicalBERT, BlueBERT-base and LinkBERT (base and large) were not significantly different from each other; however, all those models performed significantly better than the BlueBERT-large model.

RQ2 - Can we train a model that is transferable between institutions without relying on samples from the target institution? The results of the abnormal report classification performance achieved by PubMedBERT models and their comparison with the earlier reported performance of the SVM and CNN models are presented in Table 2. PubMedBERT achieved comparable or higher F1-score compared to SVM and CNN in all three cross-institution fine-tuning/testing splits. In the case of the data split, where the models were fine-tuned on RBWH + RCH and tested on GCH,

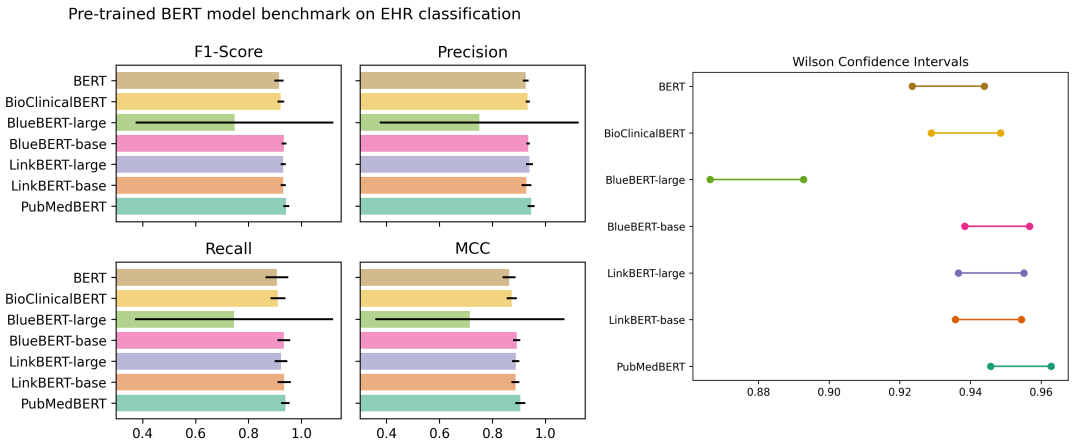


Figure 1: F1-score, Precision, Recall and Matthew's Correlation Coefficient (MCC) computed over 5-fold cross-validation (mean and standard deviation) of six pre-trained BERT models fine-tuned on a mixed dataset (RBWH, RCH, GCH). Two-sided, 95% Wilson confidence intervals for each model.

		F1-score			TP			TN			FP			FN		
		RBWH	RCH	GCH	RBWH	RCH	GCH	RBWH	RCH	GCH	RBWH	RCH	GCH	RBWH	RCH	GCH
RBWH	No ChatGPT	-	0.9477	**0.9431**	-	154	141	-	**327**	**242**	-	**4**	**7**	-	13	10
	ChatGPT	-	**0.9619**	0.9255	-	**164**	**149**	-	321	227	-	10	22	-	**3**	**2**
RCH	No ChatGPT	0.8769	-	0.9037	520	-	136	814	-	**235**	48	-	**14**	98	-	15
	ChatGPT	**0.9016**	-	**0.9201**	**545**	-	**144**	**816**	-	231	**46**	-	18	**73**	-	**7**
GCH	No ChatGPT	**0.8835**	**0.9358**	-	508	153	-	**838**	**324**	-	**24**	**7**	-	110	14	-
	ChatGPT	0.8627	0.8595	-	**550**	**159**	-	755	287	-	107	44	-	**68**	**8**	-

Table 3: Confusion matrix computed for testing cases of PubMedBERT fine-tuned on a dataset containing radiology reports from RBWH, RCH, and GCH, with and without synthetic 100 radiology reports generated by ChatGPT. The models are evaluated on the corresponding two remaining hospital radiology reports datasets by computing F1-Score, the number of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN).

PubMedBert achieved a 1.3% F1-score increase compared to CNN and a 12% F1-score increase compared to SVM. When fine-tuned on RBWH + GCH and tested on RCH, PubMedBERT achieved a 0.8% F1-score increase compared to CNN and a 7% F1-score increase compared to SVM. When the models under test were fine-tuned on GCH + RCH and tested on RBWH, PubMedBERT was similar to CNN but obtained a 14% increase in F1 compared to SVM. The F1 scores achieved by PubMedBERT follow the same trend as SVM and CNN, where higher F1 were achieved in training scenarios where the train set involved substantially more samples than the test set (e.g., fine-tuning on RBWH+RCH and testing on GCH). Overall, according to the obtained results presented in Table 2, PubMedBERT generalises better in the cross-institutional setting than previously proposed SVM and CNN-based models.

RQ3 - What is the impact of using ChatGPT as an additional data source of synthetic data for classification model fine-tuning? The evaluation results (F1 score, TP, TN, FP and FN) of PubMedBERT fine-tuned with and without ChatGPT-generated reports are detailed in Table 3. Compared with models without ChatGPT data, those fine-tuned with ChatGPT data resulted in more true positives (reports for which both the ML-classifier and the expert labeler indicated an abnormality was present) but also more false positives (reports for which the ML-classifier indicated an abnormality when an abnormality was not present). Conversely, the models fine-tuned with ChatGPT data resulted in fewer true negatives and fewer false negatives compared with models without ChatGPT data. This pattern appeared in all training scenarios except when the model was trained on reports from RCH and tested on real reports from RBWH, whereby the models with ChatGPT data resulted in more true negatives and fewer false positives.

Although these trade-offs do not manifest as a clear improvement in metrics such as the F1-score (Table 3), the observed trade-off trend has important implications on the downstream task considered here of automated abnormality classification from radiology reports. The role of an ML-based classifier in practice would be to automatically shortlist or highlight all reports that indicate the presence of a radiological abnormality. This would allow ED clinicians to focus on the "abnormal" reports and conduct a more efficient reconciliation process. A model that generates high numbers of true positives and true negatives, while keeping the number of false negatives (potential missed abnormalities) and false positives low is desirable, and our output is consistent with this. Despite relatively low numbers of false positives and false negatives, the high true positive and true negative cases could help to significantly reduce the manual report reconciliation burden on ED clinicians. According to Table 3, fine-tuning PubMedBERT on real reports plus ChatGPT-generated synthetic leads to much lower number of FN than using real reports alone. For example, we saw a 25.5% reduction in FN when *training on RCH and testing on RBWH* and a 76. 9% reduction when *training on RBWH and testing on RCH*.

RQ4 - How does adding ChatGPT synthesised radiology reports to fine-tuning dataset impact the downstream reconciliation task? The results obtained, detailed in Table 4, suggest the same trend observed when answering RQ3. Table 4 reveals the trade-off between the ability of the models to reconcile discharge diagnosis with greater disagreement between abnormal radiology classification outcome and normal ED discharge diagnosis (PubMedBERT fine-tuned with ChatGPT-generated reports); or normal radiology classification outcome and abnormal ED discharge diagnosis (PubMedBERT fine-tuned without ChatGPT-generated reports). The consequences of the reconciliation disagreement between these two model groups impact patients in the retrospective review process of the ED differently. The automatic classification outcomes from models fine-tuned with real radiology report only result in a lower number of reports that require manual processing by a clinician but a higher number of misdiagnosed discharged patients. In contrast, automatic classification results from models fine-tuned with real and ChatGPT-generated reports result in a higher number of radiology reports that require manual processing by a clinician and a lower number of misdiagnosed discharged patients. On average, across the six testing scenarios, for a 48.38% higher number of reconciliation disagreements between the abnormal radiology model classification outcome and normal ED discharge diagnosis (requiring manual review), the number of actual misdiagnosed reconciliation cases is 15.35% lower. This implies a lower number of disagreements between normal radiology model classification outcome and abnormal ED diagnosis. Since the severity and cost of misdiagnosis in undiscovered patients can be higher than the cost of a manual retrospective review of radiology reports, PubMedBERT models fine-tuned on the combination of real and ChatGPT-generated reports achieve higher performance than PubMedBERT models fine-tuned on real reports only.

5 Discussion and Conclusion

We determined that PubMedBERT was the best-performing of six pre-trained BERT-like models for classifying free-text radiology reports of X-rays for suspected limb fractures in ED patients. Compared to SVM and CNN models, PubMedBERT had better performance (measured by F1-score) for classifying radiology reports when training data and testing data were from different hospitals, suggesting that PubMedBERT has better transferability in cross-institution settings, especially in a low-data regime where the data from the target hospital is unavailable.

We also found that PubMedBERT models, which included some ChatGPT-generated synthetic radiology reports in fine-tuning, resulted in higher numbers of true positives and false positives and lower numbers of true negatives and false negatives than models without synthetic reports. The trade-off in detecting more true positives, using the model enhanced by ChatGPT data, is that there were also more false positives. While this implies that more patients with misdiagnoses would be identified, it also increases the number of reports that must be manually reconciled. This is an important observation in the reconciliation process since the higher number of FPs has less severe consequences on reconciliation than the higher number of FNs. This is because every FP-classified radiology report would require manual clarification, and every FN-classified report stands for a misdiagnosed case. Nevertheless, if all radiology reports are requried to be reviewed, as is done in current practice, our approach to reconciliation can allow patient cases to be prioritised for clinical follow-up such that suspected misdiagnosed cases would be prioritised for manual review.

To address the issue of data imbalance, it is common in the literature to perform over- or under-sampling when developing prediction models (Hassanzadeh et al., 2014; van den Goorbergh et al.,

		Both Abnormal			Both Normal			Radiology Abnormal, ED Normal			Radiology Normal, ED Abnormal		
		RBWH	RCH	GCH	RBWH	RCH	GCH	RBWH	RCH	GCH	RBWH	RCH	GCH
RBWH	No ChatGPT	-	126	109	-	**302**	**148**	-	**29**	**20**	-	38	52
	ChatGPT	-	**129**	**118**	-	289	140	-	42	28	-	**35**	**43**
RCH	No ChatGPT	357	-	109	**806**	-	**148**	**183**	-	**20**	83	-	52
	ChatGPT	**366**	-	**115**	792	-	144	197	-	24	**74**	-	**46**
GCH	No ChatGPT	347	126	-	**831**	**299**	-	**158**	**32**	-	93	38	-
	ChatGPT	**356**	**134**	-	718	265	-	271	66	-	**84**	**30**	-

Table 4: Reconciliation results encapsulate the agreement between ED discharge diagnosis and radiology report classification model results, where the agreement between the two falls into one of the four categories: 1) Both Abnormal, 2) Both Normal, 3) Radiology Abnormal, ED Normal, and 4) Radiology Normal, ED Abnormal. Two radiology report classification models were compared, the radiology report classifier where ChatGPT-generated reports were and were not used in fine-tuning. The bold numbers represent the better performing model based on the reconciliation outcome.

2022). Using synthetic data generated from ChatGPT can be viewed as another approach to augment modeling by changing the data distribution. We demonstrate that using synthetic data reduces the number of unwanted predictions, such as false negatives. This shows that augmenting with ChatGPT has a similar effect to balancing the data distribution by increasing the sample size of rare classes, in this case the abnormal diagnoses.

Overall, when developing a solution for automated reconciliation of radiology reports and discharge diagnoses in a setting where labelled radiology reports from the target institution are unavailable, pre-trained transformer models such as PubMedBERT fine-tuned on available labelled reports from partner institutions, together with ChatGPT-synthesised radiology reports can boost the automatic reconciliation performance. As we showed the promise of using NLP models to facilitate diagnosis reconciliation for ED clinicians, more works may investigate similar approaches to streamline the manual review process, flag mismatches, and explore workflow integration.

Acknowledgements

We thank the anonymised reviewers for their thoughtful comments to help improve the manuscript during the review process.

References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In *Clinical Natural Language Processing Workshop*, pages 72–78.

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 *NeurIPS*, volume 33, pages 1877–1901.

Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, Wei Liu, Ninghao Liu, Sheng Li, Dajiang Zhu, Hongmin Cai, Lichao Sun, Quanzheng Li, Dinggang Shen, Tianming Liu, and Xiang Li. 2023. AugGPT: Leveraging ChatGPT for text data augmentation.

Berry de Bruijn, Ann Cranney, Siobhan O'Donnell, Joel D Martin, and Alan J Forster. 2006. Identifying wrist fracture patients with high accuracy by automatic categorization of x-ray reports. *Journal of the American Medical Informatics Association: JAMIA*, 13(6):696–698.

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 NAACL*, pages 4171–4186.

Luka Gligic, Andrey Kormilitzin, Paul Goldberg, and Alejo Nevado-Holgado. 2020. Named entity recognition in electronic health records using transfer learning bootstrapped neural networks. *Neural networks: the official journal of the International Neural Network Society*, 121:132–139.

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

Hamed Hassanzadeh, Tudor Groza, Anthony Nguyen, and Jane Hunter. 2014. Load balancing for imbalanced data sets: Classifying scientific artefacts for evidence based medicine. In *PRICAI 2014: Trends in Artificial Intelligence*, pages 972–984. Springer International Publishing.

Hamed Hassanzadeh, Mahnoosh Kholghi, Anthony Nguyen, and Kevin Chu. 2018a. Clinical document classification using labeled and unlabeled data across hospitals. *AMIA Annual Symposium proceedings*, 2018:545–554.

Hamed Hassanzadeh, Anthony Nguyen, Sarvnaz Karimi, and Kevin Chu. 2018b. Transferability of artificial neural networks for clinical document classification across hospitals: A case study on abnormality detection from radiology reports. *Journal of biomedical informatics*, 85:68–79.

Keliang Jia. 2022. Sentiment classification of microblog: A framework based on BERT and CNN with attention mechanism. *Computers & Electrical Engineering*, 101:108032.

Kamal Raj Kanakarajan, Bhuvana Kundumani, and Malaikannan Sankarasubbu. 2021. BioELECTRA:Pretrained biomedical text encoder using discriminators. In *Proceedings of the 20th Workshop on Biomedical Language Processing*, pages 143–154, Online. Association for Computational Linguistics.

Bevan Koopman, Guido Zuccon, Amol Wagholikar, Kevin Chu, John O'Dwyer, Anthony Nguyen, and Gerben Keijzers. 2015. Automated reconciliation of radiology reports and discharge summaries. *AMIA Annual Symposium proceedings*, 2015:775–784.

Irene Li, Jessica Pan, Jeremy Goldwasser, Neha Verma, Wai Pan Wong, Muhammed Yavuz Nuzumlalı, Benjamin Rosand, Yixin Li, Matthew Zhang, David Chang, R Andrew Taylor, Harlan M Krumholz, and Dragomir Radev. 2022. Neural natural language processing for unstructured data in electronic health records: A review. *Computer Science Review*, 46:100511.

Jin Li, Yu Tian, Runze Li, Tianshu Zhou, Jun Li, Kefeng Ding, and Jingsong Li. 2021a. Improving prediction for medical institution with limited patient data: Leveraging hospital-specific data based on multicenter collaborative research network. *Artificial intelligence in medicine*, 113:102024.

Pengfei Li, Peixiang Zhong, Kezhi Mao, Dongzhe Wang, Xuefeng Yang, Yunfeng Liu, Jianxiong Yin, and Simon See. 2021b. ACT: an attentive convolutional transformer for efficient text classification. *Proceedings of 2021 AAAI Conference*, 35(15):13261–13269.

Jinghui Liu, Daniel Capurro, Anthony Nguyen, and Karin Verspoor. 2022. "Note Bloat" impacts deep learning-based NLP models for clinical prediction tasks. *Journal of biomedical informatics*, 133:104149.

Aaron J Masino, Robert W Grundmeier, Jeffrey W Pennington, John A Germiller, and E Bryan Crenshaw, 3rd. 2016. Temporal bone radiology report classification using open source machine learning and natural langue processing libraries. *BMC medical informatics and decision making*, 16:65.

Sajan B Patel and Kyle Lam. 2023. ChatGPT: the future of discharge summaries? *The Lancet. Digital health*, 5(3):e107–e108.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *BioNLP Workshop and Shared Task*, pages 58–65.

Ruben van den Goorbergh, Maarten van Smeden, Dirk Timmerman, and Ben Van Calster. 2022. The harm of class imbalance corrections for risk prediction models: illustration and simulation using logistic regression. *Journal of the American Medical Informatics Association: JAMIA*, 29(9):1525–1534.

Cao Xiao, Edward Choi, and Jimeng Sun. 2018. Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review. *Journal of the American Medical Informatics Association: JAMIA*, 25(10):1419–1428.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2022. LinkBERT: Pretraining language models with document links. In *Proceedings of the 60th Conference of ACL*, pages 8003–8016, Dublin, Ireland.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2020. Big bird: transformers for longer sequences. In *NeurIPS*, 1450, pages 17283–17297.

Yihua Zhou, Per K Amundson, Fang Yu, Marcus M Kessler, Tammie L S Benzinger, and Franz J Wippold. 2014. Automated classification of radiology reports to facilitate retrospective study in radiology. *Journal of digital imaging*, 27(6):730–736.

Guido Zuccon, Amol S Wagholikar, Anthony N Nguyen, Luke Butt, Kevin Chu, Shane Martin, and Jaimi Greenslade. 2013. Automatic classification of Free-Text radiology reports to identify limb fractures using machine learning and the SNOMED CT ontology. *AMIA Joint Summits on Translational Science proceedings.*, 2013:300–304.

A Appendix Tables

Initial prompts	**Abnormal case**	**Normal case**
1	"Write an example of a limb x-ray radiology report with an abnormality."	"Write an example of a limb x-ray radiology report without abnormalities."
2	"Write an example of a limb x-ray radiology report with several abnormalities."	"Write an example of a normal limb x-ray radiology report."
3	"Write an example of a limb x-ray radiology reports with max 12 abnormalities."	"Write an example of a limb x-ray radiology reports with max 12 normal observations."
4	"Write an example of a limb x-ray radiology report with an abnormality, use lowercase abbreviations with no explanation, and no full stop after an abbreviation."	"Write an example of a normal limb x-ray radiology report, use lowercase abbreviations with no explanation, and no full stop after an abbreviation."
Auxiliary prompts		
1	"Give me another example."	
2	"Give me another example with more clinical detail."	
3	"Give me another example with more specific details."	
4	"Give me another example with more specific details, but less repetitive."	
5	"Give me another example. Use abbreviations without explanation."	

Table 5: Prompts ChatGPT was presented to obtain synthetic radiology report examples used for training. The auxiliary prompts were used to gather more diverse synthetic samples.

Pre-trained BERT model	Corpora
BERT	3,300 million words from BooksCorpus and English Wikipedia
PubMedBERT	PubMedCentral abstracts and full-text articles
LinkBERT (base and large)	A large corpus of documents and their links (e.g., hyperlinks, citation links)
BlueBERT (base and large)	PubMed texts (about 4000 million words)
BioClinicalBERT	All notes from MIMIC III
BioELECTRA (base and large)	PubMed abstracts only with biomedical domain vocabulary

Table 6: Pre-trained BERT models and training corpora.

Turning Flowchart into Dialog: Augmenting Flowchart-grounded Troubleshooting Dialogs via Synthetic Data Generation

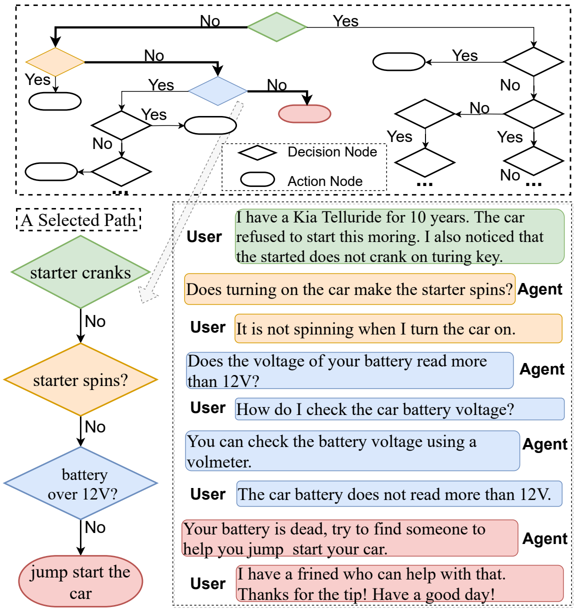
Haolan Zhan, Sameen Maruf, Lizhen Qu, Yufei Wang
Ingrid Zukerman and **Gholamreza Haffari**
Department of Data Science & AI, Monash University, Australia
{firstname.lastname}@monash.edu

Abstract

Flowchart-grounded troubleshooting dialogue (FTD) systems, which follow the instructions of a flowchart to diagnose users' problems in specific domains (e.g., vehicle, laptop), have been gaining research interest in recent years. However, collecting sufficient dialogues that are naturally grounded on flowcharts is costly, thus FTD systems are impeded by scarce training data. To mitigate the data sparsity issue, we propose a plan-based synthetic data generation (`PlanSDG`) approach that generates diverse synthetic dialog data at scale by transforming concise flowchart into dialogues. Specifically, its generative model employs a variational-base framework with a hierarchical planning strategy that includes *global* and *local* latent planning variables. Experiments on the FloDial dataset show that synthetic dialogue produced by `PlanSDG` improves the performance of downstream tasks, including flowchart path retrieval and response generation, in particular on the *Out-of-Flowchart* settings. In addition, further analysis demonstrate the quality of synthetic data generated by `PlanSDG` in paths that are covered by current sample dialogues and paths that are not covered.

1 Introduction

Flowchart-grounded Troubleshooting Dialogue (*FTD*) systems (Leake et al., 2005; Boye, 2007; Williams, 2007; Paek and Pieraccini, 2008; Janarthanam and Lemon, 2008; Wei et al., 2018; Raghu et al., 2021), which communicate with users to help them diagnose problems through the guidance of a flowchart, have been gaining interest in recent years. FTD systems face additional challenges to those faced by typical task-oriented dialogue systems (Wen et al., 2017; Budzianowski et al., 2018), e.g., FTD systems must accurately follow the instructions of a flowchart, actively detect the root cause of issues, and provide users with reasonable solutions by following an action instruction along with the *path* in a flowchart (Figure 1).



User I have a Kia Telluride for 10 years. The car refused to start this moring. I also noticed that the started does not crank on turing key.

Does turning on the car make the starter spins? **Agent**

User It is not spinning when I turn the car on.

Does the voltage of your battery read more than 12V? **Agent**

User How do I check the car battery voltage?

You can check the battery voltage using a volmeter. **Agent**

User The car battery does not read more than 12V.

Your battery is dead, try to find someone to help you jump start your car. **Agent**

User I have a frined who can help with that. Thanks for the tip! Have a good day!

Figure 1: A sample flowchart-grounded troubleshooting dialogue. Agent follows the path of a flowchart to help user diagnose problems.

Collecting sufficiently large flowchart-related dialogue corpora for FTD is challenging, since it requires domain experts with relevant knowledge. This problem also applies to a crowd-sourced FTD corpus, such as *FloDial* (Raghu et al., 2021), whose collection still involved a great deal of human effort. Despite this, the 1,789 dialogues in *FloDial* (§ 3.1) cover only 65% of the paths in the underlying flowcharts on average (Figure 2). An alternative approach to obtaining additional dialogues could involve crawling through websites. However, most of these data obtained in this manner focus on anecdotes and subjective opinions (Dai et al., 2022), and are thus unsuitable for FTD systems.

In this paper, we propose `PlanSDG`: a **Plan**-based **S**ynthetic **D**ata **G**eneration approach that generates synthetic dialogues from flowchart paths. Specifically, `PlanSDG` takes as input a path extracted from an underlying flowchart, and generates a dialogue

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 88–99
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

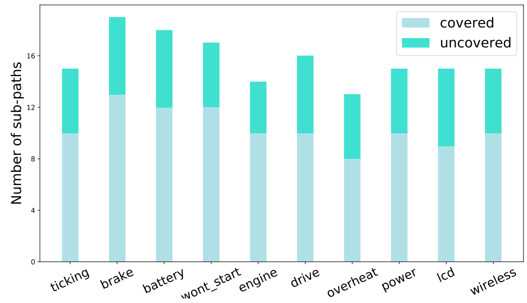


Figure 2: Statistics on the percentage (%) of (un)covered paths in the *FloDial* (containing ten flowcharts in two domains: Vehicle and Laptop) – each flowchart pertains to a specific problem. In total, more than 35% of paths are not covered by dialogue instances.

session consisting of dialogue acts and utterances. **PlanSDG** is formalised as a probabilistic generative model with structured planning latent variables, specifically *global* and *local* latent variables, that guide the generation process. The *global* latent variables are responsible for modeling the dialogue acts between the dialogue turns, providing a high-level sketch. To be able to model these global variables, we manually labeled the dialogue acts for the utterances in the *FloDial* dataset. The *local* latent variables control the diversity of generated synthetic dialogues during sentence realization.

We conducted *extrinsic* and *intrinsic* evaluations of our approach on the *FloDial* corpus, as well as follow-up ablation studies. Our extrinsic evaluation shows that the retrieval and generative models trained on the synthetic dialogues produced by **PlanSDG** achieve better performance than other augmentation methods in terms of the downstream tasks: flowchart path retrieval and response generation, particularly on the *Out-of-Flowchart* settings. Our intrinsic evaluation, which examines the quality of the synthetic dialogues, indicates that **PlanSDG** outperforms strong baseline models in term of diversity and faithfulness. Our ablation studies demonstrate the effectiveness of our proposed *global* and *local* latent planning variables. Further analysis demonstrate the quality of synthetic data generated by **PlanSDG** in *uncovered paths* that are included by flowchart but not in dialogues.

2 Plan-based Synthetic Data Generation

2.1 Task Formulation

The goal of **PlanSDG** is to take a sampled path from the flowchart, and generate a complete synthetic dialogue as well as the dialogue acts. In this paper, we only have access to a (relatively small) training set $\mathcal{T} = \{(\boldsymbol{x}, \boldsymbol{a}, \boldsymbol{y})_i\}_{i=1}^{m}$, where $\boldsymbol{x} = \{x_1, x_2, \dots, x_n\}$ is a flowchart path. A path includes tuples of nodes and edges from the flowchart. Each $x_i \in \boldsymbol{x}$ on the path corresponds to a sub-dialogue $y_i = [y_{i,0}, \cdots, y_{i,|y_i|}] \in \boldsymbol{y}$, where $y_{i,j}$ is an utterance associated with a dialogue act $a_{i,j} \in \boldsymbol{a}$. For example in the flowchart path in Figure 1, the node "battery over 12V" (x_3) corresponds to the sub-dialogue starting from the turn "Does the voltage of . . ." and ending to the turn "The car battery does not . . ." ($y_{3,0}$ to $y_{3,3}$), where each turn is associated with a dialogue act.

Given a flowchart path $\boldsymbol{x}$, our proposed data augmentation method **PlanSDG** generates synthetic dialogue acts $\hat{\boldsymbol{a}}$ and dialogues turns $\hat{\boldsymbol{y}}$, and produces the synthetic dataset $\mathcal{T}_{Syn} = \{(\boldsymbol{x}, \hat{\boldsymbol{a}}, \hat{\boldsymbol{y}})_i\}_{i=1}^{n}$ where n could be much larger than m (e.g., 10x). Our goal is that the downstream retrieval and generative dialogue models trained using $\mathcal{T} \cup \mathcal{T}_{Syn}$ outperform the models trained using only $\mathcal{T}$.

2.2 Flowchart Path Extraction

As shown in Figure 1, the flowcharts used in this paper consist of decision nodes and action nodes. The decision nodes include a question, and they are connected with other nodes by the user responses (e.g., Yes, No). The action nodes at the bottom of the flowcharts indicate the recommended actions.

For training **PlanSDG**, we directly extract the flowchart paths for the dialogues in the training set. For syntactic data generation, to ensure full coverage for the flowchart paths, we extract the flowchart paths by *Depth-First-Search* from the top decision node to the bottom action nodes. The resulting flowchart paths are then used as the inputs for **PlanSDG**.

2.3 Synthetic Dialogue Generation

PlanSDG is designed to generate diverse and high-quality synthetic dialogues from the extracted flowchart paths. Even though the input flowchart paths include textual questions, user responses and final actions, conditioning only on this information could result in tedious conversations consisting of rigid sequences of question-answer pairs. Starting from a node in a flowchart, there could be many feasible open-ended dialogues. To facilitate coverage of this dialogue space, we employ intermediate latent variables in **PlanSDG**. Dialogue acts are an intuitive choice to characterise these variables, as they describe the basic function of a dialogue

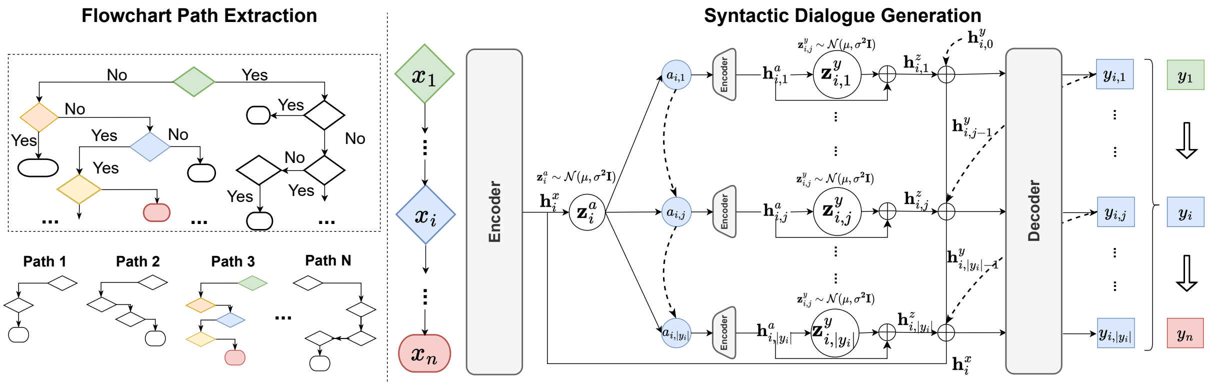


Figure 3: Detailed framework of **`PlanSDG`**, including path extraction and synthetic dialogue generation.

turn/utterance (e.g., inform, clarification), and reflect users' intentions (Stolcke et al., 2000; Bunt, 2011). We denote them by global latent variables $\boldsymbol{z}_i^a$, responsible for modeling the dialogue act transition process *over the turns*. We further introduce local latent variables $\boldsymbol{z}_{ij}^y$, responsible for generating lexically diverse utterances for each turn. As such, **`PlanSDG`** is formally a probabilistic generative model with structured latent variables (Figure 3), explained below in more details.

Global Planning over Dialogue Acts. We inject stochasticity into the global planning process using a continuous latent variable in each dialogue turn $\boldsymbol{z}_i^a$, which is assumed to follow the isotropic Gaussian distribution (Kingma and Welling, 2014). We first sample $\mathbf{z}_i^a$ from its prior distribution $p_\theta^{\boldsymbol{z}^a}(\boldsymbol{z}_i^a|x_i)$, and then generate a sequence of dialogue acts auto-represively:

$$\boldsymbol{z}_i^a \sim p_\theta^{\boldsymbol{z}^a}(\boldsymbol{z}_i^a|x_i) \quad (1)$$

$$a_{i,j} = p_\theta^a(.|a_{i,j-1}, x_i, \boldsymbol{z}_i^a) \quad (2)$$

where $p_\theta^a(a_{i,j}|a_{i,j-1}, \boldsymbol{z}_i^a, \boldsymbol{h}_i^x)$ is a 2-layer MLP with the softmax on top. We train $p_\theta^{\boldsymbol{z}^a}(\boldsymbol{z}_i^a|x_i)$ to approximate the posterior distribution $q_\phi(\boldsymbol{z}_i^a|x_i, y_i)$ using Gaussians in the training phase. The parameters in the prior and posterior distributions, $\boldsymbol{\mu}_a^p$, $\boldsymbol{\sigma}_a^p$, $\boldsymbol{\mu}_a^q$ and $\boldsymbol{\sigma}_a^q$, are parameterised as follows:

$$\begin{aligned}\boldsymbol{\mu}_a^p &= MLP_\theta^p(\boldsymbol{h}_i^x),\\ \boldsymbol{\sigma}_a^p &= softplus(MLP_\theta^p(\boldsymbol{h}_i^x)),\\ \boldsymbol{\mu}_a^q &= MLP_\phi^q([\boldsymbol{h}_i^x, \boldsymbol{h}_i^y]),\\ \boldsymbol{\sigma}_a^q &= softplus(MLP_\phi^q([\boldsymbol{h}_i^x, \boldsymbol{h}_i^y])),\end{aligned}$$

where $\mathrm{MLP}(\cdot)$ denotes a multi-layer perceptron, $\mathrm{softplus}(\cdot)$ is a smooth approximation to ReLU, which ensures positiveness. $\boldsymbol{h}_i^x = AvgPool(Enc(x_i))$ and $\boldsymbol{h}_i^y = AvgPool(Enc([y_{i,0}, \cdots, y_{i,k}]))$, which allows $\boldsymbol{z}_i^a$ to capture the global utterance information associated with x_i. Finally, the Evidence Lower Bound (ELBO) is computed as follows:

$$\begin{aligned}\mathcal{L}_{\text{global}} = &-D_{KL}(q_\phi(\boldsymbol{z}_i^a|x_i, y_i)||p_\theta^{\boldsymbol{z}^a}(\boldsymbol{z}_i^a|x_i))\\ &+\mathbb{E}_{\boldsymbol{z}_i^a \sim q_\phi}[\sum_j \log p_\theta^a(a_{i,j}|a_{i,j-1}, \boldsymbol{z}_i^a, x_i)],\end{aligned}$$

where $D_{KL}(\cdot|\cdot)$ denotes the Kullback-Leibler divergence (Kullback and Leibler, 1951).

Local Planning for Utterance Generation. Given the dialogue act $a_{i,j}$ generated from $\boldsymbol{z}_i^a$, we focus on generating lexically diverse dialogue utterances that are faithful to the flowchart. We sample $\mathbf{z}_{i,j}^y$ from its prior distribution conditioned on $a_{i,j}$ and x_i, as follows:

$$\boldsymbol{z}_{i,j}^y \sim p_\theta^{\boldsymbol{z}^y}(\boldsymbol{z}_{i,j}^y|x_i, a_{i,j}) \quad (3)$$

We train $p_\theta^{\boldsymbol{z}^y}(\boldsymbol{z}_{i,j}^y|x_i, a_{i,j})$ to approximate the posterior distribution $q_\phi(\boldsymbol{z}_{i,j}^y|x_i, a_{i,j}, y_{i,j})$, assuming that both distributions are Gaussian. They are parameterised as follows:

$$\begin{aligned}\boldsymbol{\mu}_y^p &= \mathrm{MLP}_\theta^p(\boldsymbol{h}_i^x, \boldsymbol{h}_{i,j}^a),\\ \boldsymbol{\sigma}_y^p &= \mathrm{softplus}(\mathrm{MLP}_\theta^p(\boldsymbol{h}_i^x, \boldsymbol{h}_{i,j}^a)\\ \boldsymbol{\mu}_y^q &= \mathrm{MLP}_\phi^q(\boldsymbol{h}_i^x, \boldsymbol{h}_{i,j}^a, \boldsymbol{h}_{i,j}^y),\\ \boldsymbol{\sigma}_y^q &= \mathrm{softplus}(\mathrm{MLP}_\phi^q(\boldsymbol{h}_i^x, \boldsymbol{h}_{i,j}^a, \boldsymbol{h}_{i,j}^y)),\end{aligned}$$

where $h_{i,j}^a = AvgPool(Enc(a_{i,j}))$. In contrast with global planning, here we use the ground-truth utterance $y_{i,j}$ for training to allow **`PlanSDG`** to focus on the local information. Finally, the ELBO for the local planning variable is:

$$\begin{aligned}&\mathcal{L}_{\text{local}} =\\ &-D_{KL}(q_\phi(\boldsymbol{z}_{i,j}^y|x_i, a_{i,j}, y_{i,j})||p_\theta^{\boldsymbol{z}^y}(\boldsymbol{z}_{i,j}^y|x_i, a_{i,j}))\\ &+\mathbb{E}_{\boldsymbol{z}_{i,j}^y \sim q_\phi}[\log p_\theta(y_{i,j}|y_{i,j-1}, x_i, a_{i,j}, \boldsymbol{z}_{i,j}^y)].\end{aligned}$$

PlanSDG generates each utterance $y_{i,j}$ based on $\boldsymbol{h}^z_{i,j}$, x_i and $y_{i,j-1}$, as follows:

$$y_{i,j} = Dec(\boldsymbol{h}^y_{i,j-1}, \boldsymbol{h}^x_i, \boldsymbol{h}^z_{i,j}),$$

where $\boldsymbol{h}^z_{i,k} = \text{Concat}([\boldsymbol{h}^a_{i,j}, \boldsymbol{z}^y_{i,j}])$ is the concatenation of the global and local planning variables. Enc and Dec are based on the Transformer architecture, and their parameters are initialized from a pre-trained Seq2Seq model (e.g., BART).

2.4 Training Objective

To summarise, the probabilistic generative model of **PlanSDG** performs the following steps to produce a dialogue from a flowchart path $\boldsymbol{x}$. For each $x_i \in \boldsymbol{x}$ on the path, it starts by sampling the global latent variable $\boldsymbol{z}^a_i \sim p^{\boldsymbol{z}^a}_\theta(.|x_i)$, and then iteratively samples the turns $y_{i,j}$ as follows:

- Sample the dialogue act: $a_{i,j} \sim p^a_\theta(.|a_{i,j-1}, x_i, \boldsymbol{z}^a_i)$
- Sample the local latent variable: $\boldsymbol{z}^y_{i,j} \sim p^{\boldsymbol{z}^y}_\theta(.|x_i, a_{i,j})$
- Sample the utterance: $y_{i,j} \sim p^y_\theta(.|y_{i,j-1}, x_i, a_{i,j}, \boldsymbol{z}^y_{i,j})$

Hence, the probability of generating a conversation and the corresponding dialogue acts given the flowchart path can be written as follows:

$$\begin{aligned} p_\theta(\boldsymbol{y}, \boldsymbol{a}|\boldsymbol{x}) = & \prod_i \int d(\boldsymbol{z}^a_i) p^{\boldsymbol{z}^a}_\theta(\boldsymbol{z}^a_i|x_i) \\ & \times \prod_j \int d(\boldsymbol{z}^y_{i,j}) p^a_\theta(a_{i,j}|a_{i,j-1}, x_i, \boldsymbol{z}^a_i) \\ & \times p^{\boldsymbol{z}^y}_\theta(\boldsymbol{z}^y_{i,j}|x_i, a_{i,j}) p^y_\theta(y_{i,j}|y_{i,j-1}, x_i, a_{i,j}, \boldsymbol{z}^y_{i,j}) \end{aligned} \tag{4}$$

The overall training objective of **PlanSDG** is the sum of the ELBOs: $\mathcal{L} = \mathcal{L}_{\text{global}} + \mathcal{L}_{\text{local}}$. This is based on the variational approach to overcome the challenges of integration over the latent variables in the likelihood objective (Equation 4). We use the re-parametrization trick in (Kingma and Welling, 2014) to optimise the training objective.

3 Experiments

3.1 Setup

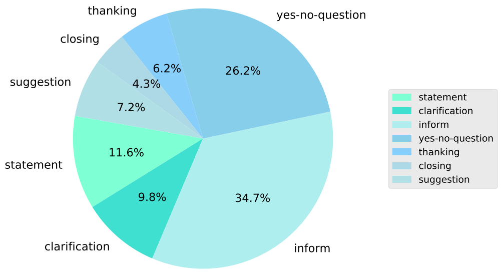


Figure 4: Statistics of dialogue act proportions in the *FloDial* dataset.

Dataset We use the *FloDial* dataset (Raghu et al., 2021) for our experiments. *FloDial* is a troubleshooting dialogue corpus containing 1,789 dialogues grounded on ten individual flowcharts[1] from two main domains: vehicle and laptop (five flowcharts in each domain). *FloDial* has two different settings: *In-Flowchart* and *Out-of-Flowchart*. In the *In-Flowchart* setting, both the training and test data are grounded on the same sets of flowcharts, while in the *Out-of-Flowchart* setting, the test dialogues are based on the flowcharts that are not included in the training stage.

Dialogue Act Labeling As the original *FloDial* dataset does not contain dialogue act labels, we manually label the dialogue act for each utterance. We investigated several widely-used dialogue act datasets, including Switchboard[2], AMI[3] and MultiWoz.[4] From these datasets, we select the most commonly used set of dialogue acts (i.e., cover 74.38% of the dialogue acts in these datasets) that are compatible with the *FloDial* dataset, including {*statement*, *inform*, *yes-no-question*, *clarification*, *thanking*, *closing*, *suggestion*}, and conduct annotation[5] for the *FloDial* dataset. Figure 4 shows the detailed statistics of the labeled dialogue acts.

Evaluation Settings In this paper, we conduct following evaluation: **1) Extrinsic Evaluation**: We aim to verify whether the synthetic data generated from the baselines and **PlanSDG** are useful for improving the performance of FTD. To precisely measure FTD performance, we use the same evaluation metrics as Raghu et al. (2021): Perplexity (PPL) and BLEU (Papineni et al., 2002) for response generation, and R@1 and R@5 for flowchart node retrieval.[6] **2) Intrinsic Evaluation**: We aim to confirm if our proposed model **PlanSDG** generate more diverse and faithful pseudo-dialogues than

---

[1] There is no path interaction or overlap between two individual flowcharts.

[2] https://catalog.ldc.upenn.edu/LDC97S62

[3] https://groups.inf.ed.ac.uk/ami/corpus/

[4] https://github.com/budzianowski/multiwoz

[5] https://github.com/zhanhl316/flowchart-dialogue-with-DA

[6] In order to diagnose problems, at each step, the agent must retrieve the most relevant node from flowchart database.

Augmentation Model	*In-Flowchart*				*Out-of-Flowchart*			
	PPL ↓	BLEU ↑	R@1 ↑	R@5 ↑	PPL ↓	BLEU ↑	R@1 ↑	R@5 ↑
FloNet	4.93	19.36	0.834	0.957	17.08	9.53	0.529	0.765
EDA	5.67	19.65	0.837	0.956	16.84	9.79	0.535	0.772
Back-Tran	4.88	19.93	0.839	0.952	19.26	10.67	0.538	0.781
GPT-2	4.37	20.69	0.844	0.958	15.93	13.70	0.574	0.813
BART	4.52	21.11	0.852	0.965	12.48	13.94	0.581	0.826
`PlanSDG` w/o $\mathcal{L}_{\text{global}}$	4.61	20.75	0.847	0.963	14.25	14.17	0.583	0.829
`PlanSDG` w/o $\mathcal{L}_{\text{local}}$	4.48	21.06	0.843	0.956	**12.45**	13.83	0.579	0.832
`PlanSDG`	**4.35***	**21.18***	**0.853***	**0.968***	12.64	**14.73****	**0.609****	**0.841****
DialoGPT	4.19	20.93	0.849	0.961	14.66	12.63	0.557	0.793
BlenderBot	**4.06**	**21.26**	0.847	0.960	13.06	12.89	0.562	0.804

Table 1: Extrinsic evaluation: Performance of augmented synthetic dialogue data generated by different models in *In-Domain* and *Out-of-Domain* settings. Results are based on the augmentation of **10x** the amount of data. Scores marked with "*" and "**" respectively indicate a significance of *p-value* < 0.05 and *p-value* < 0.01 in the t-test after Benjamini-Hochberg (BH) correction for false discovery rate (Benjamini and Hochberg, 1995).

the baseline models. To investigate the quality of generated synthetic data from `PlanSDG` and other baseline models, we use ROUGE (Lin, 2004) to assess fluency, Distinct (Li et al., 2016) and Self-BLEU (Zhu et al., 2018) for diversity, and Embedding Metrics (Average, Extrema, Greedy) and BART-Score (Yuan et al., 2021) for faithfulness.

Baselines Our baseline is **FloNet** (Raghu et al., 2021) which only uses the original training data $\mathcal{T}$. Given the newly generated synthetic data $\mathcal{T}_{Syn}$ from `PlanSDG` and other synthetic data generation models, we train the same **FloNet** model with $\mathcal{T} \cup \mathcal{T}_{Syn}$ under the same set of hyper-parameters. We compare `PlanSDG` with the following synthetic data generation models:

- **EDA** (Wei and Zou, 2019) is a rule-based approach by synonym replacement, random insertion, random swap, and random deletion.
- **Back-Tran** (Sennrich et al., 2016) is the classical back translation algorithm rooted from the machine translation task.
- Generic pre-trained language models including **GPT-2** (Radford et al., 2019), **BART** (Lewis et al., 2020).
- Conversational pre-trained models including **DialoGPT** (Zhang et al., 2020b) and **BlenderBot** (Roller et al., 2021).

We use the large version for all pre-trained models. To make a fair comparison, we incorporate annotated dialogue acts for both `PlanSDG` and other synthetic data pre-trained models.

Implementation Details We utilize the state-of-the-art pre-trained text generation model BART to initialize the encoder and decoder of `PlanSDG`, for both prior and posterior, encoder and generator. For fair comparison with baseline models, we use the $\mathrm{BART}_{\text{large}}$ for our model. In preliminary experiments, we find that fine-tuning outperforms prompt-tuning (Li and Liang, 2021) for generating valid dialogue data. For training process, we use AdamW (Loshchilov and Hutter, 2019) for gradient optimization, learning rate 0.001. batch size 8 in our experiments. We fine-tune `PlanSDG` for 50 epochs and the maximum length for utterances is set to 64. To mitigate the posterior collapse issue, we adopt the KL thresholding strategy (Kingma et al., 2016) that maximizes the KL term with a constant $\beta = 0.1$[7].

3.2 Extrinsic Evaluation

Main Results Table 1 summarizes the augmentation experiment results using 10 times (10x) for both baseline data augmentation models and `PlanSDG`. In both settings, the performance of response generation and flowchart node retrieval tasks trained with the synthetic data from `PlanSDG` are boosted up, especially in the *Out-of-Flowchart* setting. Specifically, `PlanSDG` outperforms rule-based **EDA** and naive **Back-Tran** methods by a large margin, demonstrating that widely-used data augmentation methods cannot handle the FTD situations. While comparing with strong pre-trained models (e.g, GPT-2, BART), synthetic data generated by our model have better augmentation per-

---

[7] The code will be made available upon publications.

Data Size	*In-Flowchart*				*Out-of-Flowchart*			
	PPL ↓	BLEU ↑	R@1 ↑	R@5 ↑	PPL ↓	BLEU ↑	R@1 ↑	R@5 ↑
FloNet (1x)	4.93	19.36	0.834	0.957	17.08	9.53	0.529	0.765
2x Data	5.26	20.72	0.843	0.956	13.27^{**}	11.75^{*}	0.546^{**}	0.819^{**}
5x Data	4.28	21.06^{*}	0.851^{*}	0.961	15.63^{**}	14.01^{**}	0.595^{**}	0.837^{**}
10x Data	4.35^{*}	21.18^{*}	0.853^{*}	0.968	12.64^{**}	14.73^{**}	0.609^{**}	0.841^{**}

Table 2: Extrinsic performance. FloNet (1x) is the dataset of the baseline model (Raghu et al., 2021). 2x, 5x and 10x means that we extend the original *FloDial* training set with different amounts of synthetic data. Scores marked with "*" and "**" indicate a significance of $p < 0.05$ and $p < 0.01$ in the t-test with BH correction respectively.

Model	Uncovered path within flowchart			
	PPL ↓	BLEU ↑	R@1 ↑	R@5 ↑
FloNet	12.94	11.05	0.597	0.815
EDA	12.36	11.69	0.598	0.804
Back-Tran	13.67	12.18	0.608	0.827
GPT-2	9.82	14.61	0.632	0.854
BART	8.46	15.29	0.637	0.852
`PlanSDG`	$\mathbf{8.26}^{*}$	$\mathbf{15.90}^{**}$	$\mathbf{0.654}^{**}$	$\mathbf{0.868}^{*}$

Table 3: Augmentation performance on *Uncovered path* in the flowchart (*In-Flowchart* using 10x augmented synthetic data.). Scores marked with "*" and "**" indicate a significance of $p < 0.05$ and $p < 0.01$ in the t-test with BH correction respectively.

formance. We see that `PlanSDG` is more effective in the *Out-of-Flowchart* setting, though it is on-par or better than the baselines in the *In-Flowchart* setting. In the *out-of-Flowchart* setting, `PlanSDG` achieves at least 5.6% and 4.8% for BLEU and R@1 metric than baseline models. Surprisingly, model performance supported by `PlanSDG` even surpass those models supported by DialoGPT and BlenderBot which use large-scaled dialogue data for pre-training. This result suggests that with small training data, `PlanSDG` can generalize well to the domains not encountered (i.e., dialogue) in its pre-training stage.

Analysis on Synthetic Data Size Table 2 presents the augmentation performance using different size of synthetic data. FloNet (1x) only uses original training data. As shown in Table 2, the FloNet model performance keeps improving along with the data size expansion. Especially in the *Out-of-Flowchart* setting, augmentation performance improve significantly comparing to the FloNet (1x) model. These results demonstrate that `PlanSDG` can effectively learn from existing training data and produce diverse and relevant synthetic data rather than introducing noise information.

Analysis on Uncovered Path To verify the effectiveness of `PlanSDG` on uncovered path, we conduct additional experiments on a novel uncovered path setting. As discussed above, the existing training data only cover 65% of the flowchart path in the *FloDial* dataset. We split these training datasets into training (80%), as covered path, and testing (20%), as uncovered path. Table 3 summarizes the results on the uncovered path setting. `PlanSDG` achieves the best augmentation performance comparing to other augmentation baseline models. The positive results demonstrate that `PlanSDG` is capable enhance the model performance on those uncovered flowchart paths.

Ablation on Latent Variables We conduct ablation study for the components of *local* and *global* planning variables described in Section 2.3. As shown in Table 1, the elimination of *local* and *global* planning variables undermine the performance of `PlanSDG`, showing the positive contribution of these two latent variables in generating diversity and relevant synthetic data. Specifically, the ablation of *local* planning variable leads to more performance degradation than the ablation of *global* in terms of flowchart node retrieval task, showing the importance of *local* variable in controlling the diversity on sentence realization, which further impact the training on downstream tasks.

3.3 Intrinsic Evaluation

In this section, we directly verify the quality of synthetic data by using various of automatic metrics.

Automatic Metrics We show the automatic intrinsic evaluation results on synthetic dialogue in Table 4. `PlanSDG` outperforms the baselines in terms of ROUGE-L, Dist-2/3, Embedding and BART-Score. For BLEU-4 the results of `PlanSDG` are close to the baseline models. The significant improvement obtained by `PlanSDG` for Dist-2/3 indicates that our model is able to generate more

Model	BU-4 ↑	RG-L ↑	Dist-2 ↑	Dist-3 ↑	Self-B ↓	BART-S ↓	Emb (Avg/Extr/Gre) ↑
GPT-2	26.8	43.1	0.267	0.425	0.328	-2.590	88.1/68.7/84.1
BART	**29.7**	47.2	0.351	0.541	0.271	-2.164	87.2/67.5/83.3
DialoGPT	24.7	40.1	0.366	0.563	0.257	-2.328	**89.3**/62.5/82.6
BlenderBot	19.3	35.6	0.308	0.497	0.283	-2.051	82.6/59.3/78.6
w/o $\mathcal{L}_{\text{global}}$	27.3	49.1	0.382	0.574	0.249	-2.156	87.1/68.3/84.7
w/o $\mathcal{L}_{\text{local}}$	27.8	47.6	0.365	0.568	0.261	-2.321	85.7/68.2/83.8
`PlanSDG`	28.5	**51.2****	**0.397****	**0.602****	**0.225****	**-2.037***	86.1/**69.4***/**85.7****

Table 4: Intrinsic evaluation results for pseudo dialogue generation. The metrics BLEU-4, ROUGE-L, Distinct-2/3, Self-BLEU, BART-score and Embedding are abbreviated as BU-4, RG-L, Dist-2/3, Self-B, BART-S and Emb respectively. The best results are highlighted with **bold**. Scores marked with "⋆" and "⋆⋆" indicate a significance of $p < 0.05$ and $p < 0.01$ in the t-test with BH correction respectively.

diverse texts than the baselines – a result of our latent variable modeling. The high scores of Embedding and BART-Score indicate that our model also has the capacity to generate utterances that are semantically coherent with the input flowchart.

Ablation on Latent Variables We first show the ablation study of different training objectives in Table 4. We observe a certain performance drop when removing global planning latent variable $\mathcal{L}_{\text{global}}$ or local planning latent variable $\mathcal{L}_{\text{local}}$ during fine-tuning. Specifically, the removal of $\mathcal{L}_{\text{local}}$ results in a significant drop in Dist-2/3 metric, showing that the local planning latent variable, together with dialogue act, is responsible for utterance diversity. We then highlight that the significance of dialogue act plays an important role in high-level sketch. The absence of $\mathcal{L}_{\text{global}}$ also results in a drop of performance in terms of BLEU-4, RG-L and Dist-2/3, showing that global planning latent variable play an important role in both relevance and diversity of the generated synthetic data. Thus, the combination of $\mathcal{L}_{\text{global}}$ and $\mathcal{L}_{\text{local}}$ guarantees the quality of generated synthetic dialogues.

3.4 Case Study

In this section, we conduct a case study towards the output of `PlanSDG` when given covered path and uncovered path as inputs, respectively. The output examples are given in Table 5.

Covered path We first focus on the `PlanSDG` outputs (Gen-1 and Gen-2) for covered flowchart path. By interpolating the latent variables, `PlanSDG` is able to generate diverse utterances along with the corresponding dialogue act sequences. For instance, in addition to generating a series of "yes-no-question" dialogue acts during the problem diagnosis process, `PlanSDG` can incorporate other dialogue acts as well, such as " clarification " and " suggestion ". This will further lead to the diversity in their corresponding utterance generation. In addition, although being diverse, both example dialogues still maintain high faithfulness towards the input flowchart path.

Covered Flowchart paths : car won't start → starter crank? Yes → Engine fires? No → Spark to plugs? No → Spark from coil? No → 12V+ at coil primary? No → Ignition system wiring, voltage regulator.

Gen-1:

user: My family and children were very happy for the week end trip in my Honda CR-V. When everything was packed and we got seated the car would not simply start. **(statement)**
agent: After the key is turned does the starter crank? **(yes-no-q)**
user: Yes, The starter cranks when the key is turned **(inform)**
agent: When you turn the key, does the engine fire right up? **(yes-no-q)**
user: No, When the key is turned the engine does not fire up. **(inform)**
agent: Are the sparks able to reach the plug? **(yes-no-q)**
user: How do I check if it is able to reach the plug? (**clarification**)
agent: Hold the metal clip of the spark plug wire with a screw driver after pulling it and ask your assistant to crank it over. A bright spark will be visible to you. (**suggestion**)
user: No, I'm not seeing any evidence of that. The sparks aren't reaching the plug. **(inform)**
agent: Did spark come out of the ignition coil? **(yes-no-q)**
user: There is no spark near the ignition coil. **(inform)**
agent: Does the voltmeter reading show 12 volt+ at the primary coil? **(yes-no-q)**
user: No, the voltmeter reading does not show 12 volt + at the primary coil. **(inform)**
agent: There are a couple of suggestions. Either your car ignition system wiring or the voltage regulator is at fault. **(suggestion)**

Gen-2:

user: I am unable to start my Lincoln Continental. I think the problem is major, because I've only had the car for a week. **(statement)**
agent: Does turning the key make the starter crank? **(yes-no-q)**
user: It cranks when I turn the key. **(inform)**
agent: Can you tell me if the engine fires on turning the key? **(yes-no-q)**
user: The engine doesn't fire on turning the key. **(inform)**
agent: Is the spark reaching the plugs? **(yes-no-q)**
user: Spark isn't reaching the plugs. **(inform)**
agent: Is there any kind of spark from the ignition coil? **(yes-no-q)**
user: It's not coming from there. **(inform)**
agent: Is the reading more than 12 volt at the primary coil? **(yes-no-q)**
user: No, the reading is not more than 12 volt at the primary coil. **(inform)**
agent: The problem is the car ignition system wiring or voltage regulator is not working. Hope I was helpful. **(suggestion)**

Uncovered Flowchart paths : laptop wireless problem. → See Wireless Network? No → Switch on, enabled? Yes → outer active, in range? No → Get router wireless activity, LED blinking, try laptop in same room

user: I am unable to connect to the wireless network with my Lenovo. Any solution I can fix this? **(statement)**
agent: Can you see the wireless network you want to connect to? **(yes-no-q)**
user: How do I check if my laptop to see the wireless network? **(clarification)**
agent: Click on the network icon, then you can see it. **(Suggestion)**
user: Sorry, man, it isn't showing up. **(inform)**
agent: Ok, Is the wireless adapter switched ON? **(yes-no-q)**
user: Yes, of course. It's switched on. **(Inform)**
agent: Is the router in range and active? **(yes-no-q)**
user: No, I checked my network router and it's within range and working. **(inform)**
agent: Make sure the router is turned ON and its wireless activity LED is blinking. You should be able to connect in this way. **(suggestion)**

Table 5: Pseudo dialogue generation case by our `PlanSDG` model, including covered path and uncovered path.

Model	Faith.	Rel.	Info.	*kappa*
EDA	1.37	1.85	2.09	0.64
Back-Tran	1.62	2.27	2.18	0.59
GPT-2	2.24	2.53	**2.65**	0.56
BART	2.19	2.59	2.16	0.59
PlanSDG	**2.33**	**2.60**	2.54	0.57

Table 6: Human Evaluation. Annotators are required to judge each instances individually generated by baselines and our model.

Uncovered Paths As only 65% flowchart paths are covered in the *FloDial* training data, we conduct a further qualitative analysis to explore whether **PlanSDG** can generate acceptable synthetic dialogues for those *uncovered* paths. As shown in the bottom case in Table 5, we can tell from the example that basic requirements such as fluency, naturalness, and faithfulness have been fulfilled. We hypothesise that, through fine-tuning on those covered dialogue instances, dialogue systems trained on **PlanSDG** augmented data acquire and memorize relevant domain knowledge in flowcharts. Therefore, these dialogue systems will likely to have better performance compared to the ones which have not seen training data instances for the uncovered flowchart paths.

3.5 Human Evaluation

We have shown that our proposed **PlanSDG** method can achieve better performance in both extrinsic and intrinsic evaluations. However, the automatic metrics do not necessarily reflect human preference of the generated text. We therefore select 150 output samples for each baseline synthetic models and **PlanSDG** model. For each individual sample, we ask three annotators to judge from three aspects: *Faithfulness*, *Relevance* and *Informativeness*. The scale ranges from 0 (low) to 3 (high). Table 6 summarizes human evaluation results. The kappa scores indicate that the annotators came to a fair agreement in the judgement. Compared to baseline models, our **PlanSDG** approach achieves higher performance on its generated synthetic dialogues. Thus, synthetic data from **PlanSDG** also aligns well with human preferences.

4 Related Work

4.1 Troubleshooting Dialogue Systems

Troubleshooting dialogues typically appear in problem-solving scenarios between a novice and an expert (Boye, 2007; Williams, 2007; Janarthanam and Lemon, 2008). In such scenarios, experts with domain knowledge help novices by asking a series of questions to identify the problem, while the novice mostly supplies answers. Recently, Wei et al. (2018) built an end-to-end system for patient diagnosis, and a flowchart-grounded troubleshooting dialogue scenario was proposed by (Raghu et al., 2021). However, these methods are only explored in limited domains and datasets (e.g., computer, car), while **PlanSDG** is a general approach to synthesize pseudo dialogues.

4.2 Data Augmentation for Dialogue

Data augmentation for dialogue-related tasks has been explored in several previous works: Quan and Xiong (2019) presented sentence and word-level data augmentation approaches for end-to-end task-oriented dialogues; Hou et al. (2018) presented a seq2seq framework to augment dialogue utterances for dialogue language understanding, including a ranking system to produce diverse utterances; Zhang et al. (2020a) proposed a Multi-Action Data Augmentation (MADA) model, which uses dialog states to summarize the dialog history, and then maps dialog states to their system actions. Data augmentation methods for spoken dialogue and language understanding, including generative latent variable models, were investigated in (Hou et al., 2018; Kim et al., 2019; Yoo et al., 2019). However, most of the previous works focus on data augmentation for discriminative tasks. Kann et al. (2022) used retrieval-based data augmentation to improve response generation performance in open-domain dialogues, which heavily rely on relevant external resource. Given the limited relevant external resource in FTD, the retrieval-based data augmentation method cannot be applied for FTD systems.

5 Conclusions

In this paper, we explore the synthetic dialogue generation as a data augmentation approach with pre-trained model for flowchart-grounded troubleshooting dialogue systems. In further, in order to incorporate dialogue-specific features efficiently, we present a planning-based generative model **PlanSDG** for generating synthetic dialogues on troubleshooting dialogue task. The generated augmented dataset is then used to train an FTD systems. Experiments on the *FloDial* benchmark show the effectiveness of our proposed method. In the future, we plan to generalise our method to more complex dialogues, and apply it to other tasks.

Ethics Statement

We emphasize several ethical consideration in this work. First, we would like to achknowledge the efforts of crowd-workers and annotators throughout the dataset annotation and human evaluation processes. This study underwent a thorough review and received approval from an internal board. Every annotator received a compensation of 25 AUD per hour during the annotation and evaluation stages. The associated dataset is strictly for research purpose only.

References

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.

Johan Boye. 2007. Dialogue management for automatic troubleshooting and other problem-solving applications. In *Proceedings of the 8th SIGdial Workshop on Discourse and Dialogue*, pages 247–255, Antwerp, Belgium. Association for Computational Linguistics.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Harry Bunt. 2011. The semantics of dialogue acts. In *Proceedings of the Ninth International Conference on Computational Semantics (IWCS 2011)*.

Zhuyun Dai, Arun Tejasvi Chaganty, Vincent Y Zhao, Aida Amini, Qazi Mamunur Rashid, Mike Green, and Kelvin Guu. 2022. Dialog inpainting: Turning documents into dialogs. In *International Conference on Machine Learning*, pages 4558–4586. PMLR.

Yutai Hou, Yijia Liu, Wanxiang Che, and Ting Liu. 2018. Sequence-to-sequence data augmentation for dialogue language understanding. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1234–1245, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Srinivasan Janarthanam and Oliver Lemon. 2008. User simulations for online adaptation and knowledge-alignment in troubleshooting dialogue systems. *Semantics and Pragmatics of Dialogue (LONDIAL)*, page 45.

Katharina Kann, Abteen Ebrahimi, Joewie Koh, Shiran Dudy, and Alessandro Roncone. 2022. Open-domain dialogue generation: What we can do, cannot do, and should do next. In *Proceedings of the 4th Workshop on NLP for Conversational AI*, pages 148–165, Dublin, Ireland. Association for Computational Linguistics.

Hwa-Yeon Kim, Yoon-Hyung Roh, and Young-Kil Kim. 2019. Data augmentation by data noising for open-vocabulary slots in spoken language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 97–102, Minneapolis, Minnesota. Association for Computational Linguistics.

Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. 2016. Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29.

Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86.

David B Leake, Steven Bogaerts, Michael Evans, Rick McMullen, Michael Oder, and Alejandro Valerio. 2005. Using cases to support divergent roles in distributed collaboration. In *FLAIRS Conference*, pages 117–122.

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.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California. Association for Computational Linguistics.

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.

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.

Tim Paek and Roberto Pieraccini. 2008. Automating spoken dialogue management design using machine learning: An industry perspective. *Speech communication*, pages 716–729.

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.

Jun Quan and Deyi Xiong. 2019. Effective data augmentation approaches to end-to-end task-oriented dialogue. In *2019 International Conference on Asian Language Processing (IALP)*, pages 47–52. IEEE.

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.

Dinesh Raghu, Shantanu Agarwal, Sachindra Joshi, and Mausam. 2021. End-to-end learning of flowchart grounded task-oriented dialogs. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4348–4366, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Eric Michael Smith, Y-Lan Boureau, and Jason Weston. 2021. Recipes for building an open-domain chatbot. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 300–325, Online. Association for Computational Linguistics.

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.

Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3):339–374.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. 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 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Zhongyu Wei, Qianlong Liu, Baolin Peng, Huaixiao Tou, Ting Chen, Xuanjing Huang, Kam-fai Wong, and Xiangying Dai. 2018. Task-oriented dialogue system for automatic diagnosis. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 201–207, Melbourne, Australia. Association for Computational Linguistics.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449, Valencia, Spain. Association for Computational Linguistics.

Jason Williams. 2007. Applying POMDPs to dialog systems in the troubleshooting domain. In *Proceedings of the Workshop on Bridging the Gap: Academic and Industrial Research in Dialog Technologies*, pages 1–8, Rochester, NY. Association for Computational Linguistics.

Kang Min Yoo, Youhyun Shin, and Sang-goo Lee. 2019. Data augmentation for spoken language understanding via joint variational generation. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7402–7409. AAAI Press.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Advances in Neural Information Processing Systems*, 34:27263–27277.

Yichi Zhang, Zhijian Ou, and Zhou Yu. 2020a. Task-oriented dialog systems that consider multiple appropriate responses under the same context. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9604–9611. AAAI Press.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing

Liu, and Bill Dolan. 2020b. DIALOGPT : Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278, Online. Association for Computational Linguistics.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 1097–1100. ACM.

A Appendix

A.1 Derivation of Variational Lower Bound

$$
\begin{aligned}
&\mathrm{log} p_\theta(\mathbf{a},\mathbf{y}|\mathbf{x}) \\
&= \log \int_{\mathbf{z}_a}\int_{\mathbf{z}_y} p_\theta(\mathbf{a}|\mathbf{z}_a,\mathbf{x})\cdot \\
&p_\theta(\mathbf{y}|\mathbf{z}_y,\mathbf{a},\mathbf{x})p_\phi(\mathbf{z}_y|\mathbf{a},\mathbf{x})p_\phi(\mathbf{z}_a|\mathbf{x})d_{\mathbf{z}_a} \\
&= \log \int_{\mathbf{z}_a} p_\theta(\mathbf{a}|\mathbf{z}_a,\mathbf{x})p_\phi(\mathbf{z}_a|\mathbf{x})\frac{q_\phi(\mathbf{z}_a|\mathbf{x},\mathbf{y})}{q_\phi(\mathbf{z}_a|\mathbf{x},\mathbf{y})}\cdot \\
&\int_{\mathbf{z}_y} p_\theta(\mathbf{y}|\mathbf{z}_y,\mathbf{a},\mathbf{x})p_\phi(\mathbf{z}_y|\mathbf{a},\mathbf{x})\frac{q_\phi(\mathbf{z}_y|\mathbf{x},\mathbf{a},\mathbf{y})}{q_\phi(\mathbf{z}_y|\mathbf{x},\mathbf{a},\mathbf{y})}d_{\mathbf{z}_x} \\
&= \log \int_{\mathbf{z}_a} p_\theta(\mathbf{a}|\mathbf{z}_a,\mathbf{x})p_\phi(\mathbf{z}_a|\mathbf{x})\frac{q_\phi(\mathbf{z}_a|\mathbf{x},\mathbf{y})}{q_\phi(\mathbf{z}_a|\mathbf{x},\mathbf{y})}\cdot \\
&\mathbb{E}_{q_\phi(\mathbf{z}_y|\mathbf{x},\mathbf{a},\mathbf{y})}\left[\frac{p_\theta(\mathbf{y}|\mathbf{z}_y,\mathbf{a},\mathbf{x})p_\phi(\mathbf{z}_y|\mathbf{a},\mathbf{x})}{q_\phi(\mathbf{z}_y|\mathbf{x},\mathbf{a},\mathbf{y})}\right]d_{\mathbf{z}_x} \\
&= \mathrm{log}\mathbb{E}_{q_\phi(\mathbf{z}_a|\mathbf{x},\mathbf{y})}\{\frac{p_\theta(\mathbf{a}|\mathbf{z}_a,\mathbf{x})p_\phi(\mathbf{z}_a|\mathbf{x})}{q_\phi(\mathbf{z}_a|\mathbf{x},\mathbf{y})}\cdot \\
&\mathbb{E}_{q_\phi(\mathbf{z}_y|\mathbf{x},\mathbf{a},\mathbf{y})}\left[\frac{p_\theta(\mathbf{y}|\mathbf{z}_y,\mathbf{a},\mathbf{x})p_\phi(\mathbf{z}_y|\mathbf{a},\mathbf{x})}{q_\phi(\mathbf{z}_y|\mathbf{x},\mathbf{a},\mathbf{y})}\right]\} \\
&\geq \mathbb{E}_{q_\phi(\mathbf{z}_a|\mathbf{x},\mathbf{y})}\{\mathrm{log}\frac{p_\theta(\mathbf{a}|\mathbf{z}_a,\mathbf{x})p_\phi(\mathbf{z}_a|\mathbf{x})}{q_\phi(\mathbf{z}_a|\mathbf{x},\mathbf{y})}+ \\
&\mathbb{E}_{q_\phi(\mathbf{z}_y|\mathbf{x},\mathbf{a},\mathbf{y})}\left[\frac{p_\theta(\mathbf{y}|\mathbf{z}_y,\mathbf{a},\mathbf{x})p_\phi(\mathbf{z}_y|\mathbf{a},\mathbf{x})}{q_\phi(\mathbf{z}_y|\mathbf{x},\mathbf{a},\mathbf{y})}\right]\} \\
&\approx -KL(q_\phi(\mathbf{z_a}|\mathbf{x},\mathbf{y})||p_\theta(\mathbf{z_a}|\mathbf{x})) \\
&+\mathbb{E}_{\mathbf{z_a}\sim q_\phi}[\log p_\theta(a|\mathbf{z_a},\mathbf{x})] \\
&-KL(q_\phi(\mathbf{z_y}|\mathbf{x},a,\mathbf{y})||p_\theta(\mathbf{z_y}|\mathbf{x},a)) \\
&+\mathbb{E}_{\mathbf{z_y}\sim q_\phi}[\log p_\theta(\mathbf{y}|\mathbf{x},a,\mathbf{z}_y)]
\end{aligned}
$$

Domain	Vehicle				
	ticking	brake	battery	wont_start	engine
#Dialog	178	188	196	174	168
#path	15	19	18	17	14
Domain	Laptop				
	drive	overheating	power	lcd	wireless
#Dialog	192	186	188	178	196
#path	16	13	15	15	15

Table 7: #Dialog and #sub-path denote the number of dialogue session, and the number of sub-paths of each corresponding flowchart.

A.2 Details about *FloDial* Dataset

The *FloDial* dataset is collected for the troubleshooting situations, where the interactions between user and agent are carried to diagnose user's problem in specific domain. *FloDial* contains two main domain: vehicle and laptop. Each domain contains 5 sub-problems. For each sub-problem, there is a corresponding flowchart. Dialogues are conducted based on these flowcharts. Details about each sub-problems and flowchart are shown in Table 7. *FloDial* contains 1,789 dialogue sessions in total. In the experiments of *FloDial* paper, they construct two settings: *In-Flowchart* and *Out-of-Flowchart* settings. The test set of *In-Flowchart* setting contains the dialogue in 8 sub-problems (including ticking, brake, battery, wont_start, drive, overheating, power and lcd), which maintains the same domain with training set. Beside, the test set of *Out-of-Flowchart* setting only contains 2 sub-problems (engine, wireless), while all other 8 sub-problems are treated as training set. An example of flowchart in *car_wont_start* domain is shwon in Figure 5

Besides, as the original *FloDial* dataset does not contain any dialogue act information, we manually label the dialogue act for each dialogue utterance. The selection of dialogue acts is based on the investigation on previous work, including Switchboard (`https://catalog.ldc.upenn.edu/LDC97S62`), AMI (`https://groups.inf.ed.ac.uk/ami/corpus/`), MultiWoz (Budzianowski et al., 2018) and etc. Finally, we chose seven most frequent dialogue, which also compatible with the *FloDial* dataset. These dialogue acts include: {statement, inform, yes-no-question, clarification, thanking, closing and suggestion}. The percentage of each dialogue act in the *FloDial* is: statement: 11.6%, inform: 34.7%, yes-no-question: 26.2%, clarification: 9.8%, thanking: 6.2%, closing: 4.3% and suggestion: 7.2%.

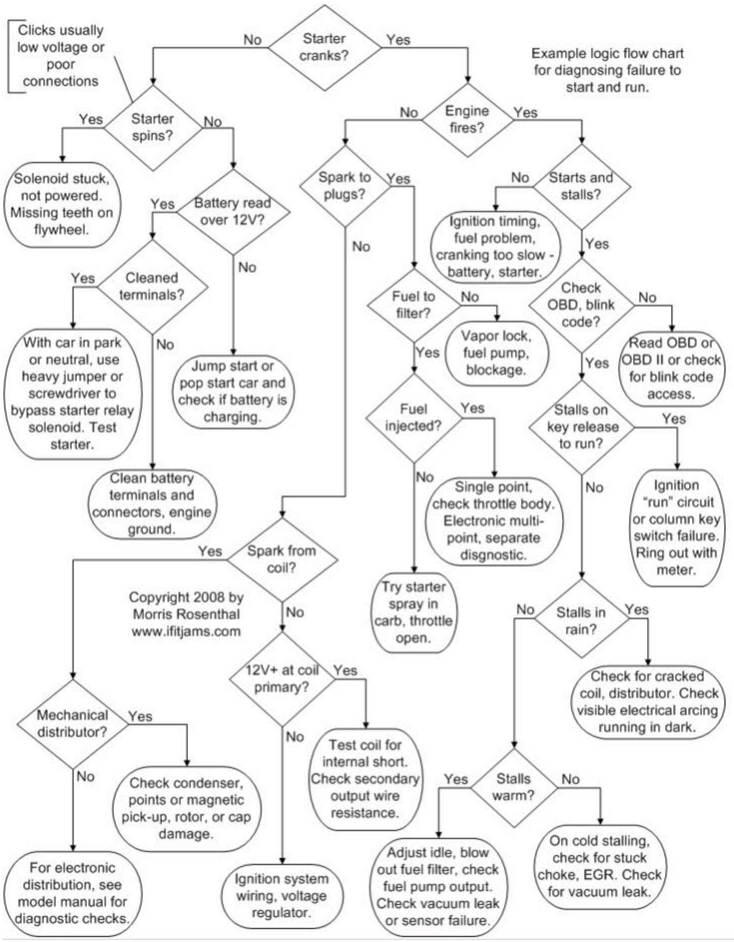

Clicks usually low voltage or poor connections
Starter cranks?
No
Yes
Example logic flow chart for diagnosing failure to start and run.
Starter spins?
Yes
No
Engine fires?
No
Yes
Solenoid stuck, not powered. Missing teeth on flywheel.
Battery read over 12V?
Yes
No
Spark to plugs?
Yes
No
Starts and stalls?
No
Yes
Ignition timing, fuel problem, cranking too slow - battery, starter.
Cleaned terminals?
Yes
No
Fuel to filter?
No
Yes
Vapor lock, fuel pump, blockage.
Check OBD, blink code?
No
Yes
Read OBD or OBD II or check for blink code access.
With car in park or neutral, use heavy jumper or screwdriver to bypass starter relay solenoid. Test starter.
Jump start or pop start car and check if battery is charging.
Fuel injected?
Yes
No
Stalls on key release to run?
Yes
No
Single point, check throttle body. Electronic multi-point, separate disgnostic.
Ignition "run" circuit or column key switch failure. Ring out with meter.
Clean battery terminals and connectors, engine ground.
Spark from coil?
Yes
No
Try starter spray in carb, throttle open.
Copyright 2008 by Morris Rosenthal www.ifitjams.com
Stalls in rain?
No
Yes
12V+ at coil primary?
Yes
No
Check for cracked coil, distributor. Check visible electrical arcing running in dark.
Mechanical distributor?
Yes
No
Test coil for internal short. Check secondary output wire resistance.
Stalls warm?
Yes
No
Check condenser, points or magnetic pick-up, rotor, or cap damage.
For electronic distribution, see model manual for diagnostic checks.
Ignition system wiring, voltage regulator.
Adjust idle, blow out fuel filter, check fuel pump output. Check vacuum leak or sensor failure.
On cold stalling, check for stuck choke, EGR. Check for vacuum leak.

Figure 5: The flowchart example of *car_wont_start* domain. The figure is directly downloaded from the website:`https://www.ifitjams.com/`, the original source of *FloDial* dataset.

Encoding Prefixation in Southern Min

Yishan Huang

The University of Sydney

yishan.huang@sydney.edu.au

Abstract

This study adopts an inter-disciplinary approach to explore how the prefixation is encoded and contributes to the word formation in Zhangzhou Southern Min, an under-described Sinitic dialect spoken in the southern Fujian of mainland China. It reveals the semantic function, morpho-syntactic characteristics, prosodic effect, pragmatic significance of prefixation in this dialect, along with their occurrence constraints. The exploration directly fills in the research gap in the study of Zhangzhou grammar, and substantially advance our knowledge of the encoding of prefixation in southern Chinese dialects. It contributes well-attested linguistic data to the typology of prefixation as an important phenomenon in the world's natural languages, while enlightening the discussion on how Sinitic languages should be better defined from the morpho-syntactic perspective.

Keywords: refixation, semantics, morpho-syntax, phonology, constraint, Zhangzhou, Southern Min

1 Introduction

Affixation broadly refers to a morphological process whereby lexical or grammatical information is added to an existing lexical base and a new lexeme, or a new form of the same lexeme is derived (Hall, 2008; Hawkins & Gilligan, 1998; McCarthy, 2002). Natural languages vary in what morphological processes are available in their grammar, how frequently the processes are used, and what types of information are encoded. For example, English expresses the plurality of nouns using suffixation (e.g., table/tables, friend/friends) (McCarthy, 2002). Zapotec, a language spoken in Oaxaca, Mexican, expresses the plurality through prefixing ka-morpheme to nominal bases (Marlett, 1985). Yoruba, a language of south-western Nigeria, uses a separate word to encode the plurality, such as the word okunrin means 'the man' and its corresponding plural form is awon 'the men' (Ajiboye, 2005). As seen, morphology is not equally prominent in spoken languages, as the same information can be encoded differently. Languages are thus classified into different types, such as analytic and synthetic, depending on the complexity of morphology and the method they employ to construct words (e.g., Banfi & Arcodia, 2007; Bybee et al., 1990; Basciano, 2017).

Sinitic languages, along with Yoruba and Vietnamese, are often cited to be isolating because of their exhibiting an extreme degree of analyticity with little affixation (Lin, 2001; Liao, 2014). However, this convention has been challenged because both inflectional and derivational affixes are continuingly discovered in the synchronic speech of those so-called isolating languages (Lin, 2001; Liao, 2014; Arcodia & Basciano, 2012; 2022). For example, in Zhangzhou Southern Min, a Sinitic dialect spoken in southern Fujian province of southeast China, the prefix *ʔɐ33* can be attached to kinship terms (*ʔɐ33-kɔŋ35 'grandfather'*) or the first name of personal names (*ʔɐ33-kun22 'address a girl whose last syllable of first name is kun 22*) to express closeness and affection. The infix *ʔɐ51* can be inserted into disyllabic bases and serve as a modifying marker (*kjɵ35-ʔɐ51-tʰɐw22 'bridge-ʔɐ51-head: a place name'*), or a coordinative marker (*dĩ35-ʔɐ51-tse41 'year-ʔɐ51-festival: traditional festivals'*). The suffix *ʔɐ51* can serve as

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 100–106
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

a purely nominal marker (*kɐm35-ʔɐ51 'orange-ʔɐ51: orange'*), a nominaliser to some verbs (*ʤɛ̃55-ʔɐ51 'to clamp-ʔɐ51: plier'*) and adjectives (*pwi35-ʔɐ51 'fat-ʔɐ51: fat person'*) or a diminutive marker to kinship terms to deliver feeling of smallness, closeness, and affection (*tsɐw55-ʔɐ51 'daughter-ʔɐ51: daughter'*). Given an increasingly high ratio of morphemes per word, it has become questionable to define these languages as being morphologically isolating.

Triggered by its intriguing affixational phenomenon, this study is devoted to exploring how prefixation is encoded and contributes to word formation in Zhangzhou Southern Min. This Sinitic dialect is under-described, because no systematic work has described its prefixing system. Fives specific research questions will be addressed, including (a) what semantic information can be encoded in the process of prefixation? (b) What morpho-syntactic consequences are induced? (c) What prosodic consequence can be evoked? (d) What pragmatic information can be conveyed by the prefixation, and (e) What linguistic factors can constrain the application of prefixation in this Sinitic dialect?

The exploration directly fills in the research gap and advances our knowledge of the encoding of prefixation in this Southern Min variety, while contributing vital linguistic data to the typology of affixation as an important morphological event not only in Sino-Tibetan language family but in human languages at the general level. The description also enlightens the discussion on how human beings employ different linguistic levels to encode and decode a particular language phenomenon in their mental grammar and language practice, while shedding light on the discussion of how Sinitic languages should be better defined.

The materials used in this study are collected from the field site by the author in 2019, and also from online consultation with native speakers over these years. The research locality is strictly limited to Xiangcheng and Longwen districts, the inner urban area of Zhangzhou, which are conventionally considered to be historically-socially-culturally-linguistically-geographically representative of Zhangzhou (Huang, 2022).

2 SEMANTICS OF PREFIXATION

Semantically, the prefixes in this dialect can be classified into either mono-semantic or polysemantic, depending on the range of lexical information that they can covey. The mono-semantic prefixes refer to those prefixes whose derived words can only cover one semantic domain. For example, as illustrated in (1), the prefix *ħwɐn35* exclusively indicates the foreign origin of the related object. For example, the lexical base *tsi22* literarily means *'potato',* while the derived word *ħwɐn33-tsi22* particularly refers *'sweet potato'*. Because sweet potatoes historically are imported from other countries, rather than being produced domestically. The prefix *ħwɐn33* indicates the foreign origin of sweet potatoes. Likewise, when attached to numbers, the prefix *te33* exclusively indicates an ordinal order of the base. For example, the lexical base *zi33* means *'two'*, while the derived word *te32-zi33* with the prefix particularly refers to 'the second'.

(1) Mono-semantics of prefixation

- **ħwɐn35 (foreign origin)**
  ħwɐn33-pɵ22 'ħwɐn33-lady: foreign lady'
  ħwɐn33-tsi22 'ħwɐn33-potato: sweet potato'
- **te33 (ordinal order)**
  te32-ʔik41 'te32-one: the first'
  te32-zi33 'te32-two: the second'
- **kɛ63.pɐk41 (internal relation)**
  kɛ63.pɐk65-ħjɐ̃35 'kɛ63.pɐk41-elder brother'
  'to address the elder son of father's brother'
  kɛ63.pɐk65-tsik41 'kɛ63.pɐk41-younger uncle'
  'the younger son of grandfather's brother'
- **pjɐw51 (external relation)**
  pjɐw35-ħjɐ̃35 'pjɐw35-elder brother'
  'the elder son of father's sister or mother's sibling'
  pjɐw35-tsik41'pjɐw35-younger uncle:
  'son of grandpa's sister or grandma's sibling'

The mono-semantic property can also be seen from the prefixes *kɛ63.pɐk41*and *pjɐw51* which can be attached to the same kinship terms but reveal different semantic readings. The prefix *kɛ63.pɐk41*, which literarily means *'next belly'*, is exclusively created in Southern Min to indicate an internal relation to one's family or clan. Such as the derived word *kɛ63.pɐk65-ħjɐ̃35*, in which the base *ħjɐ̃35* means 'elder brother', is used to address the son of one's father's brother who is elder than the addresser. On the contrary, the prefix *pjɐw51*, meaning *'surface'*, expresses an external relation to a family or a clan. For example, the derived word *pjɐw35- ħjɐ̃35* exclusively refers to the elder son of one's father's sister or mother's sibling.

On the contrary, the poly-semantic prefixes refer to those prefixes whose attachment can cover two or more semantic domains. This property is illustrated in (2). The prefix *ʔɐ35* can be attached to lexical bases of different types to transfer different semantics. Such as it can be attached to kinship terms to address one's family members who are senior in terms of age or generation with a sense of respect and closeness. For example, the derived word *ʔɐ33-ɦjẽ35* is colloquially used to address one's elder brother. It can be attached to person names of those family members who are at the same generation to deliver a feeling of closeness and affection. For example, given a person's first name contains *tsʰju33* morpheme that means *'tree'*, it is a common way to address him as *ʔɐ33-tsʰju33* that transfers a sense of closeness and affection. Likewise, the prefix *dɐw33* can be attached to kinship terms to address family members who are senior than the addresser with respect to age and/or generation with a sense of respect and closeness (*dɐw32-pɛ33 'dɐw32-father: to introduce one's father'*). It can also be attached to person names of those people at the same generation to convey a feeling of closeness and affection (*djɔk32 ʔi35.su35 → dɐw32-su35 'Su35; Si'*). Apart from these, this prefix can also be attached to an entity to express a sense of a long history *'old'*. Such as, the term *tsʰju33* itself means *'tree'*, while the derived word *dɐw32-tsʰju33* refers to *'an old tree'*.

(2) Poly-semantics of prefixation

- **Prefix ʔɐ35**
  Before kinship term (closeness and respect)
  ʔɐ33-kɔŋ35 'grandfather'
  ʔɐ33-ɓẽ51 'grandmother'
  ʔɐ33-pɐ35 'father'
  ʔɐ33-ɓẽ35 'mother'
  Before first name (closeness and affection)
  tɐn33. ʄe32.kun22 → ʔɐ33-kun22 'Qun'
  ɦwĩ22 kʰɐj33.tsʰju33 → ʔɐ33-tsʰju33 'Shu'
- **dɐw33**
  Before a kinship term (respect and closeness)
  dɐw32-pɛ35 'dɐw32-father: introduce one's dad'
  dɐw32-ɓu51 'dɐw32-mum: introduce one's mum'
  Before person name (affection and closeness)
  djɔk32 ʔi35.su35 → dɐw32-su35 'Su35; Si'
  tɐn33 kɔk65.kjɐŋ22 → dɐw32-kjɐŋ22 'Kjɐŋ22'
  Before an entity (long history)
  dɐw32-tjɐm41 'dɐw32-store: an old store'
  dɐw32-tsʰju33 'dɐw32-tree: an old tree'
- **dɐw51**
  Before number (ordering)
  dɐw35-zi33 'dɐw35-two: second eldest sibling'
  dɐw35-sẽ35'dɐw35-three: third eldest sibling'
  Before family name (closeness and respect)
  ʔɔŋ33 tsu35.zin22 → dɐw35-ʔɔŋ22 'ʔɔŋ22'
  ʔwĩ33 tsʰiŋ33.tsui51 → dɐw35-ʔwĩ22 'ʔwĩ22'

As seen, the prefixation is productive to modify the lexical meanings of existing bases and create new lexemes across several semantic domains. The process substantially enriches the inventory of local vocabulary that native speakers can choose and use in their conversations, while expanding the lexicon in their mental grammar.

3 MORPHO-SYNTAX OF PREFIXATION

Morpho-syntactically, the prefixing process seldom changes the part of speech of related item in Zhangzhou Southern Min. In most cases, the derived words are categorically the same as their corresponding bases being attached. For example, the bases that can be attached by the above-mentioned prefixes (*ɦwɐn35, te33, kɛ63.pɐk41, pjɐw51, te33, ʔɐ35, dɐw33, dɐw51*) in (1) and (2) are all nominal, so are their derived lexemes. Therefore, the prefixation process presents a categorical-preserving characteristics.

Apart from this, most of lexical bases that can undergo prefixation are dominantly nominal. Only few non-nominal bases can be prefixed to derive words of a different part of speech. This can be seen in (3), for example, the prefix *kʰɵ51*, a cognate of Mandarin *ke214* 可, can transfer verbal or nominal bases to lexemes of an adjective category to deliver a semantics of *'do-able'*, equivalent to the English suffix *'-able'*. Such as in the word *kʰɵ35-ʔɐj41*, the base *ʔɐj41* is a verb meaning *'to like, love'*, while the derived word refers to *'likable'*. In the word *kʰɵ35-kʰɐw51*, the base *kʰɐw51* is a noun meaning *'mouth'*, with this prefix, the derived word is used to describe something which is *delicious*.

(3) Categorical-changing prefixation

- **kʰɵ51**
  verbal base
  kʰɵ35-ʔɐj41 'kʰɵ35-like: likeable'
  kʰɵ35- kʰɵ41 'kʰɵ35-rely: reliable'
  nominal base
  kʰɵ35-kʰɐw51 'kʰɵ35-mouth: delicious'
  kʰɵ35-zin22 'kʰɵ35-people: lovable'
- **ɦɵ51**

verbal base
ħɵ35-tʰjɐ̃35 ‘ħɵ35-listen: sound-good’
ħɵ35-tsjɐ221 ‘ħɵ35-to eat: taste-good; delicious’
ħɵ35-sjɐ51 ‘ħɵ35-to write: good-writing’

- **ɓɐj41**

verbal base
ɓɐj35-tʰjɐ̃35 ‘ɓɐj35-to listen: sound-unpleasant’
ɓɐj35-tsjɐ221 ‘ɓɐj35-to eat: bad-tasting
ɓɐj35-kʰwɐ̃41 ‘ɓɐj35-to look: bad-looking; ugly’

The prefixes *ħɵ51* and *ɓɐj51*, which separately mean *‘good’* and *‘bad’*, are commonly attached to verbal bases to derive new lexemes of an adjective category that expresses the positive or negative characteristics of related verbal actions. For example, the derived word *ħɵ35-tsjɐ221*, with the verbal base *tsjɐ22 ‘to eat’*, is used to describe something that tastes good. Similarly, the derived word *ɓɐj35-tsjɐ221* describes something that tastes bad. As seen, the attachment of these prefixes not only changes the semantics but also the word class. However, not so many cases involve such a categorical change, instead, in most cases, the grammatical category of nominal lexical bases is maintained over the prefixation in this dialect.

4 PROSODY OF PREFIXATION

The prefixing process in Zhangzhou Southern Min does not cause any phonological alternation on their subsequent lexical bases, but all prefixes themselves are seen changing their tonal profiles to be entirely different from their corresponding forms in citation. Such a tonal alternation can be consider resulting from the operating effect of a right-dominant tone sandhi system in Zhangzhou (Huang, 2018; 2020; 2022). All tones at the non-right-most position, referred to as sandhi position, alter their realizations phonologically and phonetically, while tones at the rightmost position maintain their realization categorically similar to their citation forms, but may subject to certain variation at the phonetic level because of their sensitivity to occurring environments (Huang, 2018; 2020; 2022). So that, all prefixes are supposed to change their tonal profiles if they are attached to lexical bases to fulfil their semantics and/or morpho-syntactic functions.

What needs a further mention is that no special tone sandhi pattern is evoked over the prefixing process. But rather, all prefixes follow the general sandhi tendency to change their tonal forms. For example, tone 1 is realised as a rising [35] contour in citation but is altered to be a mid-level [33] contour in the sandhi context. Tone 3 changes its contour shape to a rising [35] from a high falling contour in citation. All tones within general morpho-syntactic phrases XP in which X may be adjective, verbal, nominal among others, are expected to change in this way (Huang, 2018; 2020). Table 1 summaries tonal changes of individual prefixes, in which the forms in bold show their corresponding sandhi forms.

Table 1. Tonal changes of individual prefixes.

Citation	**Sandhi Form**
ɗɐw33	**ɗɐw32**-ɓu51 ‘to introduce one’s mother’
ɗɐw51	**ɗɐw35-**sɐ̃35 ‘the third (of the siblings)’
ʔɐ35	**ʔɐ33-**ħjɐ̃35 ‘elder brother’
pjɐw51	**pjɐw35-**ħjɐ̃35 ‘the elder son of father's sister or mother’s sibling’
kɛ63.pɐk41	**kɛ63.pɐk65-**ħjɐ̃35 ‘the elder son of father's brother’
ʔŋ51	**ʔŋ35-**kɔ35 ‘father’s sister’
ħwɐn35	**ħwɐn33-**kjɔ̃35 ‘foreign chilli’
te33	**te32-**zi33 ‘the second’
twɐ33	**twɐ32-**tsi51 ‘the elder sister’
se41	**se63-**kɔ35 ‘husband’s younger sister’
sjɵ51	**sjɵ35-**ɓɐ̃j33 ‘younger sister’
kʰɵ51	**kʰɵ35-**ʔɐj41 ‘likeable; adorable’
ɓɐj51	**ɓɐj35-**kʰwɐ̃41 ‘ugly’
ħɵ51	**ħɵ35-**tsjɐ221 ‘delicious’

As seen, the tonal pitch of all prefixes has been changed phonetically and phonologically. Such as the prefix *ɗɐw51* has a high-falling [51] contour in citation, but when attached to lexical bases, the pitch is changed to a rising [35] contour (*ɗɐw35-sɐ̃35 ‘the third (of the siblings)’*). Similarly, the prefix *ħwɐn35* has a rising contour [35] in citation but the pitch is changed to a mid-level [33] when attached to lexical bases to denote their foreign origin (*ħwɐn33-kjɔ̃35 ‘foreign chilli’*). The tone sandhi pattern in the prefixing context is different from that occurs in other affixational environment in which tones are changed to either a rising or high-level contour, depending on their contour shape in citation (Huang, 2023).

5 Pragmatics of Prefixation

Certain prefixes are pragmatically significant to reveal certain relation within the social-cultural setting of Southern Min. This can be demonstrated by the usage of prefixes *pjɐw51*, *kɛ63.pɐk41* and

ʔŋ51. They can be attached to the same kinship terms but reveal different social relationships. As illustrated in (4), the prefix *pjɐw51* indicates an external relation to a family or a clan; the derived words are particularly used to address the descendant of (grand-) father's sisters or (grand-) mother's siblings. Such as the term *pjɐw35-ħjẽ35*, in which the base *ħjẽ35* refers to the elder brother, is specifically used to address the son of father's sister or mother's sibling who is elder than the addresser. The prefix *kɛ63.pɐk41* indicates an internal relation to one's family or clan; the derived terms are limited to address the descendant of (grand-) father's brothers. Such as the term *kɛ63.pɐk65-ħjẽ35* specifically refers to the son of father's brother who is elder than the addresser. Similarly, the prefix *ʔŋ51* expresses the core blood relationship on the paternal side. Such as the term *ʔŋ35-ħjẽ35* refers to one's blood-related elder brother who has the same parents.

(4) Pragmatics of prefixation

- **pjɐw51**

pjɐw35-ħjẽ35 'pjɐw35-elder brother'
'the elder son of father's sister or mother's sibling'
pjɐw35-tsik41 'pjɐw35-younger uncle'
'son of grandfather's sister or grandma's sibling'

- **kɛ63.pɐk41**

kɛ63.pɐk65-ħjẽ35 'kɛ63.pɐk41-elder brother'
'the elder son of father's brother'
kɛ63.pɐk65-tsik41 'kɛ63.pɐk41-younger uncle'
'the younger son of grandfather's brother'

- **ʔŋ51**

ʔŋ35-ħjẽ35 'ʔŋ35-elder brother'
'the blood-related elder brother'
ʔŋ35-tsik41 'ʔŋ35-younger uncle'
'father's younger brother (s)'

- **twɐ33**

twɐ32-pɛ41 'twɐ32-uncle'
'to address father (or husband)'s eldest brother'
twɐ32-kɔ35 'twɐ32-parental aunty'
'to address father (or husband)'s elder sister'

- **se41**

se63-kɔ35 'se63-father's sister'
'to address husband's younger sister'
se63-tsik41 'se63-father's younger brother'
'to address husband's younger brother'
se63-kim35-ʔɐ51 'se63-mother's brother's wife'

The pragmatic function can also be illustrated by the usage of the prefixes *twɐ33* and *se41* before kinship terms. The prefix *twɐ33*, lexically meaning *'big'*, is often used to indicate the elder (not necessarily needs to be the eldest) member of a group, especially within a family or a clan. On the contrary, the prefix *se41*, lexically meaning *'small'*, indicates the younger member. However, in Southern Min, housewives are discovered using exactly the same kinship words as their children to address their husband's siblings.

For example, as illustrated in (4), the word *twɐ32-kɔ35*, in which the base *kɔ35* means *'aunty on the parental side'*, can be used to refer to either father's elder sister or husband's elder sister. On the contrary, the words *se63-kɔ35* and *se63-tsik41*, which literarily mean *'younger aunty (father's younger sister)'* and *'younger uncle (father's younger brother)'*, respectively, are exclusively used by housewives to address their husband's younger siblings. The reason for existing such a special addressing is because, in the old society of Southern Min, married women are given a lower status within a family and a clan. Thus, they step down their seniority and address their husband's siblings in the same way as their children do. Such an addressing has been customized over time and is still adopted by the married women in the modern society. Therefore, as seen, the application of certain prefixation is pragmatically related by the local social culture and custom.

6 OCCURRENCE CONSTRAINT

The prefixation is productive in this dialect to expand its lexicon; however, several different constraints can be seen governing what prefixes can be used; what lexical bases can be attached, and/or what new lexemes can be generated.

6.1 Semantic constraint

The occurrence of certain prefixes is strictly limited to a particular semantic domain. For example, the prefixes *pjɐw51* (*pjɐw35-twɐ32-tsi51 'to address the daughter of father's sister or mother's sibling who is elder than the addresser'*), *kɛ63.pɐk41* (*kɛ63.pɐk65-twɐ32-tsi51 'to address the daughter of father's brother who is elder than the addresser'*) and *ʔŋ51* (*ʔŋ35-kɔŋ35 'to address the blood-related grandfather'*) can only occur before kinship terms to deliver the blood relation or the affinity by marriage to a family or a clan. On the contrary, the prefix *te33* (*te32-sã35 'te32-three: the third'*) can occur before numbers to indicate ordering.

6.2 Phonological constraint

The prefixing process in Southern Min can be constrained phonologically because there exist some cases in which only parts of the lexical bases, rather than the entire bases, can be prefixed to fulfil a particular semantic function. For example, as illustrated in (5), the prefixes *ɗɐw33*, *ɗɐw51*, *ʔɐ35* and *sjɵ51* can all be attached to person names, but the prefixes *ɗɐw51* and *sjɵ51* are typically attached to the family names, while the prefixes *ɗɐw33* and *ʔɐ35* are exclusively attached to the last syllables of the first names. If they are attached to other part of the lexical bases, the derived forms are considered to be colloquially ill-formed.

(5) Phonological constraint on prefixation

Before family name

- **ɗɐw51 (seniority and respect)**
  ʔɔŋ33 tsu35.zin22 → ɗɐw35-ʔɔŋ22 ‘ʔɔŋ22; Wang’
  tɐn33 kɐk65.kɛ̃35 → ɗɐw35-tɐn22 ‘Tɐn22; Chen’
- **sjɵ51 (diminution and affection)**
  tɐn33. ʄe32.kun22 → sjɵ35-tɐn22 ‘Chen’
  ħwĩ22 sjɔk65.kwan35 → sjɵ35-ħwĩ22 ‘Huang’

Before first name

- **ɗɐw33 (seniority and respect)**
  ɗjɔk32 ʔi35.su35 → ɗɐw32-su35 ‘Su35; Si’
  tɐn33 kɔk65.kjɐŋ22 → ɗɐw32-kjɐŋ22 ‘Kjɐŋ22’
- **ʔɐ35 (diminution and affection)**
  ħwĩ22 ʄe32.kjɐŋ22 → ʔɐ33-kjɐŋ22 ‘Qiang’
  lim33 sjɔk65.kwan → ʔɐ33-kwɐn35 ‘Juan’

For example, given a person is called *ʔɔŋ33 tsu35.zin22*, in which *ʔɔŋ22* is the family name and *tsu35.zin22* is his first name, it is grammatically well-formed to address him as *ɗɐw35-ʔɔŋ22* with a sense of respect and seniority. Similarly, the prefix *sjɵ51* can also be attached to family names to convey a sense of diminution and affection, such as *sjɵ35-tɐn22 ‘Chen’* given a person’s family name is *tɐn22*. The two prefixes can only be attached to family names, and it is grammatically ill-formed to attach them to other position of the person names.

On the contrary, the prefixes *ɗɐw33* can only be attached to the last syllables of the first names to deliver a semantics of seniority and respect. Such as, given a person is called *ɗjɔk32 ʔi35.su35* in which *ɗjɔk32* is the family name, it is grammatically well-formed and colloquially respectful to address him as *ɗɐw32-su35*, but ill-formed to call him as **ɗɐw32-ɗjɔk32* or **ɗɐw32-ʔi35.su35*. Likewise, the prefix *ʔɐ35* can only be attached to the last syllables of the first names to convey a sense of closeness and diminution, such as *ʔɐ33-kwɐn35* given a person is called *lim33 sjɔk65.kwan35*. As seen, the occurrence of certain prefixes is phonologically limited to certain syllables, and this has become an important part of the mental grammar of native speakers.

6.3. Pragmatic constraint

The occurrence and usage of certain prefixes can also be constrained by the pragmatic factor. For example, as discussed in Section 5, the derived words with the prefix *pjɐw51* are exclusively used to address descendants of (grand-) father’s sisters; and/or (grand-) mother’s siblings. The derived words with the prefix *kɛ63.pɐk41* are dominantly used to address descendants of (grand-)father’s brother(s), whereas the derived words with the prefix *ʔŋ51* only refer to those core family members who hold a blood relationship on the parental side. As well as this, the married women in Southern Min address their husband’s siblings in the same way as their children. Such as the derived word *twɐ32-kɔ35* can be used to address one’s father’s eldest sister, but also one’s husbands’ elder sister(s). These reflects the constraints from the social culture and custom in Southern Min.

7 Discussion

As discussed in this paper, the prefixing process has been developed as an important device in the word formation in Zhangzhou Southern Min. Semantically, the prefixation is constructive to create new lexemes and substantially enlarge the local vocabulary. Some prefixes are polysemantic to derive words that can cover different semantic domains, like the prefixes *ɗɐw51*, *ɗɐw33*, and *ʔɐ35*, while some prefixes are shown to be mono-semantic whose derived items can only cover one single semantic domain, such as *ħwɐn35* and te33.

Morpho-syntactically, the prefixing process, in general, presents a categorical-preserving property, because both the lexical bases and their derived words are largely nominal, the process of which does not evoke a new word class. But some prefixes in few cases, like *kʰɵ51*, *ħɵ51*, and *ɓɐj41* can change the category of the bases from a non-adjective category to adjective. Phonologically, the tonal forms of all prefixes are changed categorically over the process, because of the forcing factor of the right-dominant tone sandhi system in this Southern Min variety. Pragmatically, the usage of certain prefixes, like *pjɐw51*, *kɛ63.pɐk41*, *ʔŋ51, twɐ33* and *se41*, can reveal some particular social culture and custom of Southern Min. As seen, the prefixing can substantially affect the semantic, morpho-syntactic, and prosodic structures of this dialect. However, the encoding also subjects to several constraints from semantics, morpho-syntactics, phonology and pragmatics, reflecting a close interface between different linguistic levels in this dialect.

This study is the first to discuss the prefixation in Zhangzhou, which directly fills in the research gap of this under-described Southern Min variety. The exploration substantially stretches and advances our knowledge of how prefixation is encoded and contributes to the word formation in southern Chinese dialect, shedding an important light on the generalization of areal characteristics of prefixation within the Sino-Tibetan language family. The innovative description also contributes vital empirical data to the typology of prefixation in world's natural languages. The discussion also challenges the conventional assumption of regarding Sinitic languages as being isolating without significant affixation, while enlightening the theoretical discussion on how Sinitic languages should be better defined from the morpho-syntactic perspective, given their existing rich prefixing system in the synchronic speech.

8 REFERENCE

Andrew Carstairs McCarthy. 2002. *An Introduction to English Morphology: Words and Their Structure*. Scotland, UK: Edinburgh University Press.

Huang. 2018. *Tones in Zhangzhou: Pitch and Beyond*. Doctoral Dissertation: Australian National University.

Huang. 2022. Right-Dominant Tones in Zhangzhou: On and Through Phonetic Surface. In *d*, 21-22 November 2022, Taipei Medical University, Taiwan, China.

Huang. 2023. Suffix ʔɐ51 in Zhangzhou: An Interdisciplinary Exploration. *In Proceedings of The 24th Chinese Lexical Semantics Workshop (CLSW 2023)*. 29-31 May. Singapore.

Huang. *Tones in Zhangzhou: Pitch and Beyond. 2020*. Cambridge, UK: Cambridge Scholar Publishing.

Bianca Basciano. Morphology. 2017. In Rint Sybesma, Wolfgang Behr, Yueguo Gu, Zev Handel, James Huang and James Myers, *Encyclopedia of Chinese Language and Linguistics*, pp.104-118. Leiden, Netherlands: Brill.

Christopher J. Hall. 2008. Prefixation, Suffixation and Circumfixation. In Geert Booij, Christian Lehmann, Joachim, Kesselheim Mugdan, Wolfgang and Stavros Skopeteas (eds.) *Morphology, An International Handbook on Inflection and Word Formation. Handbooks of Linguistics and Communication Science*, 17 (1): 535-545. Berlin, Germany: De Gruyter.

Emanuele Banfi and Giorgio Francesco Arcodia. 2007. The 生 shēng/sheng Complex Words in Chinese between Morphology and Semantics. In *Proceedings of the Sixth Mediterranean Morphology Meeting. Morphology and Dialectology*, 6:190-204.

Giorgio Francesco Arcodia and Bianca Basciano. 2012. On the Productivity of the Chinese Affixes −兒 −r, −化 −huà and −頭 −tou. *Taiwan Journal of Linguistics*, 10(2): 89-118.

Giorgio Francesco Arcodia and Bianca Basciano. 2022. *Chinese Linguistics: An Introduction*. Oxford, UK: Oxford University Press.

Hillary Chappell. 2019. A Sketch of Southern Min Grammar. In Alice Vittrant and Justin Watkins (eds). *The Mainland Southeast Asia Linguistic Area*, pp.176-233, Berlin, Germany: De Gruyter Mouton.

Huei Ling Lin. 2001. The Interface between Syntax and Morphology: Taiwanese Verbal Complexes. In *Proceedings of the 16th Pacific Asia Conference on Language, Information and Computation*, pp.308–319, Jeju, Korea. The Korean Society for Language and Information.

Joan. L. Bybee, William Pagliuca, and Revere D Perkins. 1990. On the Asymmetries in the Affixation of Grammatical Material. In William Croft, Keith Denning and Suzanne Kemmer (eds), *Studies in Typology and Diachrony: Papers Presented to Joseph H. Greenberg on his 75th Birthday*, pp.1-42. Amsterdam, Netherland: John Benjamins.

John A. Hawkins and Gary. Gilligan. 1988. Prefixing and Suffixing Universals in Relation to Basic Word Order. *Lingua*, 74 (2-3): 219-259.

Ọládiípọ̀ Jacob Ajiboye. 2005. *Topics on Yoruba Nominal Expression*. Doctoral thesis: The University of British Columbia.

Stephen Marlett. 1985. Some Aspects of Zapotecan Clausal Syntax. *Work papers of the summer institute of linguistics*, University of North Dakota Session, 29, Article 6.

Wei-wen Roger Liao. 2014. Morphology. In C.-T. James Huang, Y.-H. Audrey Li and Andrew Simpson (eds). *The Handbook of Chinese Linguistics*, pp.3-25, Hoboken, New Jersey, United States: John Wiley & Sons.

An Ensemble Method Based on the Combination of Transformers with Convolutional Neural Networks to Detect Artificially Generated Text

Vijini Liyanage and **Davide Buscaldi**
LIPN , Université Sorbonne Paris Nord, CNRS UMR 7030
99 av. Jean-Baptiste Clément, 93430 Villetaneuse, France
`{liyanage,davide.buscaldi}@lipn.univ-paris13.fr`

Abstract

Thanks to the state-of-the-art Large Language Models (LLMs), language generation has reached outstanding levels. These models are capable of generating high quality content, thus making it a challenging task to detect generated text from human-written content. Despite the advantages provided by Natural Language Generation, the inability to distinguish automatically generated text can raise ethical concerns in terms of authenticity. Consequently, it is important to design and develop methodologies to detect artificial content. In our work, we present some classification models constructed by ensembling transformer models such as SciBERT, DeBERTa and XLNet, with Convolutional Neural Networks (CNNs). Our experiments demonstrate that the considered ensemble architectures surpass the performance of the individual transformer models for classification. Furthermore, the proposed SciBERT-CNN ensemble model produced an F1-score of 98.36% on the ALTA shared task 2023 data.

1 Introduction

Nowadays, people have access to state-of-the-art LLMs which help them simplify some of their daily activities. One of the most notable breakthroughs in recent years is the evolution of OpenAI's GPT models which are capable of generating text that looks as if they are written by a human. Especially, the latest models such as ChatGPT and GPT4 (OpenAI, 2023) have won global attention for providing solutions to any kind of question or concern that humans possess. Moreover, these models produce outputs that appear to be written by a human.

Thus there is a potential risk in determining the authenticity of textual content that mankind refers to. Especially, in a domain such as academia, leveraging generation models in composing articles might raise an ethical concern. For example in ICML 2023, they have included a note under the "Ethics" section prohibiting the use of text generated by ChatGPT and other LLMs, unless "presented as part of the paper's experiential analysis." [1]. Accordingly, it is essential to have mechanisms for detecting artificially composed text from human written text.

Currently, a substantial amount of research has focused on the detection of automatically generated text. Recent research ((Zellers et al., 2019), (Glazkova and Glazkov, 2022) and Liyanage and Buscaldi (2023)) mostly consider detection as a binary classification task and leverage SOTA classification models to distinguish machine-generated text from original text. Besides, some employ statistical detection tools such as GLTR (Gehrmann et al., 2019) or latest deep learning based tools such as GPT2 output detector[2], DetectGPT (Mitchell et al., 2023) or GPTZero [3]. Moreover, several researchers (Liyanage et al. (2022), (Kashnitsky et al., 2022)) have published corpora composed of machine-generated content, which can be utilized by future research on detection.

Our work is based on the participation of our team in the ALTA shared task 2023 (Molla et al., 2023) The objective of the task is to build automatic detection systems that can discriminate between human-authored and synthetic text generated by Large Language Models (LLMs). Their corpus is composed of artificial contents that belong to a variety of domains (such as law, medical) and are generated by models such as T5 (Raffel et al., 2020) and GPT-X.

This paper is organized as follows. We provide the corpus and task description in Section 2. In Section 3, we describe our methodology and Section 4, deliver the experimental setup and the official results. Section 5 concludes this paper.

---

[1] `https://icml.cc/Conferences/2023/llm-policy`

[2] `https://openai-openai-detector--5smxg.hf.space`

[3] `https://gptzero.me/`

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 107–111
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

2 Task Overview

2.1 Task Definition

The task at hand revolves around distinguishing between automatically generated and human-written texts. In essence, it involves a binary classification challenge where the goal is to categorize provided texts into two distinct and exclusive groups. To outline this formally:

- Input: We are presented with text segments.
- Output: The objective is to assign one of two possible labels to each text segment: either "human-written" or "machine-generated".

This undertaking aims to establish a clear boundary between texts created through automated processes and those crafted by human authors. The primary aim is to develop a model that can effectively differentiate between these two categories based on the characteristics of the given excerpts.

2.2 Corpus

The dataset published for the ALTA shared task is a balanced one composed of 9000 original (human written) excerpts and 9000 fake (artificially generated) excerpts. On average, the excerpts consist of 35 words each. To gain a deeper comprehension of the corpus, category-wise (original vs generated) statistics with respective example excerpts are provided in Table 1.

3 Methodology

Given that the shared task frames detection as a binary classification challenge, we utilized a range of classification models to address this objective. In the subsequent subsections, in-depth explanations are provided pertaining to the examined statistical, recurrent and transformer models, and the corresponding ensemble architectures.

3.1 Statistical Models and their Respective Ensemble Architectures

In our work, we primarily employed Naive Bayes, Passive Aggressive and Support Vector Machine (SVM), which are classification algorithms used in machine learning to categorize data points into different classes (Bishop and Nasrabadi, 2006). Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem and it is widely used for tasks such as spam detection. It assumes that the features are conditionally independent given the class label. Passive Aggressive is a type of algorithm that aims to make aggressive updates when it encounters a misclassified point and passive updates when the point is correctly classified. SVM is a powerful supervised machine learning algorithm used for classification and regression tasks. It is a popular algorithm in text classification tasks. These algorithms were employed in conjunction with the two text encoding methodologies, namely Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF).

Furthermore, we harnessed the capabilities of ensembles comprising the aforementioned statistical models, applying various ensemble methodologies such as voting, stacking, bagging, and boosting. By amalgamating the predictions of multiple models, ensemble techniques aim to enhance the overall predictive power of our system. Voting combines the outputs through a majority or weighted decision, stacking involves training a meta-model on the predictions of base models, bagging leverages bootstrapped subsets of data for training individual models, and boosting iteratively adjusts model weights to prioritize difficult-to-classify instances. Through these ensemble strategies, we sought to extract richer insights from our data and attain improved classification performance.

3.2 Recurrent Models and their Respective Ensemble Architectures

Recurrent models, a subset of neural network architectures, are models designed to capture temporal dependencies and patterns within sequences. We conducted experiments with LSTM and Bi-LSTM models, which are a type of RNN architecture specifically designed to address the vanishing gradient problem that can occur in traditional RNNs. To further improve classification accuracies of these models, we ensembled them with a Convolutional Neural Networks (CNNs) architecture. The proposed hybrid RNN-CNN approach helps in enhancing the predictive capabilities overall model by capitalizing on their respective strengths in capturing temporal dependencies and spatial features. We trained the entire ensemble end-to-end, allowing the network to learn how to best combine the features extracted by both LSTM and CNN components.

	Original	**Generated**
Min. word count	10	1
Max. word count	96	192
Avg. word count	25	45
Example excerpt	This is the data I collected so far (motorcycle standing on central stand, back wheel revolving, velocity comes from the back wheel, ABS LED blinking).	In this sense, she emphasized that it was a mistake to tie development aid to times of economic booms, as it is a "permanent commitment".

Table 1: Statistics of the ALTA shared task corpus (The avg. figures are rounded off to the nearest whole number)

3.3 Transformer Models and their Respective Ensemble Architectures

For our classification experiments, we leveraged cutting-edge transformer models, namely BERT, SciBERT, DeBERTa, and XLNet. These state-of-the-art architectures have demonstrated exceptional proficiency in a wide spectrum of natural language processing tasks, including classification. BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2018) introduces bidirectional context by pretraining on a massive corpus and then fine-tuning on task-specific data. SciBERT (Beltagy et al., 2019) is specialized for scientific text, adapting BERT's embeddings to domain-specific language. DeBERTa (Decoding-enhanced BERT with Disentangled Attention) (He et al., 2020) enhances attention mechanisms, capturing dependencies among words more effectively. XLNet (Yang et al., 2019) employs a permutation-based training approach to capture bidirectional context and alleviate BERT's limitations.

Initially, we created ensembles by combining the capabilities of SciBERT and DeBERTa models with the foundational BERT model. This process involves channeling the data through each base model, which comprises the transformer block along with a subsequent max pooling layer. Subsequently, the outcomes derived from these individual models are concatenated to generate a unified representation, which is then channeled into a linear classification layer for making refined predictions.

Furthermore, we combined the transformer model with Convolutional Neural Networks (CNNs) to build ensemble architectures that exhibit enhanced performance. As depicted in the architectural diagram 1, the embeddings produced by the transformer model are used as input for a CNN. This network includes three stacked convolutional layers to cover a large enough part of the input. The output of the three stacked layers is then passed through a dropout, a max pooling and another dropout layer before being passed to a dense layer for the classification. In our approach, we don't need to embed the output using nn.Embedding layers, as there is no need for a lookup table.

4 Experiments and Results

The text underwent preliminary processing, involving the elimination of stopwords and stemming, before being supplied to either statistical or neural network architectures. The processed data was then transformed into numerical vectors using Bag of Words (BoW) or tf-idf encoding techniques, which were subsequently utilized as inputs for the statistical models. All of the employed statistical models, as well as their corresponding ensemble methods, were imported from the Scikit-learn library. For constructing LSTM and CNN models, the relevant layers were imported from TensorFlow's Keras module. Training these recurrent models, including those combined with CNN ensembles, involved running 10 epochs. The LSTM and Bi-LSTM architectures were trained using batch sizes of 64 and 128, respectively.

Concerning transformer architectures and their associated ensembles, pre-trained models from Hugging Face (Wolf et al., 2020) were imported and subsequently fine-tuned through the utilization of Simple Transformers [4]. The BERT tokenizer was consistently employed across all models. The fine-tuning process involved 3 epochs, a batch size of 16, and a maximum sequence length of 128. Leveraging the T4 GPU Hardware accelerator, the average training time for models was approximately 30 minutes. For standalone models, the input consisted of unprocessed text, while ensembles underwent pre-processing involving punctuation removal and conversion to lowercase. As represented

---

[4] `https://simpletransformers.ai`

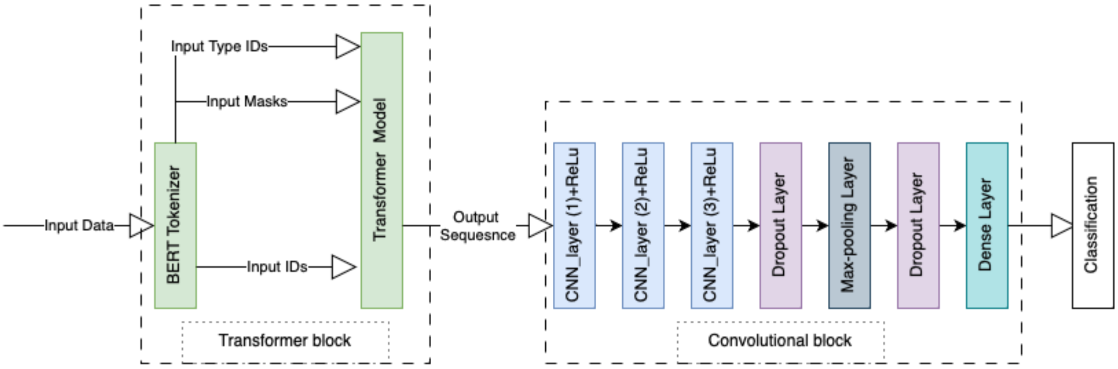


Figure 1: Architecture of Transformer-CNN Ensemble (Here, the "input type ids," "input masks," and "input ids" are the components used to prepare and encode the input data for the transformer model.)

in Figure 1, the CNN block of the ensembles was composed of three convolutional layers.

The dataset was split in 80:20 ratio for training and testing. To assess the classification performance of the models under consideration, the F1 score was employed. This score, being a balanced combination of precision and recall, offers a comprehensive evaluation. Each model underwent a total of five experimental iterations, and the resultant average F1 scores are presented in Table 2.

In general, the ensemble architectures have exhibited superior performance compared to their corresponding original models. Our best-performing solution is the combination of DeBERTa$_{large}$ with CNN, achieving an F1 score of **98.36%**.

Considering that baseline models such as Naïve Bayes and tf.idf weighting obtain scores close to 90%, it is clear that the dataset is not well balanced. In fact, looking at the Multinomial Naïve Bayes and the log probabilities differences for all features, we observed a thematic bias. Specifically, the top most probable words in the negative category (human-generated) are law-oriented: "plaintiff", "defendant", and "judgment". On the other hand, LLM-generated text contains words like "round", "league", "players", etc. Therefore, it is not clear whether these results are generalizable to the general task of detecting artificial text.

Model	**F1**
Statistical Models	
NB + BoW	89.04
PA + BoW	84.07
SVM + BoW	87.51
NB + tf-idf	89.02
NB + tf-idf	91.00
NB + tf-idf	91.42
Ensembles of Statistical Models	
Voting (NB + PA + SVM) + BoW	90.29
Stacking (NB + PA + SVM) + BoW	88.23
Bagging (NB + PA + SVM) + BoW	91.56
Boosting (NB + PA + SVM) + BoW	90.28
Recurrent Models	
LSTM	49.08
Bi-LSTM	90.58
Ensembles of RNNs	
LSTM + CNN	49.08
Bi-LSTM + CNN	90.02
Transformer Models	
BERT$_{base}$	90.81
SciBERT	94.89
DeBERTa$_{large}$	96.67
XLNet$_{large}$	93.62
Ensembles of BERT models	
BERT$_{base}$ + SciBERT	97.80
BERT$_{base}$ + DeBERTa$_{large}$	97.47
Ensembles of transformers with CNN	
BERT$_{base}$ + CNN	97.42
SciBERT + CNN	97.56
DeBERTa$_{large}$ + CNN	**98.36**
XLNet$_{base}$ + CNN	97.44

Table 2: Classification Scores

5 Conclusion

In this work, we have explored the application of different SOTA classification models on the detection of automatically generated text from human written text. Moreover, we have created various ensemble methods with the aforementioned models and examined their performance on the detection task. Our results on the test data showed that generally the ensemble architectures outperform the considered original models. However, an analysis of the dataset raises some doubts about the generalizability of these results as it looks like the data are thematically biased. Therefore, these results should be considered only within the scope of the ALTA 2023 shared task.

As future work, we plan to examine the applicability of our ensemble architectures in detecting artificially generated text in multilingual corpora. Another potential research direction involves assessing the effectiveness of knowledge-based approaches for detecting artificial text.

References

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*.

Christopher M Bishop and Nasser M Nasrabadi. 2006. *Pattern recognition and machine learning*, volume 4. Springer.

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*.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander M Rush. 2019. Gltr: Statistical detection and visualization of generated text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 111–116.

Anna Glazkova and Maksim Glazkov. 2022. Detecting generated scientific papers using an ensemble of transformer models. In *Proceedings of the Third Workshop on Scholarly Document Processing*, pages 223–228.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Yury Kashnitsky, Drahomira Herrmannova, Anita de Waard, Georgios Tsatsaronis, Catriona Fennell, and Cyril Labbé. 2022. Overview of the dagpap22 shared task on detecting automatically generated scientific papers. In *Third Workshop on Scholarly Document Processing*.

Vijini Liyanage and Davide Buscaldi. 2023. Detecting artificially generated academic text: The importance of mimicking human utilization of large language models. In *International Conference on Applications of Natural Language to Information Systems*, pages 558–565.

Vijini Liyanage, Davide Buscaldi, and Adeline Nazarenko. 2022. A benchmark corpus for the detection of automatically generated text in academic publications. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4692–4700.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. *arXiv preprint arXiv:2301.11305*.

Diego Molla, Haolan Zhan, Xuanli He, and Qiongkai Xu. 2023. Overview of the 2023 alta shared task: Discriminate between human-written and machine-generated text. In *Proceedings of the 21st Annual Workshop of the Australasian Language Technology Association (ALTA 2023)*.

OpenAI. 2023. Gpt-4 technical report.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 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.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. *Advances in neural information processing systems*, 32.

Chat Disentanglement: Data for New Domains and Methods for More Accurate Annotation

Sai R. Gouravajhala†, **Andrew M. Vernier**†, **Yiming Shi**†, **Zihan Li**†
Mark S. Ackerman†, **Jonathan K. Kummerfeld**†‡*
†Computer Science & Engineering, University of Michigan, Ann Arbor
‡School of Computer Science, University of Sydney

Abstract

Conversation disentanglement is the task of taking a log of intertwined conversations from a shared channel and breaking the log into individual conversations. The standard datasets for disentanglement are in a single domain and were annotated by linguistics experts with careful training for the task. In this paper, we introduce the first multi-domain dataset and a study of annotation by people without linguistics expertise or extensive training. We experiment with several variations in interfaces, conducting user studies with domain experts and crowd workers. We also test a hypothesis from prior work that link-based annotation is more accurate, finding that it actually has comparable accuracy to set-based annotation. Our new dataset will support the development of more useful systems for this task, and our experimental findings suggest that users are capable of improving the usefulness of these systems by accurately annotating their own data.

1 Introduction

Rapid synchronous chat involving a large group often leads to overlapping conversations. The challenge of disentangling these conversations has been studied for over a decade, but the main datasets are expert annotated and based on discussion of Linux (Elsner and Charniak, 2008) and Ubuntu (Kummerfeld et al., 2019). Recent work has considered scripts from movies (Chang et al., 2023), but there is still the need for data from additional sources to measure the generalizability of methods.

A range of methods have been proposed to avoid expensive expert annotation in NLP, e.g., crowd work (Snow et al., 2008), games with a purpose (Jurgens and Navigli, 2014) and user feedback (Iyer et al., 2017). Various annotation methods have been used for disentanglement, but all focused on experts and only one study has compared annotation tools (Cerezo et al., 2021).

This work takes two key steps to expand this task to new domains: (1) we created a new, multi-domain, gold-standard dataset, and (2) we explored annotation methods to see if domain experts and crowd workers can do the task.

Our dataset includes several important variations not seen in existing datasets: (a) new types of conversations (e.g., meetings), (b) new types of user relationships (e.g., business-customer), and (c) a range of Internet Relay Chat (IRC) networks. We annotated 600 messages from each channel, which is enough to evaluate out-of-domain ccuracy.

It is impossible to collect expert labels for every domain. However, if we can develop the right tools, owners and users of channels may be able to improve models by annotating some of their own data. We conducted a user study with domain experts and crowd workers, exploring two types of variation in user interfaces: (1) whether annotators receive automatic guidance, and (2) what structure is annotated. Prior work has speculated that link annotation[1] is more accurate than set annotation[2] (Elsner and Charniak, 2010), but our work is the first controlled comparison.

We found that domain experts can effectively annotate data, and improve with automatic guidance. Crowd workers struggled with the task, doing worse than an automatic model. Set-based and link-based annotation are actually comparable in accuracy. We recommend link annotation as it provides the internal structure of conversations.

The dataset we release[3] will support the development of more generalizable models, and our findings show how to help domain experts annotate effectively. Together, these results will enable progress on this challenge in new domains, making conversations easier to follow for everyone online.

---

* jonathan.kummerfeld@sydney.edu.au

[1] Labeling reply-to relations between pairs of messages, then each connected graph of messages is a conversation.

[2] Putting messages into groups, where each group is a conversation.

[3] https://www.jkk.name/irc-disentanglement/

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 112–117
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

Channel	Network	Purpose	Msg / Hr	Users / Hr	Tok / Msg	κ
Mediawiki	Wikimedia	Technical support regarding mediawiki software.	71	4.1	10	0.78
Rust	Mozilla	Help related to the Rust programming language.	33	8.0	12	0.80
Stripe	Freenode	Customer support for the payments processing service.	76	6.6	16	0.81
Ubuntu Meeting	Ubuntu	Developer meetings.	371	9.9	8	0.71
Ubuntu	Ubuntu	Technical support for users of the operating system.	395	32	10	0.72

Table 1: Expert annotator agreement (κ) and properties of the four channels we annotated and the Ubuntu channel used in Kummerfeld et al. (2019). The channels span multiple topics (programming languages, customer support, web applications) and conversation styles (question-answer, meetings).

2 Related Work

All prior annotation for conversation disentanglement has been done by trained experts, like many tasks in NLP (Ide and Pustejovsky, 2017). Early work on the task asked annotators to form sets of messages (Elsner and Charniak, 2008, 2010), but they speculated that annotators may be more consistent at annotating reply-to links. Subsequent work took the link approach (Riou et al., 2015; Mehri and Carenini, 2017; Kummerfeld et al., 2019; Cerezo et al., 2021). This work is the first controlled comparison of the two. Cerezo et al. (2021) compared a command-line UI and GUI, finding that annotators preferred the GUI, but accuracy was the same, and using the GUI was slower. Our study complements theirs by considering: (1) variation in who annotators are, (2) variation in the form of annotation, and (3) guidance.

Crowd work can be cheaper and more scalable than expert annotation (Snow et al., 2008). Effective crowd annotation user interfaces and workflows have been developed for a range of tasks (e.g., Dumitrache et al., 2018; Finin et al., 2010; Larson et al., 2020), but there has been no prior work for disentanglement.

Guiding annotators using an automatic system has improved speed for other tasks (Marcus et al., 1993; Chiou et al., 2001). Recent work has applied similar ideas to crowd work (Gormley et al., 2010; Ramírez et al., 2019). We apply this idea to conversation disentanglement for the first time.

3 Data in New Domains

When multiple synchronous conversations are happening in the same channel they can be difficult to understand.[4] Conversation disentanglement is the task of identifying separate conversations, to make them understandable and useful.

Channel	F	1-1
Mediawiki	46	90
Rust	60	91
Stripe	83	94
Ubuntu Meeting	22	73
Ubuntu	43	82

Table 2: Model accuracy on conversations for each of the channels.

There are hundreds of active Internet Relay Chat (IRC)[5] channels, but only two have disentanglement annotations: #Ubuntu (Kummerfeld et al., 2019) and #Linux (Elsner and Charniak, 2008, 2010). To create a realistic out-of-domain setting, we annotated data from four diverse channels, described in Table 1. We chose channels that: (1) have public logs, (2) have various topics and conversation styles, and (3) are from different IRC networks, which may exhibit different conventions.

For each channel, we used three random samples, each 1,200 messages long (200 to annotate, 1,000 for context). This leads to a total of 2,400 annotated messages and a further 12,000 context messages. Our data is in the same format as Kummerfeld et al. (2019) to enable easy evaluation. This is the first work to annotate multi-domain data, enabling out-of-domain evaluation.

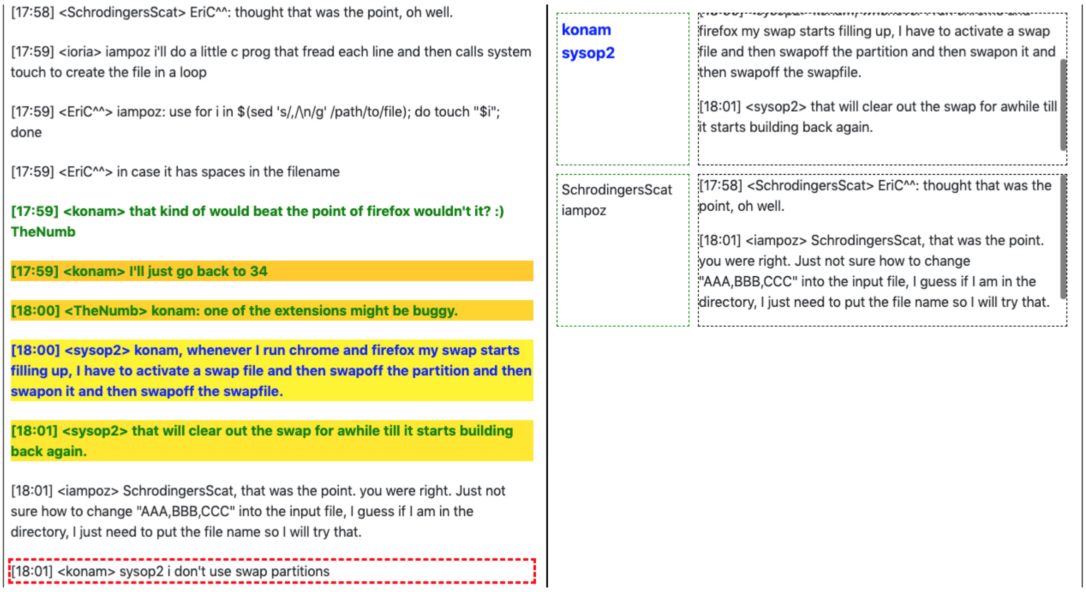
Expert Annotation To make a gold-standard reference, two of the authors labeled each file, then one of the authors adjudicated disagreements. To match the annotations of Kummerfeld et al. (2019) as closely as possible, we labeled reply-to links using their tool, SLATE (Kummerfeld, 2019), and the same annotation guidelines. Conversations are the connected components in the reply-to graph.

Table 1 shows agreement scores for reply-to

---

[4] Some services, e.g., Slack, WebEx, and Microsoft Teams, have the ability to split a conversation starting at a message, but that only solves the problem if the split is created as soon as a new topic is started and the conversation remains on topic.

[5] IRC is a protocol for synchronous chat in use since 1988.



[17:58] <SchrodingersScat> EriC^^: thought that was the point, oh well.

[17:59] <ioria> iampoz i'll do a little c prog that fread each line and then calls system touch to create the file in a loop

[17:59] <EriC^^> iampoz: use for i in $(sed 's/,/\n/g' /path/to/file); do touch "$i"; done

[17:59] <EriC^^> in case it has spaces in the filename

[17:59] <konam> that kind of would beat the point of firefox wouldn't it? :) TheNumb

[17:59] <konam> I'll just go back to 34

[18:00] <TheNumb> konam: one of the extensions might be buggy.

[18:00] <sysop2> konam, whenever I run chrome and firefox my swap starts filling up, I have to activate a swap file and then swapoff the partition and then swapon it and then swapoff the swapfile.

[18:01] <sysop2> that will clear out the swap for awhile till it starts building back again.

[18:01] <iampoz> SchrodingersScat, that was the point. you were right. Just not sure how to change "AAA,BBB,CCC" into the input file, I guess if I am in the directory, I just need to put the file name so I will try that.

[18:01] <konam> sysop2 i don't use swap partitions

konam
sysop2

firefox my swap starts filling up, I have to activate a swap file and then swapoff the partition and then swapon it and then swapoff the swapfile.

[18:01] <sysop2> that will clear out the swap for awhile till it starts building back again.

SchrodingersScat
iampoz

[17:58] <SchrodingersScat> EriC^^: thought that was the point, oh well.

[18:01] <iampoz> SchrodingersScat, that was the point. you were right. Just not sure how to change "AAA,BBB,CCC" into the input file, I guess if I am in the directory, I just need to put the file name so I will try that.

Figure 1: Part of the user interface for link annotation with guidance. The left side is the log of messages and the right side is the set of annotated conversations. The red box is the message to be annotated. Yellow / orange highlights are four of the predictions from the out-of-domain model. Blue and green text are explained in Section 4.1. The annotator needs to select the earlier message that the red message is replying to. The red message will then be added to the same conversation (on the right) as the message it is replying to. If the red message is the start of a new conversation the annotator will press a special button (not shown here).

links before adjudication. Agreement is as good or better than prior work. Based on our experience doing annotation, the Ubuntu Meeting channel was harder to annotate because the discussion was rapid and interleaved. The model struggles in this domain, with by far the lowest performance, as shown in Table 2.

4 Improving Annotation

To go beyond expert-annotated resources, we need effective annotation methods for either users (e.g., domain experts who run a channel and are willing to annotate data for their own use) or crowd workers (who can be recruited at larger scale). We perform the first experiments in annotation with both of these groups, exploring several variations in tool support for them.

4.1 Annotation Tools

Figure 1 shows a screenshot of part of our tool. We considered two forms of variation: (1) the type of annotation and (2) whether guidance is provided. In all cases, there was an interactive tutorial that explained the interface and annotation conventions.

Annotation Type Conversations can be annotated in two ways: forming sets of messages, where each set is a conversation (set-based); or creating a graph of reply-to links between messages, in which case each connected component in the graph is a conversation (link-based). This is the first systematic comparison of these two types of annotation.

Guidance We implemented guidance to help annotators. We used the feedforward neural network model from Kummerfeld et al. (2019), trained on their Ubuntu data, to predict reply-to links. For details of the model architecture, training, and in-domain accuracy, see Kummerfeld et al. (2019). Our data is out-of-domain for the model, and it is not perfect even in-domain, so we showed the top five predictions, with darker shades of yellow indicating more likely options. On this data, the top five predictions have an average recall of 92%. In the link annotation case, we highlighted individual messages, as shown in Figure 1. In the set annotation case, we highlighted the conversations those messages belong to.[6]

We also changed the colour of messages to indicate likely interactions: (a) messages written by the current user and any message that addresses[7] the current user were green, (b) if the current message addresses someone, then we made messages from

---

[6] If multiple predicted messages were in the same conversation then the shade of yellow is based on the max probability of the options.

[7] This is when one user mentions another user in a message.

that user green as well, and (c) messages where both (a) and (b) were true were blue. Figure 1 shows examples of these variations.

4.2 Participants

Participants were randomly split into the four task conditions. They completed an interactive tutorial, then annotated a 34 message sample from each channel. Following Kummerfeld et al. (2019), we provided 1,000 prior messages as context. To mitigate learning and task fatigue effects, we varied the order of the channels across participants.[8]

Domain Experts We recruited seventeen fluent English speakers who were PhD students in Computer Science at the University of Michigan, but not doing research in NLP. They have knowledge of the subject area, but no prior experience with disentanglement. Each participant received an Amazon gift card valued at $25 for assisting in the study. We have excluded one participant, who misunderstood the task, performed extremely poorly, and expressed confusion.

Crowd We recruited 128 workers via Amazon Mechanical Turk, requiring that workers had a 98% HIT approval level and be U.S.-based. Each HIT was worth $3.75, an effective rate of $15 per hour when counting time spent reading instructions and doing the tutorial as well as the task.

4.3 Metrics

We considered three measures of agreement between our participants and the experts: κ, the standard metric applied to reply-to links; *Conv-F*, an F-Score calculated based on how many conversations match exactly; and *1-1*, a conversation-matching metric from Elsner and Charniak (2008). We also measured the time taken. Note that κ can only be calculated for cases where the type of annotation is reply-to links (Kummerfeld et al., 2019).

We also include the accuracy of the model that provided guidance. This provides a baseline that annotators must exceed for their work to be helpful.

We do significance testing with one-tailed unpaired t-tests. To control for family-wise errors, we apply the Holm-Bonferroni Method (Holm, 1979). Results of tests are described where relevant in the text below.

Anno. Type	Guidance	Accuracy			Time (min)
		κ	Conv-F	1-1	
Computer Science PhD Students					
Conv	No	-	51	80	6
Conv	Yes	-	58	87	7
Link	No	0.68	43	80	6
Link	Yes	**0.79**	**69**	**92**	10
Crowd workers					
Conv	No	-	33	74	5
Conv	Yes	-	39	70	6
Link	No	0.52	19	64	8
Link	Yes	0.55	37	69	9
Automatic		0.68	53	78	-

Table 3: Accuracy and time for each condition. Metrics are defined in Section 4.3. Domain experts provide high quality annotations, particularly with guidance.

4.4 Ethics

The use of public IRC logs was approved by the University of Michigan's IRB, as was the annotation study with human participants (Study IDs HUM00176661 and HUM00172084). To protect the identities of crowd workers, their Amazon IDs will not be released. Details of compensation are provided above, with values chosen to ensure fair payment without being so high as to be coercive. Our results are limited by the range of participants we had in the task and so may not be representative of all domains. This work does not introduce any significant new risks that we are aware of.

5 Results

Table 3 shows results for each of the conditions, which allow us to answer several questions.

Domain experts can annotate accurately. Comparing the top half of the table to the automatic results (bottom row), our participants provide annotations that are more accurate than the model, but only when given guidance (this difference is statistically significant).

Guidance helps domain experts. The conditions with guidance have higher accuracy (significant at the 0.05 level), though at the cost of more time (also significant). This is the reverse of the pattern seen in annotations for tasks such as POS tagging and NER, where guidance improves speed of annotation while keeping accuracy the same. One possible explanation is that the guidance is prompting annotators to read additional options, which helps them find an option they may have otherwise missed, but also leads them to read more,

---

[8] The orders were: SRUM, RMSU, USMR, MURS (S = Stripe, R = Rust, U = Ubuntu Meeting, and M = Mediawiki).

which takes time. In contrast, guidance in classification tasks such as POS tagging and NER does not reveal additional options (there is a fixed, known tag set) and does not lead to more reading.

Further work is needed to support crowd workers. Crowd workers are worse than the out-of-domain model in every condition. This indicates that further research is needed to help crowd workers succeed. It also shows that the needs of crowd workers and domain experts are different, as the domain experts were effective and improved with guidance, while crowd workers did not (the variations are not statistically significant). However, a few workers did have high accuracy. In a survey, we found that some of our workers had substantial technical knowledge, for example "My Unix experience goes back to SVR4 days (mostly IRIX & Solaris - ugh), and I still code on Linux occasionally". This suggests that domain experts exist in the crowd workforce and if they can be identified, e.g., by pre-screening, they may be as accurate as the students in our study.

Link-based and set-based annotation are comparable in accuracy. When comparing conditions that are equivalent except for the type of annotation, there is no statistically significant difference. This result answers the question from Elsner and Charniak (2010). We advise future work to annotate reply-to links as it provides additional information about the internal structure of conversations.

How should future work annotate disentanglement? Use domain experts, provide them with guidance, and ask them to annotate links. This led to our best results and provides internal structure.

6 Limitations

There are three main limitations of this work. First, the study participants are an approximation of domain experts, rather than being actual users of the IRC channels we consider. We believe Computer Science students are a reasonable proxy, given their knowledge of the subjects discussed in these channels, but it is possible that they are unaware of community-specific conventions or jargon.

Second, we only considered online communities writing in English. It is possible that communities writing in other languages use significantly different conventions that make this task easier or harder.

Third, our sample size is only large enough to make strong claims about some of the variations in results. It's possible that other variations in Table 3 would also be significant if we had a larger set of participants.

7 Conclusion

This work makes two key contributions. First, the new dataset we are releasing expands the scope of multi-domain evaluation of conversation disentanglement models. Second, our user study of variations in annotation tools shows that domain experts can effectively annotate, particularly when given automatic guidance. Together, these contributions show how better models and systems can be created that give domain-expert users the ability to improve systems. That will enable the use of this technology in a wide variety of new domains.

Acknowledgements

This material is based in part on work supported by DARPA (grant #D19AP00079), and the ARC (DECRA grant).

References

Jhonny Cerezo, Felipe Bravo-Marquez, and Alexandre Henri Bergel. 2021. Tools impact on the quality of annotations for chat untangling. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop*, pages 215–220.

Kent Chang, Danica Chen, and David Bamman. 2023. Dramatic conversation disentanglement. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4020–4046, Toronto, Canada.

Fu-Dong Chiou, David Chiang, and Martha Palmer. 2001. Facilitating treebank annotation using a statistical parser. In *Proceedings of the First International Conference on Human Language Technology Research*.

Anca Dumitrache, Lora Aroyo, and Chris Welty. 2018. Capturing ambiguity in crowdsourcing frame disambiguation. In *Proceedings of The Sixth AAAI Conference on Human Computation and Crowdsourcing*, pages 12–20.

Micha Elsner and Eugene Charniak. 2008. You talking to me? a corpus and algorithm for conversation disentanglement. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 834–842.

Micha Elsner and Eugene Charniak. 2010. Disentangling chat. *Computational Linguistics*, 36(3):389–409.

Tim Finin, William Murnane, Anand Karandikar, Nicholas Keller, Justin Martineau, and Mark Dredze. 2010. Annotating named entities in Twitter data with crowdsourcing. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 80–88.

Matthew R. Gormley, Adam Gerber, Mary Harper, and Mark Dredze. 2010. Non-expert correction of automatically generated relation annotations. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 204–207.

Sture Holm. 1979. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70.

Nancy Ide and James Pustejovsky. 2017. *Handbook of Linguistic Annotation*. Springer Netherlands.

Srinivasan Iyer, Ioannis Konstas, Alvin Cheung, Jayant Krishnamurthy, and Luke Zettlemoyer. 2017. Learning a neural semantic parser from user feedback. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 963–973.

David Jurgens and Roberto Navigli. 2014. It's all fun and games until someone annotates: Video games with a purpose for linguistic annotation. *Transactions of the Association for Computational Linguistics*, 2:449–464.

Jonathan K. Kummerfeld. 2019. Slate: A super-lightweight annotation tool for experts. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 7–12.

Jonathan K. Kummerfeld, Sai R. Gouravajhala, Joseph J. Peper, Vignesh Athreya, Chulaka Gunasekara, Jatin Ganhotra, Siva Sankalp Patel, Lazaros Polymenakos, and Walter S. Lasecki. 2019. A large-scale corpus for conversation disentanglement. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3846–3856.

Stefan Larson, Anthony Zheng, Anish Mahendran, Rishi Tekriwal, Adrian Cheung, Eric Guldan, Kevin Leach, and Jonathan K. Kummerfeld. 2020. Iterative feature mining for constraint-based data collection to increase data diversity and model robustness. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8097–8106.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Shikib Mehri and Giuseppe Carenini. 2017. Chat disentanglement: Identifying semantic reply relationships with random forests and recurrent neural networks. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 615–623.

Jorge Ramírez, Marcos Baez, Fabio Casati, and Boualem Benatallah. 2019. Understanding the impact of text highlighting in crowdsourcing tasks. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, pages 144–152.

Matthieu Riou, Soufian Salim, and Nicolas Hernandez. 2015. Using discursive information to disentangle French language chat. In *2nd Workshop on Natural Language Processing for Computer-Mediated Communication (NLP4CMC 2015) / Social Media at GSCL Conference 2015*, pages 23–27.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Ng. 2008. Cheap and fast – but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 254–263.

Enhancing Bacterial Infection Prediction in Critically Ill Patients by Integrating Clinical Text

Jinghui Liu **Anthony Nguyen**
The Australian e-Health Research Centre
CSIRO
`{jinghui.liu,anthony.nguyen}@csiro.au`

Abstract

Bacterial infection (BI) is an important clinical condition and is related to many diseases that are difficult to treat. Early prediction of BI can lead to better treatment and appropriate use of antimicrobial medications. In this paper, we study a variety of NLP models to predict BI for critically ill patients and compare them with a strong baseline based on clinical measurements. We find that choosing the proper text-based model to combine with measurements can lead to substantial improvements. Our results show the value of clinical text in predicting and managing BI. We also find that the NLP model developed using patients with BI can be transferred to the more general patient cohort for patient risk prediction.

1 Introduction

Data-driven AI models for healthcare have much potential to facilitate clinical care, promote healthcare efficiency, and support medical research (Topol, 2019; Rajpurkar et al., 2022). An important domain of medicine that could benefit from AI is infectious disease, where AI can help better understand infections so that we can design more effective approaches to monitor, diagnose, and treat infections (Wong et al., 2023). Among the different types of infections, bacterial infection (BI) is one of the most common and is estimated to be associated with more than 13 million deaths in 2019 alone (Collaborators, 2022).

Previous works have studied various types of AI models to predict the occurrence of BI-related diseases using data from Electronic Health Records (EHR), especially sepsis (Moor et al., 2021). Meanwhile, the prediction of BI in general is less studied, whereby structured measurements were used predominantly to develop predictive models (Yang et al., 2023; Eickelberg et al., 2023). The value of clinical text in BI prediction remains unclear.

In this study, we explore the usefulness of NLP for infection-related prediction task by focusing on BI prediction in critically ill patients. We follow an existing study (Eickelberg et al., 2020) on BI prediction that relies on a range of clinical measurements as features, and we compare it with common NLP models that rely solely on routinely collected clinical text (illustrated in Figure 1). We then use the best performing text encoder to develop multimodal fusion models for BI prediction, which obtains the state-of-the-art result. Finally, we study the applicability of NLP models for mortality prediction in different patient cohorts, showing that the model trained using patients with BI is more robust to data shift.

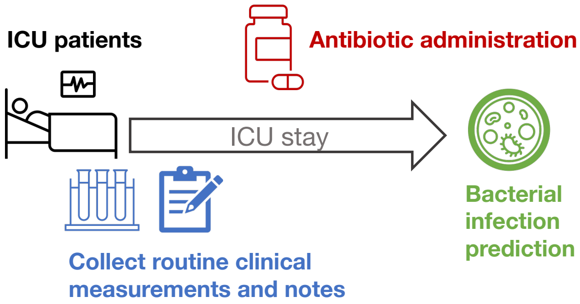

ICU patients
Antibiotic administration
ICU stay
Bacterial infection prediction
Collect routine clinical measurements and notes

Figure 1: Clinical text is integrated with clinical measurements to enhance the early prediction of bacterial infection, potentially helping inform clinical decisions regarding shortening the duration of unnecessary antibiotics to reduce risk of adverse patient outcomes and antimicrobial resistance.

2 Related Work

Many studies have developed machine learning models to predict diseases caused by bacterial infections, with urinary tract infection (Taylor et al., 2018; Dhanda et al., 2023) and sepsis (Liu et al., 2019; Moor et al., 2019) being the two most prominent examples. Early identification of these diseases is helpful, and sometimes essential, for clinicians to arrange lifesaving treatments. These studies typically use clinical measurements as features

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 118–124
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

for model development and may sometimes derive features from text as a supplement (Goh et al., 2021; Yan et al., 2022). Previous work studying BI prediction used clinical measurements (Eickelberg et al., 2020), and this was recently extended in a multicenter study (Eickelberg et al., 2023). Although text has been applied to predict specific diseases or organisms (Zhang et al., 2020), the contribution of text to BI prediction in general remains understudied.

Many previous work shows NLP models are effective for various clinical predictive tasks (Seinen et al., 2022; Liu et al., 2022a). Typical early prediction targets include patient mortality, length of stay in the hospital, readmission, diagnosis groups, or specific diseases. Multimodal fusion of different modalities in the EHR also shows promise in improving classification performance, such as combining clinical notes [1] and measurements (Deznabi et al., 2021; Soenksen et al., 2022). While previous works tend to focus on a specific type of text encoder or fusion mechanism to compare with unimodal modeling, the impact of varying these configurations on performance is not well understood.

The transferability of AI or ML models for clinical care is an important topic since many factors in healthcare can cause data shift (Finlayson et al., 2021). Applying models across different patient cohorts is also important in low-resource patient groups and to ensure fairness (Amir et al., 2021; Han et al., 2021). For example, a recent study shows that model trained in adult patients can be successfully transferred to pediatric patients (Lemmon et al., 2023). More studies are needed to understand the generalisability of models in healthcare.

3 Methods and Experiments

3.1 Task and cohort extraction

We follow Eickelberg et al. (2020) to extract a cohort of adult patients from the MIMIC-III ICU database (Johnson et al., 2016) suspected of having BI in the early phase of ICU admission. Suspicion is defined as 1) receiving at least one antibiotic within 96h after admission to the ICU and 2) having a microbiology culture tested within 24h before or after antibiotic use. For antibiotics, a duration over 96h is considered prolonged antibiotic use. For microbiology cultures, a positive culture means that a bacterial organism is detected [2]; thus, infection occurs. Unlike works focusing on specific bacteria, such as *E. coli*, we consider all possible bacteria identified from microscopy. Then, the binary classification task of BI considers prolonged antibiotic use and positive microbiologic culture as *positive* and short use and negative culture as *negative*. We follow the open source implementation to construct and process the cohort [3].

For input, we extract clinical measurements and clinical notes for patients suspected of BI. We follow Wang et al. (2020) to extract clinical measurements within the 24h data collection window from the first antibiotic dose after ICU admission. These measurements include routinely collected vital signs (such as heart rate and blood pressure) and laboratory results (such as white blood cell counts). We refer the readers to Wang et al. (2020) for a complete list of 104 clinical measurements. We did not experiment with longer windows as in (Eickelberg et al., 2020) for the purpose of this study. For clinical notes, to consider context before ICU admission, we collect all notes written before the 24th hour of ICU admission, such as those written when the patient was admitted to the hospital but not yet transferred to the ICU. We remove patients who do not have any notes recorded from the cohort. We then follow Eickelberg et al. (2020) to create train/validation/test sets with 70/10/20 ratio, where we ensure that a patient with multiple admissions appears only in one set. The statistics of the datasets are presented in Table 1.

	Train	Validation	Test
Num of cases	5937	984	2972
BI rate	19.6%	20.7%	19.6%
Mortality rate	11.7%	12.5%	10.3%
Avg num of notes	13.4	13.6	14.4
Avg num of words	4164.7	4114.2	4596.1

Table 1: Statistics of the BI cohort.

3.2 Data representation and modeling

3.2.1 Modeling clinical measurements

The structured clinical measurements are preprocessed and formatted as time series following the existing benchmark (Wang et al., 2020). We

---

[1] We use the terms of *clinical note* and *clinical text* interchangeably in this paper.

[2] Common contaminations are controlled by counting certain bacteria twice, i.e., *Staphylococcus*.

[3] `https://github.com/geickelb/mimiciii-antibiotics-opensource`

Model	AUC-ROC		AUC-PRC	
Measurement-based model	**0.772 (0.0029)**		**0.505 (0.0029)**	
Text-based models	Default ordering	Reverse ordering	Default ordering	Reverse ordering
TextCNN	0.706 (0.0041)	0.759 (0.0054)	0.346 (0.0062)	0.434 (0.0088)
BiLSTM	0.585 (0.0056)	0.646 (0.0108)	0.245 (0.0023)	0.289 (0.0093)
BERT	0.635 (0.0118)	0.717 (0.0074)	0.275 (0.0082)	0.399 (0.0145)
BERT+LSTM	0.703 (0.0099)	0.715 (0.0041)	0.337 (0.0112)	0.391 (0.0049)
Longformer	0.629 (0.0057)	0.743 (0.0026)	0.281 (0.0016)	0.437 (0.0032)

Table 2: Results of the measurement-based model and different NLP models for BI prediction. The best scores are bolded, and the second best are underscored. All scores are averaged over five runs with different random seeds.

use GRU-D in our study (Che et al., 2018), which is a strong baseline for classifying physiological time series (Rubanova et al., 2019).

3.2.2 Modeling clinical text

We consider a variety of NLP models to process clinical notes for the BI prediction task.

TextCNN: We follow the standard implementation of the classic text CNN model with multiple filters (Kim, 2014). Pretrained, in-domain word embeddings are used (Zhang et al., 2019). All notes are concatenated as a single text string as input.

BiLSTM: Previous work shows that bidirectional LSTM can be a competitive baseline even compared with more complex models for text classification (Adhikari et al., 2019). The input text is processed as for TextCNN.

BERT: We fine-tune BERT (Devlin et al., 2019) for BI classification. As pretrained BERT has an input cap of 512 tokens, the notes are concatenated and then truncated to fit this size. We use the in-domain ClinicalBERT (Alsentzer et al., 2019).

BERT+LSTM: BERT is used to encode each clinical note (first 512 tokens) and form a time series for modeling with another encoder (Zhang et al., 2020; Liu et al., 2023). We adopt this hierarchical strategy by encoding notes with ClinicalBERT to get [CLS] token representations to then model with an LSTM.

Longformer: To expand the capacity of pretrained language models, we fine-tune Longformer (Beltagy et al., 2020) with an input size of 2048 tokens. We also initialize it with in-domain pretrained weights (Li et al., 2023).

We tested two methods of ordering clinical notes. The first is the default ordering following temporal order. The other is to reverse the temporal ordering so that the most recent note appears first. Having the most updated notes appear first can be important for models with limited context length.

3.2.3 Multimodal fusion

Clinical measurements and text are combined to see if BI prediction performances can be improved. The measurements are again encoded by GRU-D. We follow previous work (Liu et al., 2023) to adopt BERT+LSTM as text encoder and then fuse with GRU-D using late fusion (Huang et al., 2020) or the attention-based fusion mechanism (Liu et al., 2023). Finally, to obtain the best result and explore whether text encoder selection matters, we select the best NLP model from the models we examined and combine it with measurement using late fusion.

3.3 Experiments

We use the area under the receiver operating curve and the precision recall curve (AUC-ROC and AUC-PRC) as metrics to evaluate the performance. We perform early stopping based on AUC-ROC (main metric) in the validation set if the score plateaus for more than five epochs for CNN and LSTM models. We tune hyperparameters for all models with grid search (see search space in the Appendix A). After finding the best configuration, the model is trained using five random seeds, whose results in the test set are averaged and presented as mean and standard deviation.

4 Results and Discussion

4.1 Modeling clinical measurement is overall better than text for BI prediction

We present the modeling results using a single input modality in Table 2. The first observation is that our implementation of GRU-D using measurements from the 24h data collection window achieves a similar performance in Eickelberg et al. (2020), where their AUC-ROC results with different classifiers range from 0.763 to 0.776, indicating that our experimental setup is consistent with previous work. We then find that the measurement-based model performs better than all the NLP models examined. This trend is similar to other clinical prediction tasks, such as mortality prediction, where structured data can outperform text (Hsu et al., 2020). This is likely because measurements can capture detailed and quantitative fast-changing physiology in patients, not consistently found in clinical notes. (Gong and Guttag, 2018).

4.2 Choice of NLP models is important for BI prediction

Nevertheless, we find text-based models can achieve competitive performances for BI prediction, especially when we reverse the order of notes. TextCNN and Longformer obtain the second best results with reversed note ordering for AUC-ROC and AUC-PRC, respectively, and approach the best results from the measurement-based model. Reverse ordering (i.e., using the lastest portions of clinical notes) brings significant benefits for models with limited context length (i.e., BERT and Longformer), which means having more sophisticated methods to select specific portions of clinical notes (Zheng et al., 2023) or remove text redundancies (Liu et al., 2022b) can potentially bring further performance boosts for BI prediction – an avenue for furture investigations.

In addition, we also observe the significant disparity between different NLP models. For example, BiLSTM obtains unexpectedly poor results compared to other methods. This may indicate that RNN is not suitable for clinical text (Boag et al., 2018) as term-level triggers may be sufficient, which can be better identified by CNN. Our results indeed show that TextCNN performs well under all settings, except when compared with Longformer under AUC-PRC. The pretrained transformer models overall underperform the simpler CNN model despite having adapted to the clinical domain and prolonged input context (i.e., Longformer). We suspect that this is because the vocabularies used by ClinicalBERT and ClinicalLongformer are not domain-specific (Koto et al., 2021) and do not handle the noise in the clinical text well. In addition, we follow Li et al. (2023) to decide the hyperparameter space when fine-tuning Longformer. It is possible that Longformer can achieve better results with more computation resources and further hyperparameter tuning. In this study, we have choosen TextCNN to balance performance and efficiency for BI prediction, and used it in combination with clinical measurements for multimodal fusion.

Model	AUC-ROC	AUC-PRC
Measurement-based model	0.772 (0.0029)	0.505 (0.0029)
Fusion with note representations encoded by BERT		
Late fusion	0.774 (0.0019)	0.508 (0.0049)
Attention-based fusion	0.781 (0.0045)*	0.508 (0.0077)
Fusion with the best text-based model		
Late Fusion	**0.799 (0.0047)***	**0.541 (0.0052)***

Table 3: BI prediction results using both measurement and text. Scores with * denote statistically significant improvement compared to measurement-based model (p-value < 0.01).

4.3 Fusion with proper NLP model improves BI prediction

Table 3 presents the results of combining measurement and text for the prediction of BI. We follow previous works to use BERT+LSTM as text encoder (Liu et al., 2023), but it provided limited benefit even with more complicated attention-based fusion mechanisms. It shows that BI prediction is different from common clinical prediction tasks in utilizing information from the two modalities. Also, the text-based BERT+LSTM alone achieves suboptimal results, which is likely the factor that limits its fusion performance. We thus select the best text encoder from Table 2 (TextCNN with reverse note ordering) and combine with measurement-based model using late fusion, which obtains significantly improved performances (p-value < 0.01, T-test). This shows that finding a proper NLP encoder for multimodal fusion can bring considerable boost to the early prediction of BI.

4.4 BI cohort is robust to training NLP models for risk prediction

Finally, we use the BI cohort to train an NLP model to predict in-hospital mortality and compare with another model trained using a general cohort of ICU patients, who may or may not have bacterial infection. The size of the GENERAL cohort is about 4.5 times that of the BI cohort (more details in Appendix B). Patients in each of the train, validation and test sets of the BI cohort appear in the corresponding set of the GENERAL cohort. We again use TextCNN with reverse ordering for model training and evaluation.

Table 4 shows the results of the mortality pre-

Model	AUC-ROC	AUC-PRC
Model trained using BI cohort		
BI test set	0.814 (0.0085)	0.377 (0.0121)
GENERAL test set	0.809 (0.0044)	0.368 (0.0106)
Model trained using GENERAL cohort		
GENERAL test set	0.893 (0.0016)	0.592 (0.0031)
BI test set	0.757 (0.0134)	0.481 (0.0241)

Table 4: The mortality prediction results on two cohorts.

diction in the two cohorts. Models trained on the BI cohort and the GENERAL cohort achieve the AUC-ROC of 0.814 and 0.893 in their corresponding in-distribution test sets. The model trained on GENERAL appears to be more capable given that it has seen more samples. We then apply these models to the test sets from the different cohorts.

Now we see that the model trained on GENERAL performs significantly worse on the BI test set (0.893 to 0.757), while the BI model maintains its performance (0.814 to 0.809). This has two implications. First, it shows that a risk prediction model trained using a general population cannot be directly applied to patients with bacterial infection (AUC-ROC drops from 0.814 to 0.757) and a dedicated model needs to be trained. This relates to the effect of data bias on subpopulations that causes models to learn shortcuts and perform differently across various groups of patients (Brown et al., 2023). Second, patients with bacterial infection turn out to be a valuable resource for training a robust risk prediction model that can be applied to a broader cohort. We consider that this finding warrants future investigation of the factors that lead to the difference and ways to develop a more transferable clinical prediction model for different groups of patients.

5 Conclusion

Clinical text can help predict BI in critically ill patients and NLP models trained using BI patients can be transported to those without BI. NLP and multimodal models can develop better data-driven strategies to stratify the risk of BI in patients, which can be compared with prompt-based large language models (LLMs) in future work. Clinical co-development will be pursued to ensure that the developed models are optimised for clinical workflow, capable of refining antibiotic therapy in the absence of test results, and have the potential to enhance antimicrobial stewardship, thereby mitigating antimicrobial resistance. In the future, we would like to investigate how text can help improve BI treatment, such as antimicrobial stewardship and predict potential antimicrobial resistance.

Acknowledgements

We thank the anonymised reviewers for their comments to help improve the manuscript during the review process.

References

Ashutosh Adhikari, Achyudh Ram, Raphael Tang, and Jimmy Lin. 2019. Rethinking complex neural network architectures for document classification. In *NAACL*, pages 4046–4051.

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In *Clinical Natural Language Processing Workshop*, pages 72–78.

Silvio Amir, Jan-Willem van de Meent, and Byron Wallace. 2021. On the impact of random seeds on the fairness of clinical classifiers. In *NAACL*, pages 3808–3823.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The Long-Document transformer.

Willie Boag, Dustin Doss, Tristan Naumann, and Peter Szolovits. 2018. What's in a note? unpacking predictive value in clinical note representations. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2017:26–34.

Alexander Brown, Nenad Tomasev, Jan Freyberg, Yuan Liu, Alan Karthikesalingam, and Jessica Schrouff. 2023. Detecting shortcut learning for fair medical AI using shortcut testing. *Nature communications*, 14(1):4314.

Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. 2018. Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 8(1):6085.

GBD 2019 Antimicrobial Resistance Collaborators. 2022. Global mortality associated with 33 bacterial pathogens in 2019: a systematic analysis for the global burden of disease study 2019. *The Lancet*, 400(10369):2221–2248.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186.

Iman Deznabi, Mohit Iyyer, and Madalina Fiterau. 2021. Predicting in-hospital mortality by combining clinical notes with time-series data. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4026–4031, Online. Association for Computational Linguistics.

Gurpreet Dhanda, Mirna Asham, Denton Shanks, Nicole O'Malley, Joel Hake, Megha Teeka Satyan, Nicole T Yedlinsky, and Daniel J Parente. 2023. Adaptation and external validation of pathogenic urine culture prediction in primary care using machine learning. *Annals of family medicine*, 21(1):11–18.

Garrett Eickelberg, L Nelson Sanchez-Pinto, and Yuan Luo. 2020. Predictive modeling of bacterial infections and antibiotic therapy needs in critically ill adults. *Journal of biomedical informatics*, 109:103540.

Garrett Eickelberg, Lazaro Nelson Sanchez-Pinto, Adrienne Sarah Kline, and Yuan Luo. 2023. Transportability of bacterial infection prediction models for critically ill patients. *Journal of the American Medical Informatics Association: JAMIA*.

Samuel G Finlayson, Adarsh Subbaswamy, Karandeep Singh, John Bowers, Annabel Kupke, Jonathan Zittrain, Isaac S Kohane, and Suchi Saria. 2021. The clinician and dataset shift in artificial intelligence. *The New England journal of medicine*, 385(3):283–286.

Kim Huat Goh, Le Wang, Adrian Yong Kwang Yeow, Hermione Poh, Ke Li, Joannas Jie Lin Yeow, and Gamaliel Yu Heng Tan. 2021. Artificial intelligence in sepsis early prediction and diagnosis using unstructured data in healthcare. *Nature communications*, 12(1):711.

Jen J Gong and John V Guttag. 2018. Learning to summarize electronic health records using Cross-Modality correspondences. In *Proceedings of the 3rd Machine Learning for Healthcare Conference*, volume 85 of *Proceedings of Machine Learning Research*, pages 551–570. PMLR.

Xudong Han, Timothy Baldwin, and Trevor Cohn. 2021. Diverse adversaries for mitigating bias in training. In *EACL*, pages 2760–2765.

Hrayr Harutyunyan, Hrant Khachatrian, David C Kale, Greg Ver Steeg, and Aram Galstyan. 2019. Multitask learning and benchmarking with clinical time series data. *Scientific data*, 6(1):96.

Chao-Chun Hsu, Shantanu Karnwal, Sendhil Mullainathan, Ziad Obermeyer, and Chenhao Tan. 2020. Characterizing the value of information in medical notes. In *Findings of EMNLP 2020*, pages 2062–2072.

Shih-Cheng Huang, Anuj Pareek, Saeed Seyyedi, Imon Banerjee, and Matthew P Lungren. 2020. Fusion of medical imaging and electronic health records using deep learning: a systematic review and implementation guidelines. *NPJ digital medicine*, 3:136.

Alistair E W Johnson, Tom J Pollard, Lu Shen, Li-Wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific data*, 3:160035.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *EMNLP*, pages 1746–1751.

Fajri Koto, Jey Han Lau, and Timothy Baldwin. 2021. IndoBERTweet: A pretrained language model for Indonesian Twitter with effective Domain-Specific vocabulary initialization. In *EMNLP*, pages 10660–10668.

Joshua Lemmon, Lin Lawrence Guo, Ethan Steinberg, Keith E Morse, Scott Lanyon Fleming, Catherine Aftandilian, Stephen R Pfohl, Jose D Posada, Nigam Shah, Jason Fries, and Lillian Sung. 2023. Self-supervised machine learning using adult inpatient data produces effective models for pediatric clinical prediction tasks. *Journal of the American Medical Informatics Association: JAMIA*.

Yikuan Li, Ramsey M Wehbe, Faraz S Ahmad, Hanyin Wang, and Yuan Luo. 2023. A comparative study of pretrained language models for long clinical text. *Journal of the American Medical Informatics Association: JAMIA*, 30(2):340–347.

Jinghui Liu, Daniel Capurro, Anthony Nguyen, and Karin Verspoor. 2022a. Improving text-based early prediction by distillation from privileged Time-Series text. In *Proceedings of the The 20th Annual Workshop of the Australasian Language Technology Association*, pages 73–83, Adelaide, Australia. Australasian Language Technology Association.

Jinghui Liu, Daniel Capurro, Anthony Nguyen, and Karin Verspoor. 2022b. "Note Bloat" impacts deep learning-based NLP models for clinical prediction tasks. *Journal of biomedical informatics*, 133:104149.

Jinghui Liu, Daniel Capurro, Anthony Nguyen, and Karin Verspoor. 2023. Attention-based multimodal fusion with contrast for robust clinical prediction in the face of missing modalities. *Journal of biomedical informatics*, 145:104466.

Ran Liu, Joseph L Greenstein, Stephen J Granite, James C Fackler, Melania M Bembea, Sridevi V Sarma, and Raimond L Winslow. 2019. Data-driven discovery of a novel sepsis pre-shock state predicts impending septic shock in the ICU. *Scientific reports*, 9(1):6145.

Michael Moor, Max Horn, Bastian Rieck, Damian Roqueiro, and Karsten Borgwardt. 2019. Early recognition of sepsis with gaussian process temporal convolutional networks and dynamic time warping. In

Proceedings of the 4th Machine Learning for Healthcare Conference, volume 106 of *Proceedings of Machine Learning Research*, pages 2–26. PMLR.

Michael Moor, Bastian Rieck, Max Horn, Catherine R Jutzeler, and Karsten Borgwardt. 2021. Early prediction of sepsis in the ICU using machine learning: A systematic review. *Frontiers of medicine*, 8:607952.

Pranav Rajpurkar, Emma Chen, Oishi Banerjee, and Eric J Topol. 2022. AI in health and medicine. *Nature medicine*, 28(1):31–38.

Yulia Rubanova, Ricky T Q Chen, and David K Duvenaud. 2019. Latent ordinary differential equations for Irregularly-Sampled time series. In *NeurIPS*, volume 32.

Tom M Seinen, Egill A Fridgeirsson, Solomon Ioannou, Daniel Jeannetot, Luis H John, Jan A Kors, Aniek F Markus, Victor Pera, Alexandros Rekkas, Ross D Williams, Cynthia Yang, Erik M van Mulligen, and Peter R Rijnbeek. 2022. Use of unstructured text in prognostic clinical prediction models: a systematic review. *Journal of the American Medical Informatics Association: JAMIA*, 29(7):1292–1302.

Luis R Soenksen, Yu Ma, Cynthia Zeng, Leonard Boussioux, Kimberly Villalobos Carballo, Liangyuan Na, Holly M Wiberg, Michael L Li, Ignacio Fuentes, and Dimitris Bertsimas. 2022. Integrated multimodal artificial intelligence framework for healthcare applications. *NPJ digital medicine*, 5(1):149.

R Andrew Taylor, Christopher L Moore, Kei-Hoi Cheung, and Cynthia Brandt. 2018. Predicting urinary tract infections in the emergency department with machine learning. *PloS one*, 13(3):e0194085.

Eric J Topol. 2019. High-performance medicine: the convergence of human and artificial intelligence. *Nature medicine*, 25(1):44–56.

Shirly Wang, Matthew B A McDermott, Geeticka Chauhan, Marzyeh Ghassemi, Michael C Hughes, and Tristan Naumann. 2020. MIMIC-Extract: a data extraction, preprocessing, and representation pipeline for MIMIC-III. In *ACM Conference on Health, Inference, and Learning*, pages 222–235.

Felix Wong, Cesar de la Fuente-Nunez, and James J Collins. 2023. Leveraging artificial intelligence in the fight against infectious diseases. *Science*, 381(6654):164–170.

Melissa Y Yan, Lise Tuset Gustad, and Øystein Nytrø. 2022. Sepsis prediction, early detection, and identification using clinical text for machine learning: a systematic review. *Journal of the American Medical Informatics Association: JAMIA*, 29(3):559–575.

Ying Yang, Yi-Min Wang, Chun-Hung Richard Lin, Chi-Yung Cheng, Chi-Ming Tsai, Ying-Hsien Huang, Tien-Yu Chen, and I-Min Chiu. 2023. Explainable deep learning model to predict invasive bacterial infection in febrile young infants: A retrospective study. *International journal of medical informatics*, 172:105007.

Dongyu Zhang, Jidapa Thadajarassiri, Cansu Sen, and Elke Rundensteiner. 2020. Time-Aware transformer-based network for clinical notes series prediction. In *Proceedings of the 5th Machine Learning for Healthcare Conference*, volume 126 of *Proceedings of Machine Learning Research*, pages 566–588. PMLR.

Yijia Zhang, Qingyu Chen, Zhihao Yang, Hongfei Lin, and Zhiyong Lu. 2019. BioWordVec, improving biomedical word embeddings with subword information and MeSH. *Scientific data*, 6(1):52.

Hongyi Zheng, Yixin Zhu, Lavender Jiang, Kyunghyun Cho, and Eric Oermann. 2023. Making the most out of the limited context length: Predictive power varies with clinical note type and note section. In *The 61st ACL (Student Research Workshop)*, pages 104–108.

A Hyperparamter Tuning

For TextCNN, BiLSTM, and BERT+LSTM models, we sweep through the space: number of RNN hidden state/CNN filter number $\in [128, 256, 512]$; dropout rate $\in [0.2, 0.4, 0.6]$; weight decay $\in [0, 0.01]$; learning rate $\in [1e-3, 1e-4]$. The batch size is kept as 32. For BERT fine-tuning, we explore epoch $\in [3, 5, 10]$ and learning rate $\in [2e-5, 3e-5, 5e-5]$. For Longformer fine-tuning, we explore learning rate $\in [1e-5, 2e-5, 5e-5]$ and kept epoch as 5 to save computation. The batch size for the two models is kept as 16 using gradient accumulation.

B Constructing GENERAL Cohort

We follow the criteria in previous work (Hsu et al., 2020; Harutyunyan et al., 2019) to select this cohort of patients and use notes charted before 24 hours of admission to the ICU as input, the same as in the BI cohort. There are three criteria for selection: 1) adult patients, 2) no repeated ICU admissions, and 3) hospital discharge time is at least 30 hours away from ICU admission. Table 5 shows the statistics of the cohort.

	Train	Validation	Test
Num of cases	30162	4475	10320
Mortality rate	10.2%	10.4%	9.6%
Avg num of notes	8.7	8.6	8.7
Avg num of words	2440.3	2432.8	2468.4

Table 5: GENERAL cohort statistics.

Predicting Empathic Accuracy from User-Designer Interviews

Steven Nguyen[*] **Daniel Beck**[*] **Katja Hölttä-Otto**[**]
[*]School of Computing and Information Systems, The University of Melbourne, Australia
[**]Department of Mechanical Engineering, The University of Melbourne, Australia
psnguyen@student.unimelb.edu.au d.beck@unimelb.edu.au
katja.holttaotto@unimelb.edu.au

Abstract

Measuring empathy as a natural language processing task has often been limited to a subjective measure of how well individuals respond to each other in emotive situations. Cognitive empathy, or an individual's ability to accurately assess another individual's thoughts, remains a more novel task. In this paper, we explore natural language processing techniques to measure cognitive empathy using paired sentence data from design interviews. Our findings show that an unsupervised approach based on similarity of vectors from a Large Language Model is surprisingly promising, while adding supervision does not necessarily improve the performance. An analysis of the results highlights potential reasons for this behaviour and gives directions for future work in this space.[1]

1 Introduction

User interviews are an important part of modern product development frameworks as meeting user needs defines success in Engineering Design. Typically these interviews, conducted between a potential user and a designer, are used to either gather knowledge about the user's problem or their experiences with current products, or to gain feedback on the product as it is being developed. However it remains a question as to whether these processes improve user understanding and lead to good outcomes, and the factors which contribute to these.

One such factor regards whether or not designers are able to understand the user during these interviews - this is referred to as 'empathic understanding' (Surma-aho and Hölttä-Otto, 2022). If a designer is able to grasp the user's experiences and thoughts, does this necessarily lead to better outcomes? To answer this question, Chang-Arana et al. (2020) developed a method borrowed from the social sciences to quantitatively measure empathic understanding through interviews. The method requires laborious manual annotation, involving the original user-designer pair and additional raters.

In this paper, we propose to use natural language processing (NLP) approaches to automate the measurement of empathic understanding in interviews, especially due to the advent of out-of-the-box Large Language Models (LLMs). This can not only streamline the process of analysing interviews in Engineering Design but also provide a test bed for automatically measuring empathy in conversations, an open problem in NLP. Automated evaluation in this way may be useful more broadly in other fields, where empathy is highly valued, such as teaching.

2 Background and Related Work

Work measuring empathy in NLP has been explored, with open domain dialogue data such as `EmpatheticDialogues` (Rashkin et al., 2019) existing as benchmarks for the task. Much work has been done detecting how empathy is expressed in dialogues in a variety of contexts from healthcare (Sharma et al., 2020; Xiao et al., 2015) in both speech and text, as well as in online communities (Zhou and Jurgens, 2020). However the theme of these works is primarily focused on empathy in the emotional sense. That is, there is a large focus on studying how individual express empathy towards others through dialogues. A common example is choosing the 'right' emotional words to comfort another individual in distress, guiding work in generating empathetic responses (Welivita et al., 2021).

On the other hand, in a review, Lahnala et al. (2022) points out that tasks revolving around *cognitive empathy* are not as prevalent in the NLP literature. While empathy is a complex concept, loosely we can distinguish between *emotional empathy* as

---

[1] Code used for our experiments is available at `https://github.com/owowouwu/empathic-accuracy`. Data is available under request to Katja Hölttä-Otto, `katja.holttaotto@unimelb.edu.au`.

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 125–129
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

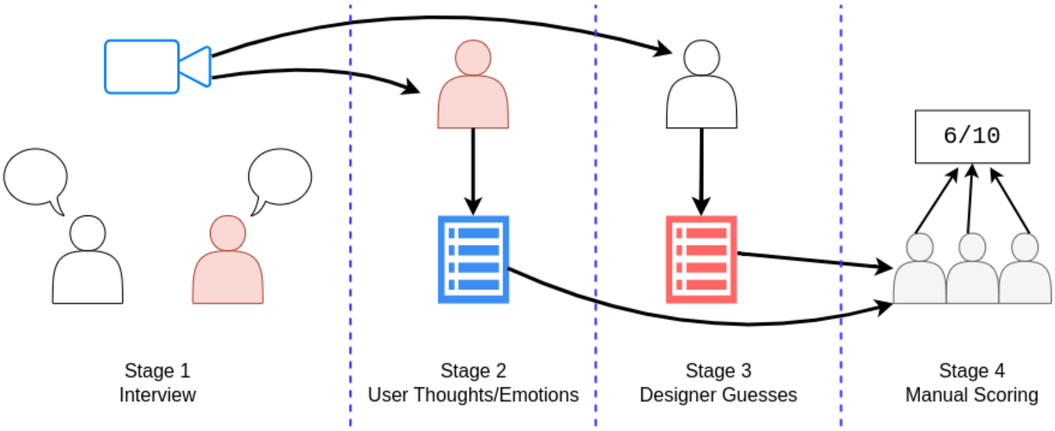


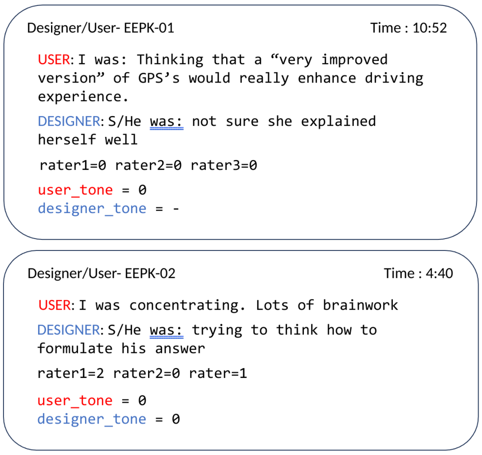
Figure 1: Overview of the collection of empathic accuracy ratings for interviews. Stage 1 represents the original user-designer interview, which is recorded. The same user then write their thoughts in Stage 2, with the designer aiming at guessing these thoughts in Stage 3. The annotation process finishes at Stage 4, where a set of human raters (3 in this case) assign a score to each aligned thought and guess, with the final empathic accuracy score being the average of these ratings.

processing and responding to another's emotions effectively, and *cognitive empathy* as being able to infer their thoughts in a broader sense (Cuff et al., 2016). One may be able to identify how another person is feeling and act appropriately, but may not necessarily know what the other person is thinking. A key distinction between our work and more common tasks involving 'empathy' in NLP is that we primarily try to measure cognitive empathy from pairs of thoughts.

3 Data

The dataset was collected from user-designer interview experiments in Salmi et al. (2023). Figure 1 gives an overview of the annotation process for empathic accuracy ratings. Each interview was recorded in video format. Interviewees were played back the recording and were asked at any time to pause the video and write down their thoughts. The same recording was played back to the interviewer, where they were tasked with guessing the user's thoughts in those moments.

In total, 46 users were interviewed by 3 designers, although not every user and designer were paired. Each instance of the dataset is indexed by a (user, designer) pair and contains a timestamped sentence pair - one being the user's thoughts at that particular moment, and the other being a guess of the user's thoughts by the designer at the same moment. Each pair is rated by 3 judges with a three-level Likert scale ($\{0, 1, 2\}$), with the average taken as a score indicating the accuracy of the designer's prediction. The designer is also tasked with predicting the user's self-evaluated tone of speech at that moment.


Designer/User- EEPK-01 Time : 10:52

USER: I was: Thinking that a "very improved version" of GPS's would really enhance driving experience.

DESIGNER: S/He was: not sure she explained herself well

rater1=0 rater2=0 rater3=0

user_tone = 0
designer_tone = -

Designer/User- EEPK-02 Time : 4:40

USER: I was concentrating. Lots of brainwork

DESIGNER: S/He was: trying to think how to formulate his answer

rater1=2 rater2=0 rater=1

user_tone = 0
designer_tone = 0

Figure 2: Example instances of data.

In this work, we focus on automating the rating stage (Stage 4 in Figure 1). Each instance contains a sentence pair (a user thought paired with a designer guess) as the input and the averaged rate given by the judges as the output. Figure 2 shows two such instances as an example. The inputs were preprocessed by removing text indicating the subject ("he/she/I was:") at the start of the string. We also rescaled the ratings to the unit interval. Table 1 details the statistics of our dataset.

4 Methods

All our models use Sentence BERT (Reimers and Gurevych, 2019, SBERT) as the LLM backbone, generating two embedding vectors for each pair

Designer	Instances	Avg. Score
1	120	0.414
2	129	0.519
3	200	0.398
All	449	0.437

Table 1: Summary statistics of our dataset. For each designer, we report their corresponding total number of sentence pairs and its average similarity score.

of user thought and designer guess. Experiments with in-domain supervised models were performed using 10-fold cross validation. For a sound comparison, we use the same 10-fold setup for the unsupervised and out-of-domain models, using only the testing folds for evaluation.

Unsupervised. Our first approach does not employ any training: we calculate the cosine similarity between the two embedding vectors and report the result as the rating.

In-Domain Supervised. Here we employ a standard cross-validation procedure, using 9 folds as training data. Our main approach finetunes a SBERT regression model following the original "siamese" method from (Reimers and Gurevych, 2019), which uses the cosine similarity between the embedding vectors as the regression output. In addition, we also employed SBERT as a feature extractor and two off-the-shelf regressors as additional models: a Gaussian Process (Rasmussen and Williams, 2006, GP) with an RBF kernel and a Multilayer Perceptron (MLP). Each input uses the concatenation of the SBERT vectors obtained from the user thought and the designer guess, plus the vector obtained from their absolute difference. On average, each training set contains 400 pairs.

Out-of-Domain Supervised. Finally, we also tested with a supervised approach trained on out-of-domain data. The rationale is that the rating can be framed as a Semantic Textual Similarity problem (Corley and Mihalcea, 2005, STS). This raises the question of whether we can employ existing STS data to create a good regressor without requiring any initial ratings for training. For these experiments, we used the widely available STS-B (Cer et al., 2017) dataset, containing approximately 6000 pairs. We used the same models as in the in-domain experiments.

5 Results

Our main results are shown in Table 2, using both Pearson's correlation and Root Mean Squared Error as evaluation metrics. As expected, the finetuned model on in-domain data gives the best performance. However, notably, it is not significantly better than the unsupervised model, potentially due to the limited amount of training data. This is further evidenced by the poor performance of the off-the-shelf regressors.

The models trainedd on the out-of-domain STS-B data did not outperform the unsupervised approach for any regressors. We believe this is due to significant differences in the STS-B and the Interview data. While both can be interpreted as sentence similarity, the pairs present in STS-B are much shorter and use simpler language, compared to the more complex sentences present in our dataset. While we were aware of this important domain difference, we still expected the performance to be better than the unsupervised approach, but our findings showed otherwise.

It is important to note that a Pearson score of 0.66 already demonstrates good prediction performance. Performance improvements could be obtained by adding in-domain training data and further model tuning. However, these results are already promising from an application perspective and could potentially lead to a reduction in human labour for obtaining empathic accuracy scores.

6 Qualitative Analysis

Here we will conduct further analysis on our data to understand the performance of our models under our task. We summarise three findings that could lead to further improvements in the prediction task.

Lack of Non-textual Context Textual similarity tasks rely on the meaning and context within the sentence itself, but in our case did not contain the extra information that raters may have when scoring pairs of text. The thoughts are often written down in an ad-hoc and conversational manner, containing *implied* information around the topic or interview itself that is able to be inferred by the raters, but which models which rely on complete information fail to do. This causes a mismatch between true scores and predicted scores. Our first instance in Table 3 shows this, as the designer is implicitly referring to the "AI system" that the user is mentioning, and is thus scored highly by the

Model	STS-B Test		Interviews	
	Pearson↑	RMSE↓	Pearson↑	RMSE↓
Unsupervised				
Cosine Similarity w/ SBERT	0.836	0.225	0.662 ± 0.060	0.227 ± 0.015
In-Domain Supervised				
Gaussian Process	-	-	0.562 ± 0.102	0.234 ± 0.022
Multilayer Perceptron	-	-	0.481 ± 0.153	0.263 ± 0.033
Finetuned SBERT	-	-	**0.680 ± 0.050**	**0.215 ± 0.019**
Out-of-Domain Supervised				
Gaussian Process	0.828	0.171	0.534 ± 0.074	0.240 ± 0.016
Multilayer Perceptron	0.800	0.191	0.515 ± 0.123	0.252 ± 0.028
Finetuned SBERT	0.858	2.424	0.618 ± 0.061	0.226 ± 0.017

Table 2: Summary of results. RMSE denotes root mean squared error. For the interview data, we report the average and standard deviation over 10 folds.

rater, but SBERT fails as in a vaccuum these two sentences do not have the same meaning without knowing what the designer refers to.

User: *thinking that this is quite hard to do in some kind of ai system*
Designer: *its technically hard to detect pedestrians*
True Score: 0.833
Predicted Score: 0.156

User: *you could just ask me what you want me to provide*
Designer: *feeling confused about the question and didn't know what answers the interviewer wants*
True Score: 0.833
Predicted Score: 0.249

Table 3: Example predictions for interview data.

Inconsistent Points of View Within our data it is often the case that the two pairs of text are written from two different points of view, resulting in sentences that may have similar content, but have different meaning. However they may still be rated highly because the designer, in their own writing, has effectively guessed the user's thoughts, even if they are not writing the thoughts from the perspective of the user.

Judge Scoring Our methods also tend to overestimate the scores in cases where the context or topic that both the designer and user are thinking of are the same, but the actual user text was different. For example, because the interviews were related to driving, both the user and designer wrote down thoughts related to driving, but these thoughts did not necessarily contain the same idea. In these cases, the human judges tended to more harshly assign scores of 0 whereas our system tended to provide a more soft assignment. This is a common problem of standard regression models, which are unable to predict extreme values outside a certain range. Future work should carefully consider how to penalise the scores based on how the two sentences diverge in actual meaning.

7 Conclusion

We introduce a novel task of predicting an individual's cognitive empathy as scored by their ability to predict, in text, the thoughts of another individual using a dataset from design engineering interviews. Using this data we demonstrate the performance and limitations of current state of the art models on our task. Our analysis shows that this problem poses unique challenges due to the unique structure and missing context of user written thoughts.

Initial directions for future work are based on our analysis in Section 6. Incorporating context from interview transcripts is an important direction, as well as improved regression models that can better predict extreme values. A more challenging, longer term goal is the prediction of empathic accuracy *directly from interviews*, without requiring user thoughts and designer guesses. This would effectively bypass Stages 2 and 3 in Figure 1, drastically reducing annotation costs and potentially enabling real-time empathy feedback *during an interview*. We believe this is a much harder problem, but that nevertheless would lead to benefits to not just our task in engineering design, but lead to novel advances in other tasks in NLP.

Acknowledgements

This research was supported by The University of Melbourne's Research Computing Services, the Petascale Campus Initiative and a CIS-ME grant (2022) from The University of Melbourne.

References

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. Association for Computational Linguistics.

Álvaro M. Chang-Arana, Matias Piispanen, Tommi Himberg, Antti Surma-aho, Jussi Alho, Mikko Sams, and Katja Hölttä-Otto. 2020. Empathic accuracy in design: Exploring design outcomes through empathic performance and physiology. *Design Science*, 6:e16.

Courtney Corley and Rada Mihalcea. 2005. Measuring the semantic similarity of texts. In *Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, pages 13–18. Association for Computational Linguistics.

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.

Allison Lahnala, Charles Welch, David Jurgens, and Lucie Flek. 2022. A critical reflection and forward perspective on empathy and natural language processing. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2139–2158, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards Empathetic Open-domain Conversation Models: A New Benchmark and Dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381, Florence, Italy. Association for Computational Linguistics.

Carl Edward Rasmussen and Christopher K. I. Williams. 2006. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press. OCLC: ocm61285753.

Nils Reimers and Iryna Gurevych. 2019. 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.

Aleksi Salmi, Jie Li, and Katja Holtta-Otto. 2023. Automatic Facial Expression Analysis as a Measure of User-Designer Empathy. *Journal of Mechanical Design*, 145(3):031403.

Ashish Sharma, Adam Miner, David Atkins, and Tim Althoff. 2020. A Computational Approach to Understanding Empathy Expressed in Text-Based Mental Health Support. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5263–5276, Online. Association for Computational Linguistics.

Antti Surma-aho and Katja Hölttä-Otto. 2022. Conceptualization and operationalization of empathy in design research. *Design Studies*, 78:101075.

Anuradha Welivita, Yubo Xie, and Pearl Pu. 2021. A Large-Scale Dataset for Empathetic Response Generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1251–1264, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Bo Xiao, Zac E. Imel, Panayiotis G. Georgiou, David C. Atkins, and Shrikanth S. Narayanan. 2015. "Rate My Therapist": Automated Detection of Empathy in Drug and Alcohol Counseling via Speech and Language Processing. *PLoS One*, 10(12):e0143055.

Naitian Zhou and David Jurgens. 2020. Condolence and Empathy in Online Communities. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 609–626, Online. Association for Computational Linguistics.

CRF-based recognition of invasive fungal infection concepts in CHIFIR clinical reports

Yang Meng
Department of Computer Science,
University of Chicago
ymeng3@uchicago.edu

Vlada Rozova
Centre for Digital Transformation of Health,
University of Melbourne
vlada.rozova@unimelb.edu.au

Karin Verspoor
School of Computing Technologies, RMIT University
karin.verspoor@rmit.edu.au

Abstract

Named entity recognition (NER) in clinical documentation is often hindered by the use of highly specialised terminology, variation in language used to express medical findings and general scarcity of high-quality data available for training. This short paper compares a Conditional Random Fields model to the previously established dictionary-based approach and evaluates its ability to extract information from a small corpus of annotated pathology reports. The results suggest that including token descriptors as well as contextual features significantly improves precision on several concept categories while maintaining the same level of recall.

1 Introduction

Invasive fungal infections (IFIs) are a significant medical concern, particularly, among immunocompromised individuals. These infections, caused by fungal pathogens that breach the body's primary barriers and infiltrate deeper tissues or disseminate through the bloodstream, can lead to severe morbidity and heightened mortality rates. Early detection and appropriate antifungal treatment are paramount, but they may be difficult to identify in clinical populations (Even et al., 2011).

To support IFI surveillance, Rozova et al. (2023b) sought to establish an automated system to identify markers of IFI in cytology and histopathology reports. The authors introduced a corpus called CHIFIR (Rozova et al., 2023a), the Cytology and Histopathology Invasive Fungal Infection Reports, to support the development and evaluation of NLP methods for concept recognition of clinical concepts relevant to IFIs.They constructed an annotation framework to detect specific terms directly indicative of a confirmed IFI diagnosis. Central to their methods was a dictionary-based approach, which relied on exact term matches in texts.

However, the dictionary-based approach has several limitations:

- Lexical variation: the same entity can be described in different ways which complicates the task of exact matching. As an illustration, while "lung" is categorized as *Positive*, its synonym "pulmonary" is not recognized by the dictionary.
- Context is paramount: a term can convey different meanings based on its surrounding text and where in the report it is located. For instance, while "cryptococcal organism" is classified as "Fungus", the term "organism" alone may refer to bacteria, fungi, etc.

In contrast, Machine Learning (ML) algorithms, when compared with dictionary methods, present a promising alternative. These algorithms have the capability to learn the patterns of usage of relevant concepts or entities, based on consideration of the context of words.

In this work, we aim to explore the effectiveness of the ML approach by applying Conditional Random Fields (Lafferty et al., 2001) to the CHIFIR dataset and comparing its performance with the original dictionary-based solution.

The following sections will delve deeper into the methodology and outline the results of this comparison, highlighting the advantages of CRF over the dictionary approach.

2 Background

Histopathology reports are structured documents that outline findings from microscopic examination of biopsied tissue. The language used is specialized, often employing a combination of medical terminology, abbreviations, and sometimes subjective descriptions based on the pathologist's observations and interpretations. The complexity and variability of the narrative, which can differ between

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 130–135
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

pathologists and institutions, make standardization difficult. Subtle nuances and contextually driven interpretations are pivotal in histopathology, making it challenging for algorithms to consistently interpret and draw accurate conclusions. Moreover, the occasional use of ambiguous or equivocal terms to describe uncertain or borderline findings can further complicate machine interpretation.

Extracting relevant concepts from clinical reports is part of a broader field of information extraction (IE). Several rule-based systems and dictionary-based entity recognition tools have been proposed offering more flexibility to combat the inherent variability in language. For instance, Funk et al. (2014) compares a ConceptMapper (Tanenblatt et al., 2010) based system with MetaMap (Aronson and Lang, 2010). While these methods offer reliability and precision, they still lack the ability to make context-specific interpretations.

In this paper, we will focus on applying Conditional Random Fields (CRFs) (Sutton et al., 2012; Sha and Pereira, 2003; Lafferty et al., 2001), which have been successfully applied to the related task of named entity recognition. CRFs are a class of statistical modeling methods and are particularly well-suited for sequence labeling tasks. CRFs consider the entire sequence, allowing for a more comprehensive contextual understanding. Additionally, CRFs are capable of ingesting a diverse set of features which can be helpful in dealing with linguistic nuances and inconsistencies across different reports. The model's flexibility enables it to effectively handle ambiguities in clinical narratives. One successful example of CRF implementation for biomedical entity recognition is BANNER (Leaman and Gonzalez, 2008).

It is worth noting that the rapid advancements in deep learning have led to the emergence of more sophisticated models, such as LSTMs and Transformer-based architectures. Such models can automatically extract features and have demonstrated superior performance across a variety of NLP tasks (Chiu and Nichols, 2016) (Santos et al., 2015). Recent literature has suggested that the use of contextualised lexical representations (e.g. in BERT (Vaswani et al., 2017)) as well as the ability to capture long-range dependencies and semantic relationships in text (Lample et al., 2016) may be particularly useful in the complex and nuanced domain of histopathology reports. However, such models might not be effective in learning IFI-specific terms because of the small and specialised nature of the CHIFIR dataset.

3 Methods

3.1 Dataset

The dataset employed for this research is the CHIFIR corpus (Rozova et al., 2023a)[1], consisting of 283 cytology and histopathology reports pertaining to 201 patients.

A characteristic feature of the cytology and histopathology reports is their extended textual format, with CHIFIR reports having an average character count of 1,384. These reports have a semi-structured layout, with headers delineating various segments for clinical annotations, macroscopic assessments, microscopic evaluations, and conclusive diagnoses.

3.2 Preparation of dataset

In this study, partitioning into development (n=230) and test (n=53) sets was replicated exactly from the original study. To ensure the results are comparable to the original study, the same stratified group k-fold cross-validation with 10 splits was applied to the development set.

Using gold standard annotations, we identified known concepts in text reports and labeled them with the corresponding categories (Table 1). The class distribution of labels is displayed in Table 2. The remaining text was tokenized into individual tokens, and each token was labeled with a default `O` label.

3.3 Model & Features

We utilized CRFSuite (Lafferty et al., 2001) as an implementation for the model and a proper set of features is needed to capture the underlying patterns in the data. We expect these features should be able to generalize, i.e., correctly discriminate the entities on new samples.

We included features that offer information on how a word appears in the text (i.e., capitalization, prefixes, suffixes) and its context. We conducted an empirical evaluation to refine the feature set: we experimented with adding semantic features, such as POS tags and special characters; sentence-level position features, such as if the word is at the start or the end of a sentence; and word-level context features, such as previous word and next word. Contrary to our intuition, the inclusion of

---

[1] `https://physionet.org/content/corpus-fungal-infections/1.0.0/`

Concept	**Description**
ClinicalQuery	Clinical query of IFI indicates the presence of an IFI.
FungalDescriptor	Descriptor for the presence of fungal organism.
Fungus	Mentions of specific fungal organisms.
Invasiveness	Descriptors for the depth and degree of fungal invasion into tissues.
Stain	Histological stains used to visualize fungal elements.
SampleType	Specification of the sampled organ, site, or tissue source.
Positive	Affirmative expression.
Equivocal	Expression of uncertainty.
Negative	Negating expression.

Table 1: List of concepts related to the IFI diagnosis.

Concept	**Total occurrences**	**#reports with at least one occurence**	**#unique phrases**	**Lexical diversity**
ClinicalQuery	65	53	36	0.55
FungalDescriptor	282	128	67	0.24
Fungus	106	60	15	0.14
Invasiveness	37	12	25	0.68
Stain	172	100	13	0.08
SampleType	198	179	55	0.28
Positive	118	42	37	9.31
Equivocal	7	5	5	0.71
Negative	152	104	11	0.07

Table 2: Summary statistics for the IFI-related concepts in the CHIFIR dataset.

those features either did not improve or worsened the performance of the model. The final list of included features appears in Table 3.

3.4 Experimental Framework

We tokenize each report and extract relevant features as described above. To tune hyperparameters and refine the feature set, we used cross-validation whereby within each fold, a CRF model is initialized with ‘lbfgs’ algorithm and a maximum iteration of 100. The final model with hyperparameters c1=0.01 and c2=0.01 was trained on the entire training dataset to generate predictions on the test set.

For evaluation, we used full-term identification. We calculated the number of true positive, false positive, and false negative concepts in each report by comparing the predictions to the gold standard annotations. For each concept category, we summarize model performance using precision and recall, and record incorrectly identified concepts for error analysis.

4 Results

4.1 Overview

Overall, the CRF approach outperformed the dictionary-based approach utilized in the original paper (Rozova et al., 2023b). Table 4 shows a significantly higher precision in detecting categories *FungalDescriptor*, *SampleType*, *Positive*, and *Negative*. For other concept categories, the CRF model had on average higher precision although the difference was not statistically significant. Table 5 shows that recall is on average comparable to that of the dictionary-based approach. Table 6 summarises the performance as F1 score showing significant improvement in categories *SampleType*, *Positive*, *Equivocal*, and *Negative*.

4.2 Strengths

First, let us consider the challenge of lexical variation. The ability of the dictionary-based approach to generalize is limited; to make a correct prediction a concept has to appear in the same form as in the training sample. For our CRF model, we found

Feature	Description
`word`	The word itself.
`start_pos and end_pos`	The start and end position of the word.
`is_capitalized`	Checks if the first letter is capitalized.
`is_all_caps and is_all_lower`	Check for casing details.
`capitals_inside`	Checks if there are capital letters inside the word.
`prefix and suffix`	Use the 3 prefix and 3 suffix characters of each word as context.
`has_hyphen`	Whether the word has hyphens.
`is_numeric`	Whether the word has numeric.

Table 3: List of features.

Concept	Precision CV Dict	Precision CV CRF	Precision TEST Dict	Precision TEST CRF
ClinicalQuery	0.92 (±0.13)	0.83 (±0.20)	1.00	1.00
FungalDescriptor	0.75 (±0.10)	0.92 (±0.05)	0.68	0.98
Fungus	0.82 (±0.30)	0.95 (±0.07)	0.88	0.94
Invasiveness	0.45 (±0.41)	0.69 (±0.41)	0.33	1.00
Stain	0.94 (±0.05)	0.97 (±0.05)	1.00	0.97
SampleType	0.15 (±0.03)	0.92 (±0.08)	0.14	1.00
Positive	0.04 (±0.02)	0.82 (±0.16)	0.03	1.00
Equivocal	0.01 (±0.02)	1.00 (±NaN)	0.00	0.00
Negative	0.14 (±0.04)	0.97 (±0.05)	0.15	1.00

Table 4: Comparison of dictionary and CRF approach precision during cross-validation and on unseen test data.

Concept	Recall CV Dict	Recall CV CRF	Recall TEST Dict	Recall TEST CRF
ClinicalQuery	0.53 (±0.35)	0.72 (±0.20)	0.69	1.00
FungalDescriptor	0.93 (±0.04)	0.90 (±0.05)	0.93	0.96
Fungus	0.92 (±0.15)	0.88 (±0.16)	0.94	0.94
Invasiveness	0.60 (±0.39)	0.63 (±0.30)	0.12	0.50
Stain	0.95 (±0.09)	0.98 (±0.04)	1.00	1.00
SampleType	0.86 (±0.10)	0.81 (±0.11)	0.86	0.79
Positive	0.83 (±0.17)	0.89 (±0.13)	0.73	0.95
Equivocal	0.58 (±0.50)	0.20 (±0.45)	0.00	0.00
Negative	0.98 (±0.05)	0.96 (±0.08)	0.90	1.00

Table 5: Comparison of dictionary and CRF approach recall during cross-validation and on unseen test data.

that about 82% of the correctly predicted concepts in the test set were exact matches from the training set, and the rest were variations of known concepts.

The CRF model can identify and combine parts of annotated concepts. For instance, "branching hyphae" was not present in the training set. CRF generalizes "branching" and "hyphae" by learning from two concepts in the training data, "acute angle branching" and "septate hyphae", which were annotated as *FungalDescriptor*. The suffix "cosis" was also captured as an indicator of the *Fungus* category. The model captures linguistic/capitalization/syntax variations, for instance, "duodenum" is generalized from "duodenal", and "groccot" from "Groccot". Besides, CRF demonstrated the ability to learn complex patterns: "? infection PJP" is detected based on the *FungalDescriptor* "PJP" present in the training data and the fact that a "?" followed by a *FungalDescriptor* often makes up a *ClinicalQuery*. The model captures

Concept	**F1** **CV Dict**	**F1** **CV CRF**	**F1** **TEST Dict**	**F1** **TEST CRF**
ClinicalQuery	0.68 (±0.27)	0.75 (±0.16)	0.81	1.00
FungalDescriptor	0.83 (±0.07)	0.91 (±0.03)	0.79	0.97
Fungus	0.91 (±0.09)	0.90 (±0.09)	0.91	0.94
Invasiveness	0.71 (±0.25)	0.68 (±0.27)	0.18	0.67
Stain	0.94 (±0.05)	0.97 (±0.03)	1.0	0.98
SampleType	0.26 (±0.04)	0.86 (±0.08)	0.24	0.88
Positive	0.08 (±0.03)	0.84 (±0.10)	0.05	0.97
Equivocal	0.05 (±0.03)	1.00 (±NaN)	NaN	NaN
Negative	0.24 (±0.06)	0.96 (±0.04)	0.26	1.00

Table 6: Comparison of dictionary and CRF approach F1 during cross-validation and on unseen test data.

the intuition that certain labels are more likely to appear after certain other labels. Lastly, phrases not present in the training data, such as "punch biopsies", "pericardium", and "abdomen" were correctly predicted, showing that the model can make inferences based on relevant contexts.

Secondly, the model did a generally good job of addressing ambiguity in the medical text. Words such as "organism" and "capsule" were consistently overdetected when using the dictionary-based approach, resulting in a high false-positive rate. The CRF model has correctly picked out the relevant mentions considering their context.

4.3 Weaknesses

In general, the detection of concepts belonging to *SampleType* and *Invasiveness* categories showed to be the most challenging, making up 45% and 17% of the total error cases, respectively. The errors were largely due to the relatively modest size of the training data, high lexical diversity and fewer occurrences in the dataset.

The modest recall characteristic of the *Invasiveness* category is likely due to high lexical diversity and longer phrases consisting of multiple tokens. For example, the model failed to classify phrases "tissue invasion" and "vessel lung parenchyma infiltrated" as *Invasiveness* concepts, even though individual words "invasiveness", "vessel", and "parenchyma" were frequently occurring in the training data. It is possible that engineering a more extensive contextual feature set is required to tackle such cases.

Some words did not appear in the training data and thus the model may have never learned an appropriate representation. This can be seen in examples involving both medical terms (e.g., "ileum", "cyst") and generic English words (e.g., "back", "leg").

The features used in the model may also occasionally be misleading. For instance, the word "RUL" is misclassified as *Stain* because a common *Stain* concept "PAS" usually appears in uppercase. Thus the model may associate the upper case with that label, illustrating an example of the model giving form much more weight than context.

5 Conclusion

In conclusion, we have seen that the CRF model performes better and, in particular, is more successful in tackling the lexical diversity and variation present in the CHIFIR corpus than the previous dictionary-based method. Although the model performance still suffers from the small sample size and challenging lexical diversity cases, we demonstrated that incorporation of context through the CRF-based concept recognition model benefits development of clinical concept recognition tools for this corpus. It would also be worth exploring and comparing this CRF-based approach with more advanced machine learning methods, which might be able to learn richer representations from data, and overcome challenges posed by the variability and linguistic nuances in histopathology texts better.

References

A.R. Aronson and F.-M. Lang. 2010. An overview of metamap: historical perspective and recent advances. *J. Am. Med. Inform. Assoc.*

J.P. Chiu and E. Nichols. 2016. Named entity recognition with bidirectional lstm-cnns. *Transactions of the association.*

C. Even, S. Bastuji-Garin, and Y. Hicheri. 2011. Impact of invasive fungal disease on the chemotherapy schedule and event-free survival in acute leukemia patients who survived fungal disease: a case-control study. *Haematologica*.

Christopher Funk, William Baumgartner, Benjamin Garcia, Christophe Roeder, Michael Bada, K Bretonnel Cohen, Lawrence E Hunter, and Karin Verspoor. 2014. Large-scale biomedical concept recognition: an evaluation of current automatic annotators and their parameters. *BMC Bioinformatics*, 15:1–29.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of the 18th International Conference on Machine Learning*, pages 282–289.

G. Lample, M. Ballesteros, and S. Subramanian. 2016. Neural architectures for named entity recognition. *Proceedings of the HLT-NAACL*.

Robert Leaman and Graciela Gonzalez. 2008. Banner: an executable survey of advances in biomedical named entity recognition. In *Pacific Symposium on Biocomputing 2008*, pages 652–663. World Scientific.

Vlada Rozova, Anna Khanina, Jasmine C. Teng, Joanne S.K. Teh, Leon J. Worth, Monica A. Slavin, Karin A. Thursky, and Karin Verspoor. 2023a. Chifir: Cytology and histopathology invasive fungal infection reports (version 1.0.0). *PhysioNet*.

Vlada Rozova, Anna Khanina, Jasmine C. Teng, Joanne S.K. Teh, Leon J. Worth, Monica A. Slavin, Karin A. Thursky, and Karin Verspoor. 2023b. Detecting evidence of invasive fungal infections in cytology and histopathology reports enriched with concept-level annotations. *Journal of Biomedical Informatics*, 139:104293.

C.N. Santos, V. Guimaraes, and R.J. Niteroi. 2015. Boosting named entity recognition with neural character embeddings. *Proceedings of NEWS2015 The Fifth Named Entities Workshop*.

Fei Sha and Fernando Pereira. 2003. Shallow parsing with conditional random fields. *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*.

Charles Sutton, Andrew McCallum, et al. 2012. An introduction to conditional random fields. *Foundations and Trends® in Machine Learning*, 4(4):267–373.

Michael A Tanenblatt, Anni Coden, and Igor L Sominsky. 2010. The conceptmapper approach to named entity recognition. In *Language Resources and Evaluation Conference*, pages 546–51.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

The uncivil empathy: Investigating the relation between empathy and toxicity in online mental health support forums

Ming-Bin Chen and **Jey Han Lau** and **Lea Frermann**
{mingbin, laujh, lfrermann}@unimelb.edu.au

Abstract

WARNING:This paper contains content related to suicide and self-harm. We explore the relationship between empathy and toxicity in the context of online mental health forums. Despite the common assumption of a negative correlation between these concepts (Lahnala et al., 2022), it has not been empirically examined. We augment the EPITOME mental health empathy dataset (Sharma et al., 2020) with toxicity labels using two widely employed toxic/harmful content detection APIs: Perspective API and OpenAI moderation API. We find a notable presence of toxic/harmful content (17.77%) within empathetic responses, and only a very weak negative correlation between the two variables. Qualitative analysis revealed contributions labeled as empathetic often contain harmful content such as promotion of suicidal ideas. Our results highlight the need for reevaluating empathy independently from toxicity in future research and encourage a reconsideration of empathy's role in natural language generation and evaluation.

1 Introduction

Natural Language Processing (NLP) technology has been instrumental in both the analysis and enhancement of online discussions, as exemplified by its application in platforms like Reddit (Medvedev et al., 2019). Specifically, the detection of toxicity in online comments has emerged as a widely embraced preventive measure for moderating online discussions (Lees et al., 2022). On the other hand, in recent years, there has been a surge of research interest in NLP on empathy (Raamkumar and Yang, 2022), due to its critical role in human communication and relationship building (Muradova, 2021; Sharma et al., 2020).

In the realm of online public discourse analysis, both toxicity and empathy are frequently studied and discussed within the broader context of civility (Friess and Eilders, 2015). While toxicity is typically characterized as a form of uncivil behavior, empathy is associated with civil interactions that contribute to pro-social outcomes. Some research in the field of NLP has made unexamined implicit assumptions based on this conceptual contrast. One such assumption posits a negative correlation between empathy and toxicity (Lahnala et al., 2022; Oswald, 2023). On the other hand, studies from psychology hold mixed views regarding the relation between the two concepts (Moyers and Miller, 2013; Breithaupt, 2018). While the effect and roles of both toxicity and empathy are complex, developing technology founded on unexamined assumptions entails the risk of unforeseen consequences.

This study analyses the correlation between toxicity and empathy using the human annotated empathy labels of EPITOME, a widely used mental health subreddit empathetic dataset (Sharma et al., 2020), and augmenting it with toxicity labels predicted by two popular APIs. We conduct a qualitative analysis of EPITOME responses which are both empathetic and predicted as toxic. Our key findings and contributions are:

1. 17.77% of human-identified empathetic responses classified as toxic/harmful by APIs.[1]
2. Contrary to intuition, no strong negative correlation found between API predicted toxic/harmful labels and human annotated EPITOME empathetic labels.
3. Qualitative analysis reveals presence of suicidal ideation and the widespread unhelpful responses, suggesting potential risks in fine-tuning empathetic language generation with EPITOME dataset.

2 Related Work

Toxicity is generally defined as language that is harmful, offensive, or suppressing the expression of others (van Aken et al., 2018). While earlier tox-

---

[1] We validated the quality of predictions in Appendix C.

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 136–147
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

icity detection tasks primarily focused on binary classification (Dixon et al., 2018), more recent studies have shifted towards incorporating more specific fine-grained labels, such as personal attacks (Wulczyn et al., 2017), hate speech (Hartvigsen et al., 2022) and many more (Price et al., 2020). Recent developments also encompass toxic span detection (Pavlopoulos et al., 2021) and implicit, context-dependent toxicity detection (Hartvigsen et al., 2022; Anuchitanukul et al., 2022).

Some of these advancements have transitioned into production as public APIs, such as the Perspective API (Jigsaw, 2023), and find practical use not only in everyday applications like online forum moderation but also in research fields beyond computer science, such as political science. However, some concerns have been raised regarding the potential inconsistency and oversimplification in the underlying definitions of toxicity within the detection models (Fortuna et al., 2020).

Driven by the interest in developing more engaging and supportive AI agents, empathy has emerged as a prominent theme in recent NLP research (Raamkumar and Yang, 2022). Earlier research on empathy primarily focused on emotional understanding and reactions, whereas recent works delve into the cognitive dimensions of empathy, including perspective-taking (Kim et al., 2021). While numerous studies aim to generate empathetic responses resembling human ones, few concentrate on automated empathy detection. This trend can be attributed, in part, to empathy's diverse definitions, spanning various fields such as cognitive neuroscience and psychology (Singer and Lamm, 2009; Cuff et al., 2016). The EPITOME dataset (Sharma et al., 2020) stands out as the sole dataset to not only label empathy levels but also annotate empathy across three distinct components: Emotion Reaction (ER), Interpretation (IP), and Exploration (EX), encompassing both emotional and cognitive aspects of empathy.

3 Methodology

In this study, we use the sub-reddit version of the EPITOME dataset, which was sourced from 55 mental health focused subreddits. The dataset includes 3081 pairs of support seeker post and peer support response. Each response message is human annotated with the levels (None: 0, Weak: 1, Strong: 2) of the three empathetic components (ER, IP, EX). Appendix A covers the detailed definitions and annotation level criteria.

We use two widely-used APIs for harmful and toxic online content detection, the Perspective API (Jigsaw, 2023) and OpenAI's moderation API (OpenAI, 2023). Perspective API is provided by Google for online content moderation. The underlying models of the API are trained on online comment labels from a variety of sources, like Wikipedia. Given an input message, the API returns continuous scores (0-1) for 6 different toxicity categories. Besides the score, the API also returns the detected toxic spans for each corresponding category.

OpenAI's moderation API was developed primarily for moderating the input and output of their flagship large language model ChatGPT. With less emphasis on toxicity per se, the API is designed to detect harmful and dangerous content. For each input message, it returns an overall binary flag (0,1) and 11 continuous category scores (0-1). Appendix B contains the detailed definitions for the labels of both APIs.

Using both APIs, we (automatically) annotate the *peer support responses* in the EPITOME datasets with toxicity labels. We are primarily interested in empathy/toxicity in EPITOME peer responses, and so feed only the responses into the APIs. To clarify, we do not include the support seeker post as part of the input, and so the classification is done using only the response. Both quantitative and qualitative analysis have been conducted based on the scores from the APIs along with the human annotated EPITOME labels.

To validate the predictions of the two APIs and ensure the validity of this study, we conducted manual annotation on 50 positive (labeled as toxic by at least one API) and 50 negative samples (not labeled as toxic by either API), resulting 0.87 accuracy (details in Appendix C), suggesting that the toxicity predictions are generally reliable. To provide a qualitative understanding on these predictions, we conducted an error analysis and identified that the predominant error cases (12 out of 13) were false positive errors. These errors were largely attributed to the predicted self-harm labels from the OpenAI's moderation API (10 out of the 12), while a smaller subset were related to profanity use (2 out of 12). Upon further qualitative analysis, we identified that the error cases frequently featured lengthy content with mixed intentions. For instance, these cases often began with individuals sharing their own suicidal thoughts or experiences

Label	Frequency	Percentage
IP(E)	1458	47.32
ER(E)	1047	33.98
EX(E)	480	15.58
Profanity(P)	315	10.22
Toxicity(P)	294	9.54
Self-harm(O)	133	4.32
Self-harm/intent(O)	124	4.02
Insult(P)	61	1.98
Harassment(O)	45	1.46
Threat(P)	35	1.14
Violence(O)	25	0.81

Table 1: Frequency and % contining posts of labels from the three label groups with frequency > 20.

but subsequently shifted towards discouraging suicide. We also observed that the APIs can at times exhibit oversensitivity to the presence of specific keywords like “suicide”, “depression” and “shit”, even when these terms are used with the goal of emphasis rather than offence.

4 Quantitative Analysis

We first analyse the frequency distribution of each toxic/harmful label and its correlation with the EPITOME component levels. In addition, we inspect the difference between the empathetic toxic text and the non-empathetic toxic text. Hereafter, for convenience, we will group the labels into three label groups based on their sources, which are E (EPITOME), P (Perspective API), and O (OpenAI moderation API).

4.1 Frequency Analysis

Table 1 presents the label-level frequency distribution across the three label groups. To calculate these frequencies, we converted the continuous confidence scores within the P and O label groups into binary values, considering any score greater than 0.5 as positive. For labels within group E, we marked both weak (1) and strong level (2) as presence.

Within the Perspective API labels, “profanity” and “toxicity” are the two most frequently occurring labels. Conversely, the OpenAI moderation API primarily identifies “self-harm” and “self-harm/intent” as the most frequent labels. Furthermore, the rarer, more severe forms of toxic or harmful speech labels, such as “hate”, “severe toxicity” and “identity attack”, exhibit frequencies below 20 ($<< 1\%$ of instances) and as such are excluded from our experiments.

Label groups	Count
E(EPITOME)	2381
P (Perspective API)	379
O (OpenAI moderation)	248
$E \cap P$	288
$E \cap O$	203
$E \cap P \cap O$	68
$E \cap (P \cup O)$	423
Total	3081

Table 2: The frequency and intersection frequency of the three labels groups.

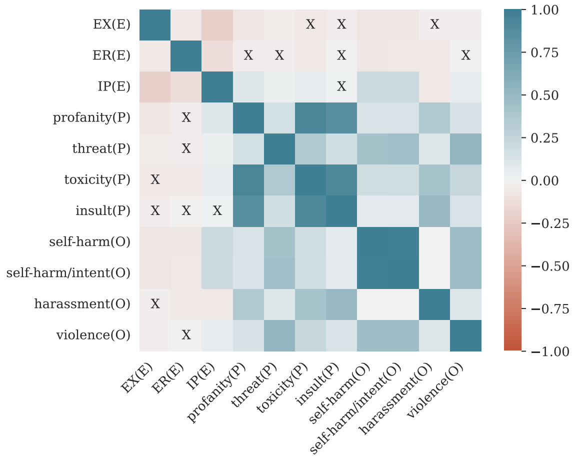


Figure 1: The Pearson correlation between labels from the three label groups. "X" indicates insignificant correlation (p-value > 0.05).

Table 2 displays the group-level frequencies for each group individually as well as their intersection. We consider a group label as present if at least one label within the group was labeled as positive. Overall, we observe a notable presence of toxic or harmful labels within empathetic instances.

4.2 Correlation Analysis

Figure 1 illustrates the Pearson correlation among the labels from the three distinct groups. Based on prior work, we would expect (a) positive correlations between labels from the two toxicity APIs and (b) negative correlations between toxicity and empathy labels. Indeed, we note weak to moderate levels of positive correlations observed between the labels from the two APIs (P and O groups). However, the correlations between empathy (E) and

Type	N	% Toxic	Length
All	3084	8.07	47.01
$E \cap (P \cup O)$	423	50.86	58.45
$\neg E \cap (P \cup O)$	109	70.18	34.69

Table 3: Comparing frequency (N), toxic coverage (% toxic), and response length in tokens between empathetic and non-empathetic toxic harmful responses.

Label	Empathetic	Toxic	Helpful
Percentage	88%	74%	26%

Table 4: Summary statistics of the quality analysis with manual annotation.

toxicity (P, O) labels exhibit a mixed pattern, comprising both insignificant correlations (indicated with an 'x' in Figure 1) as well as small but significant positive and negative correlations. While EX and ER show some significant but weak negative correlation with some toxic/harmful labels, IP has weak positive correlation with most of the toxic/harmful labels. Overall, this mixed pattern does not fully align with the common assumption of negative correlation — or, in other words, that the presence of empathy suggests a lack of toxicity and vice versa.

4.3 Toxicity in Empathetic and Non-empathetic Responses

To explore the factors contributing to the toxicity or harm in empathetic responses, we compared between empathetic toxic/harmful ($E \cap (P \cup O)$) and non-empathetic toxic/harmful ($\neg E \cap (P \cup O)$) responses. We used the Perspective API to identify toxic spans and estimate the fraction of toxic language in a response. The results, as shown in Table 3, indicate that empathetic toxic/harmful responses exhibit substantially lower fractions of toxic language, and are generally longer compared to their non-empathetic counterparts.

5 Qualitative Analysis

To better understand the interplay between empathy and toxic/harmful characteristics, we selected a subset of the top 50 samples that exhibited high levels of empathy while also being associated with either of the toxic/harmful group labels ($E \cap (P \cup O)$), and performed another manual annotation. Here we collapsed the fine-grained labels of EPITOME and both APIs categories into two binary labels "empathetic" and "toxic", and included a third class, "helpful" (also binary), to evaluate whether the responses has pragmatic benefit to the seekers. We define "helpful" as *comments or content that have the intention or potential to help/improve the future situation or lessen the negativity of the seeker physically, mentally or emotionally.* Full definitions of all three classes are given in appendix D. The motivation for introducing the "helpful" class is to fill the gap in the current EPINOME annotations, which lack a metric for measuring the desired outcome or utility. In the context of mental health support, we propose the perception of "helpfulness" serves as a proxy for the desired outcome. For this exercise, the first author of this paper annotated all 50 samples.

Table 4 displays the distribution of the three classes in the 50 samples. We see high levels of "empathetic" and "toxic" instances, aligning with the original EPITOME and API annotations (recall that these samples are drawn from $E \cap (P \cup O)$). In contrast, only a smaller proportion of the responses are categorized as "helpful", suggesting that many responses, although labelled as empathetic, are not ultimately helpful in improving the support seeker's situation.

Table 9 in Appendix D provides examples of responses featuring different label combinations and their ratios. In the first example, the response demonstrates an intention to help and convey understanding and uses of profanity for emphasis. In contrast, the second to fourth examples illustrate various instances where both toxicity and empathy are present but there is a lack of any intent to help the seeker. We also see patterns of side-taking and personal tragedy sharing. Notably, the third example contains content indicative of suicidal ideation (despite being emphathetic). Our qualitative analysis also reveals that the predominant contributor to toxic labels is the use of profanity.

6 Conclusion and Limitation

We examined the interplay of empathy and toxicity in responses to support seekers in mental health online discussions.

Our results found a mixed pattern of insignificant or weak (positive/negative) correlations between the EPITOME empathy labels and the toxic/harmful labels obtained from two widely used APIs. We also revealed a significant presence of toxic/harmful content within empathetic instances in the EPITOME dataset, dominated by "profanity" and "self-harm" labels. These outcomes challenge the standard assumption that there is a negative correlation between empathetic and toxic/harmful language.

Interestingly, we found that the majority of empathetic toxic/harmful responses are not helpful for the individuals who are seeking help. We also noticed some well-intent responses being labelled as toxic due to use of profanity. These mislabels could stem from the issues of oversimplification and ambiguity in toxicity definitions, as previously highlighted in relevant studies (Fortuna et al., 2020). As argued by some communication studies (Masullo Chen et al., 2019) (and also seen in our analyses), the utilization of toxic language does not invariably signify malicious intent. Instead, it may function as a tool for emphasis, conveying closeness, or aligning with the conventions of a particular sociolect, or online context. This observation raises further questions about the role of domain- and community-specific conceptualizations of toxicity in the realm of online content moderation.

Furthermore several empathetic instances are identified as containing suicidal ideation. This discovery raises concerns about the potential use of this dataset for empathetic fine-tuning purposes (Lahnala et al., 2022). To address these concerns, we recommend employing fine-grained toxicity detection models or APIs for data filtering along with human manual validation to ensure alignment between the filtered data and the objectives of the fine-tuning task.

We acknowledge a few limitations of this study: First, it only examines a single dataset within the mental health domain, and the predictions do not consider the context of the seeker's post due to API constraints. Second, as demonstrated by both quantitative and qualitative validation of the APIs' performance, the correspondence between the predicted toxic/harmful labels and human judgments is not perfect (though usable given the accuracy). Third, the introduction of the "helpful" label in our analysis is a preliminary endeavor aimed at addressing the absence of a desired outcome metric in EPITOME, and as such is a (gross) simplification of the problem of measuring response utility. More refined measures, like empathic concerns (Zahn-Waxler and Radke-Yarrow, 1990) or self-report surveys, might be worth considering in future studies. And lastly, the final manual annotation (emphathetic, toxic, and helpful) of the responses was done with a single annotator, and more thorough investigation is required to further validate the robustness of our findings.

For future studies, we recommend a re-evaluation and clarification of the role of empathy in text generation and understanding tasks. Given that certain social science studies have indicated potential harm from empathetic behavior (Breithaupt, 2018), further NLP research is needed to identify subcategories of empathy based on context that can either be beneficial or detrimental. Finally, we suggest incorporating a measure of desired or undesired outcomes in future NLP studies, particularly when dealing with complex and sensitive concepts. This approach will facilitate the analysis and validation of the interplay between outcomes and mediating factors, such as empathy.

References

Atijit Anuchitanukul, Julia Ive, and Lucia Specia. 2022. Revisiting contextual toxicity detection in conversations. *ACM Journal of Data and Information Quality*, 15(1):1–22.

Fritz Breithaupt. 2018. The bad things we do because of empathy. *Interdisciplinary Science Reviews*, 43(2):166–174.

Benjamin MP Cuff, Sarah J Brown, Laura Taylor, and Douglas J Howat. 2016. Empathy: A review of the concept. *Emotion review*, 8(2):144–153.

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2018. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 67–73.

Paula Fortuna, Juan Soler, and Leo Wanner. 2020. Toxic, hateful, offensive or abusive? what are we really classifying? an empirical analysis of hate speech datasets. In *Proceedings of the 12th language resources and evaluation conference*, pages 6786–6794.

Dennis Friess and Christiane Eilders. 2015. A systematic review of online deliberation research. *Policy & Internet*, 7(3):319–339.

Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. 2022. Toxigen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3309–3326.

Jigsaw. 2023. Perspective api. `https://developers.perspectiveapi.com`. Accessed: 2023-09-03.

Hyunwoo Kim, Byeongchang Kim, and Gunhee Kim. 2021. Perspective-taking and pragmatics for generating empathetic responses focused on emotion causes. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2227–2240.

Allison Lahnala, Charles Welch, Béla Neuendorf, and Lucie Flek. 2022. Mitigating toxic degeneration with empathetic data: Exploring the relationship between toxicity and empathy. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4926–4938.

Alyssa Lees, Vinh Q Tran, Yi Tay, Jeffrey Sorensen, Jai Gupta, Donald Metzler, and Lucy Vasserman. 2022. A new generation of perspective api: Efficient multilingual character-level transformers. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3197–3207.

Gina Masullo Chen, Ashley Muddiman, Tamar Wilner, Eli Pariser, and Natalie Jomini Stroud. 2019. We should not get rid of incivility online. *Social Media+ Society*, 5(3):2056305119862641.

Alexey N Medvedev, Renaud Lambiotte, and Jean-Charles Delvenne. 2019. The anatomy of reddit: An overview of academic research. *Dynamics On and Of Complex Networks III: Machine Learning and Statistical Physics Approaches 10*, pages 183–204.

Theresa B Moyers and William R Miller. 2013. Is low therapist empathy toxic? *Psychology of Addictive Behaviors*, 27(3):878.

Lala Muradova. 2021. Seeing the other side? perspective-taking and reflective political judgements in interpersonal deliberation. *Political Studies*, 69(3):644–664.

OpenAI. 2023. Openai moderation api. `https://platform.openai.com/docs/guides/moderation`. Accessed: 2023-09-03.

Lisa Oswald. 2023. Effects of preemptive empathy interventions on reply toxicity among highly active social media users.

John Pavlopoulos, Jeffrey Sorensen, Léo Laugier, and Ion Androutsopoulos. 2021. Semeval-2021 task 5: Toxic spans detection. In *Proceedings of the 15th international workshop on semantic evaluation (SemEval-2021)*, pages 59–69.

Ilan Price, Jordan Gifford-Moore, Jory Flemming, Saul Musker, Maayan Roichman, Guillaume Sylvain, Nithum Thain, Lucas Dixon, and Jeffrey Sorensen. 2020. Six attributes of unhealthy conversations. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 114–124.

Aravind Sesagiri Raamkumar and Yinping Yang. 2022. Empathetic conversational systems: A review of current advances, gaps, and opportunities. *IEEE Transactions on Affective Computing*.

Ashish Sharma, Adam Miner, David Atkins, and Tim Althoff. 2020. A computational approach to understanding empathy expressed in text-based mental health support. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5263–5276.

Tania Singer and Claus Lamm. 2009. The social neuroscience of empathy. *Annals of the New York Academy of Sciences*, 1156(1):81–96.

Betty van Aken, Julian Risch, Ralf Krestel, and Alexander Löser. 2018. Challenges for toxic comment classification: An in-depth error analysis. *EMNLP 2018*, page 33.

Ellery Wulczyn, Nithum Thain, and Lucas Dixon. 2017. Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th international conference on world wide web*, pages 1391–1399.

Carolyn Zahn-Waxler and Marian Radke-Yarrow. 1990. The origins of empathic concern. *Motivation and emotion*, 14:107–130.

A Definition and Level Criteria of Epitome Components

Component	Definition	Level Criteria
ER(Emotion Reactions)	Expressing emotions such as warmth, compassion, and concern, experienced by peer supporter after reading seekers post.	A weak communication of emotional reactions alludes to these emotions without the emotions being explicitly labeled (e.g., Everything will be fine). On the other hand, strong communication specifies the experienced emotions (e.g., I feel really sad for you).
IP(Interpretations)	Communicating an understanding of feelings and experiences inferred from the seekers post.	A weak communication of interpretations contains a mention of the understanding (e.g., I understand how you feel) while a strong communication specifies the inferred feeling or experience (e.g., This must be terrifying) or communicates understanding through descriptions of similar experiences (e.g., I also have anxiety attacks at times which makes me really terrified).
EX(Explorations)	Improving understanding of the seeker by exploring the feelings and experiences not stated in the post.	A weak exploration is generic (e.g., What happened?) while a strong exploration is specific and labels the seeker's experiences and feelings which the peer supporter wants to explore (e.g., Are you feeling alone right now?).

Table 5: The definition and level criteria of the EPITOME components

B Definition of Perspective API and OpenAI Moderation API Labels

Label	Source	Definition
Toxicity	Perspective API	A rude, disrespectful, or unreasonable comment that is likely to make people leave a discussion.
Severe toxicity	Perspective API	A very hateful, aggressive, disrespectful comment or otherwise very likely to make a user leave a discussion or give up on sharing their perspective. This attribute is much less sensitive to more mild forms of toxicity, such as comments that include positive uses of curse words.
Identity attack	Perspective API	Negative or hateful comments targeting someone because of their identity.
Insult	Perspective API	Insulting, inflammatory, or negative comment towards a person or a group of people.
Profanity	Perspective API	Swear words, curse words, or other obscene or profane language.
Threat	Perspective API	Describes an intention to inflict pain, injury, or violence against an individual or group.

Table 6: The definition of Perspective API Labels

Label	Source	Definition
Hate	OpenAI	Content that expresses, incites, or promotes hate based on race, gender, ethnicity, religion, nationality, sexual orientation, disability status, or caste. Hateful content aimed at non-protected groups (e.g., chess players) is harrassment.
Hate/Threatening	OpenAI	Hateful content that also includes violence or serious harm towards the targeted group based on race, gender, ethnicity, religion, nationality, sexual orientation, disability status, or caste.
Harassment	OpenAI	Content that expresses, incites, or promotes harassing language towards any target.
Harassment/threatening	OpenAI	Harassment content that also includes violence or serious harm towards any target.
Self-harm	OpenAI	Content that promotes, encourages, or depicts acts of self-harm, such as suicide, cutting, and eating disorders.
Self-harm/intent	OpenAI	Content where the speaker expresses that they are engaging or intend to engage in acts of self-harm, such as suicide, cutting, and eating disorders.
Self-harm/instructions	OpenAI	Content that encourages performing acts of self-harm, such as suicide, cutting, and eating disorders, or that gives instructions or advice on how to commit such acts.
Sexual	OpenAI	Content meant to arouse sexual excitement, such as the description of sexual activity, or that promotes sexual services (excluding sex education and wellness).
Sexual/minors	OpenAI	Sexual content that includes an individual who is under 18 years old.
Violence	OpenAI	Content that depicts death, violence, or physical injury.
Violence/graphic	OpenAI	Content that depicts death, violence, or physical injury in graphic detail.

Table 7: The definition of OpenAI moderation API labels

C Validation of APIs combined performance

To validate the combined performance of the two APIs, we firstly derived an overall toxic flag which is labeled as positive if any APIs returned labels(6 from Perspective API and 11 from Open AI moderation API) is positive. For the manual annotation criteria, we also derived an aggregated definition of overall toxicity by inputting all toxic/harmful labels' definitions into ChatGPT. Subsequently, we employed this unified toxicity criterion to annotate a set of 100 samples, comprising 50 predicted as positive and 50 as negative by the APIs to validate the prediction performance. The outcomes of this validation process are depicted in Figure 2.

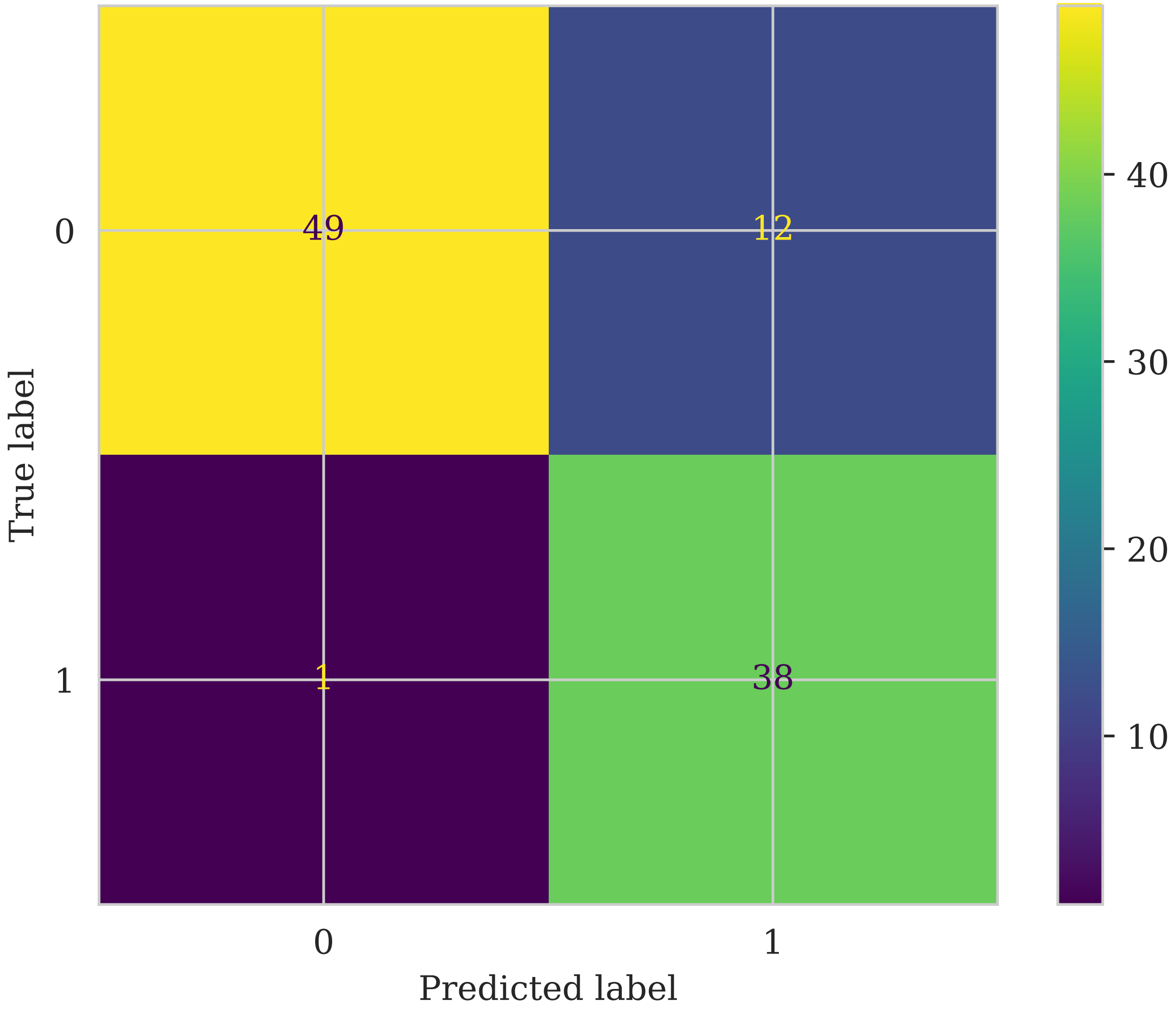


Figure 2: The confusion matrix between the combined predictions of the two APIs(X axis) and the manual annotated binary labels (Y axis).

D Qualitative analysis supplementary

Label	Definition
Empathetic	Comments or content that express emotions like warmth, compassion, and concern in response to seekers' posts, or communicate an understanding of inferred feelings and experiences, or enhance comprehension by exploring unspoken emotions and experiences.
Toxic	Comments or content that are rude, disrespectful, or offensive, likely to drive users away from a discussion, or includes hate speech, harassment, self-harm promotion, sexual content, and graphic depictions of violence or injury, targeting various identity-based and non-protected groups.
Helpful	Comments or content that have the intention or potential to help/improve the future situation or lessen the negativity of the seeker physically, mentally or emotionally.

Table 8: The definition of the labels for the qualitative analysis annotation.

Example seeker post	Example response	Empathetic	Toxic	Helpful	Ratio(%)
I want to die. I fucking hate myself and my miserable existence. I'm a shitty excuse of a human being. I can't even kill myself properly but that probably won't stop me from trying again soon.	What makes you think you're a shitty human being? I don't know who you are and I don't know your story, so pardon me if I'm coming as rude by saying this (that is not my intention, I'm solely trying to help), but maybe some of those reasons aren't founded. For example I often worry that all the people I love will abandon me, and even though it's not really founded and I know in the back of my head that it probably won't happen, it is still one of my largest fears.	1	1	1	16
I almost threw up with anxiety from walking down the high street. I used to be able to rave, go out with friends, work, study. Now I can't walk down the road without having panic attacks. Great	sorry man. If people judge you for it then FUCK THEM	1	1	0	52
I just feel like shit.. I don't really know why. Everything's just so overwhelming all of sudden. I was feeling great this past month, but it's back. Did I do something wrong? I don't know why.	I feel like shit too everyday. I just wanna fade away.	1	1	0	52
What the hell am I doing. I'm failing classes. I'm sabotaging relationships with people I care about. I have little to no motivation for social interaction. Literally what the fuck am I doing. I don't think I've reached a lower point than what I'm currently at.	we on the same boat. havent been going to school, probably already failed some of my classes. theyre just a group of fucking morons anyway. broke up with gf. have no friends. the ones I had were fake. shit.	1	1	0	52
DAE have this? Any happy moment is ruined by this thought: This would be a good moment to be my last.. Every happy experience is accompanied by this. Makes it impossible to enjoy things or get past my issues.	I have tried to kill myself **while happy** for that exact reason.	0	1	0	6

Table 9: Examplar post and response pairs with manual annotation using three labels, empathetic, toxic, and helpful. The last column "ratio" indicates the percentage of samples with the same label combination.

Overview of the 2023 ALTA Shared Task: Discriminate between Human-Written and Machine-Generated Text

Diego Molla
Macquarie University
`diego.molla-aliod@mq.edu.au`

Haolan Zhan
Monash University
`haolan.zhan@monash.edu`

Xuanli He
University College London
`xuanli.he@ucl.ac.uk`

Qiongkai Xu
The University of Melbourne
`qiongkai.xu@unimelb.edu.au`

Abstract

The ALTA shared tasks have been running annually since 2010. In 2023, the purpose of the task is to build automatic detection systems that can discriminate between human-written and synthetic text generated by Large Language Models (LLM). In this paper we present the task, the evaluation criteria, and the results of the systems participating in the shared task.

1 Introduction

The generative abilities of recent Large Language Models (LLMs) such as ChatGPT have shown impressive abilities in generating content with quality close to those generated by humans. Despite the possible advantages of LLMs, the concern about inappropriate utilization of these generated contents, accompanied by social and ethical issues, has been underscored in several preceding studies (Zellers et al., 2019; Aliman and Kester, 2021; Ranade et al., 2021; Xu et al., 2022).

Some of those LLMs are designed with watermarks (He et al., 2022; Kirchenbauer et al., 2023). However, there is also the possibility of deploying LLMs without watermarks. Consequently, effectively distinguishing texts by vanilla language models from the human-written text pieces has become an emerging and challenging task.

The goal of the 2023 ALTA shared task is to build automatic detection systems that can discriminate between human-written and text generated by LLMs. The text comes from a variety of sources and different LLMs.

Formally, this is a binary classification problem, as each candidate sentence can be generated either by human or a LLM. The evaluation metric is accuracy.

Section 2 presents related work. Section 3 details how the data have been gathered and labeled. Section 4 presents the evaluation framework. Section 5 describes a baseline that was made available to the participants. Section 6 lists the details of the participating systems and their results. Finally, Section 7 concludes this paper.

2 Related Work

The preliminary work for identifying machine-generated text involves feature-based approaches, such as utilizing linguistic patterns (Muñoz-Ortiz et al., 2023) and cues (Solaiman et al., 2019), e.g., bag-of-words. More recent work (Zellers et al., 2019) proposes to use detectors based on pre-trained language models. e.g., Liu et al. (2019) use RoBERTa as the basis of the detector. After a fine-tuning process, RoBERTa has been proven its prowess as a detector across multiple domains (Solaiman et al., 2019; Fagni et al., 2021; Rodriguez et al., 2022). To align with our research goals, we depart from the conventional assumption that detailed knowledge of synthetic data origin is readily available, which includes specifics about generative models, decoding strategies, and domains. In reality, such information often remains elusive.

It is worth noting several recent works on discriminating human- and machine-generated texts, e.g., OpenAI GPT-2 Detector (OpenAI, 2023), GPTZero (Tian and Cui, 2023), DetectGPT (Mitchell et al., 2023), DIPPER (Krishna et al., 2023) and G3-Detector (Zhan et al., 2023), which train their detectors on collected datasets with labeled human-written and machine-generated texts. Later on, a training-free detector DNA-GPT (Yang et al., 2023) was proposed to discover n-gram patterns in the machine-generated text.

Although some progress has been made in the corresponding task, its efficacy and reliability largely depend on the task settings, such as the domains of the generative tasks, the structures and scale of the generative models, etc. (Sadasivan et al., 2023) Kumarage et al. (2023) propose an assessment framework using evasive soft prompts,

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 148–152
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

and Chakraborty et al. (2023) further introduce AI detectability index as an evaluation metric for machine-generated text detection.

Related shared tasks include CLIN33[1], AuTextification[2] (Sarvazyan et al., 2023) , Detecting Generated Scientific Papers[3] (robodasha, 2022), and Machine Learning Model Attribution Challenge[4] (Merkhofer et al., 2023).

3 Data Gathering

The data for the 2023 ALTA shared task has been gathered from four generative benchmarks across multiple domains in the data. These comprise machine translation, and specifically the WMT (De-En) benchmark (Bojar et al., 2014), summarization, with CNN-DailyMail (CNNDM) (Nallapati et al., 2016), and language pre-training, including WikiData and the OpenwebText benchmark (Radford et al., 2019).

The human-written text are directly extracted from the ground-truth sentences in the above benchmarks. In contrast, the machine-generated text are produced by several widely-used generative models, all of which are GPT-based models. Specifically, these models contain GPT2-large, GPT3.5-turbo, and GPT4. We have used GPT2 model files through the Huggingface repository [5], and then fine-tuned these models on the aforementioned datasets. For the GPT3.5-turbo and GPT4 models, we use prompt-based text generation through the OpenAI API [6]. Specifically, we use the following prompts for different generative benchmarks:

Translation: `Please translate the following German sentence into English.`

Summarization: `Please summarize the following long paragraph with a short summary.`

Language Pre-training: `Please paraphrase the following sentence.`

The final data used in the 2023 ALTA shared task was selected by random sampling from the gathered data to ensure 50%-50% between human and machine-generated text (Table 1).

Partition	Human (0)	Machine (1)	Total
Training	9,000	9,000	18,000
Development	1,000	1,000	2,000
Test	1,000	1,000	2,000

Table 1: Statistics of the data used in the 2023 ALTA shared task

4 Evaluation Framework

The evaluation framework was implemented as a CodaLab competition[7] with three phases.

In the **development phase**, labelled training and unlabelled development sets were made available. Participant systems could submit their system output on the development set up to 100 times, and the evaluation results were made public to all participating systems via a leaderboard.

In the **test phase**, an additional unlabelled test set was made available, and participating systems could make up to 3 submissions. The results of the test phase form a separate leaderboard and are used for the final ranking reported in this paper.

A third **unofficial submissions** phase has no end date and is available to all participant systems so that they can make additional submissions on the test data. These submissions form a separate leaderboard and are not used for the final ranking.

Table 1 shows the statistics of the three partitions.

5 Baseline

We formulate the detection framework as a binary classification task. Based on previous observations (Fagni et al., 2021; Rodriguez et al., 2022), RoBERTa has proven successful in various detection tasks. Therefore, to provide a starting point for participants, we provide the vanilla `RoBERTa-large` (Liu et al., 2019) as a baseline system[8]. Specifically, we use the corresponding checkpoint presented in Huggingface[9], which contains 354 million parameters. The performance of `RoBERTa-large` on the test set is 0.9765 in terms of accuracy.

---

[1] https://sites.google.com/view/shared-task-clin33/home

[2] https://sites.google.com/view/autextification/home

[3] https://www.kaggle.com/competitions/detecting-generated-scientific-papers

[4] https://mlmac.io/

[5] https://huggingface.co/

[6] https://chat.openai.com/

[7] https://codalab.lisn.upsaclay.fr/competitions/14327

[8] https://github.com/zhanhl316/ALTA2023_shared_task

[9] https://huggingface.co/roberta-large

System	Category	Accuracy
OD-21	Student	**0.9910**
DetectorBuilder	Student	0.9845
AAST-NLP	Student	0.9835
SamNLP	Student	0.9820
Baseline		*0.9765*
VDetect	Student	0.9715
cantnlp	Student	0.9675
ScaLER	Student	0.9665
SynthDetectives	Student	0.9555

Table 2: Results of the 2023 ALTA shared task

6 Participating Systems and Results

A total of 9 teams submitted runs in the development phase, and 8 submitted in the test phase[10]. Table 2 shows the results of the baseline and the participating systems for the text phase.

The ALTA shared tasks have two categories, a student category where student members are not allowed to have completed a PhD degree and cannot be employed full time (with the exception of student supervisors), and an open category for those who are not eligible for the student category. However, this year (2023) only teams in the student category submitted in the test phase.

Tests of statistical significance[11] indicate that the difference between the first and the second team is statistically significant.

All of the participating systems that submitted a system description to us reported to have used LLMs in different ways, often as part of ensemble approaches, sometimes in addition to other approaches.

Team OD-21 (Gagiano and Tian, 2023) used Falcon-7B and label smoothing. They also used prompting techniques for samples with low confidence scores.

Team DetectorBuilder (Fang, 2023) used an ensemble with majority voting of BERT, RoBERTa, and DeBERTaV3.

Team AAST NLP (El-Sayed and Nasr, 2023) used an ensemble with majority voting of DistillBERT, XLMRoBERTa, and RoBERTa.

Team SamNLP (Joy and Aishi, 2023) used a feature-level ensemble of DeBERTaV3 and XLM-RoBERTa, where these LLMs are jointly trained by concatenating their last layer and adding subsequent lineal layers.

Team VDetect (Liyanage and Buscaldi, 2023) experimented with various ensemble approaches using a varied range of models including several Transformer models, RNNs, and CNN, plus SVM and Naive Bayes.

Team SynthDetectives (Nguyen et al., 2023) used an ensemble of ALBERT, ELECTRA, RoBERTa, and XLNet, where the predictions of these LLMs are fed to a linear regression classifier.

7 Conclusions

The 2023 ALTA shared task focused on the discrimination between human-written text and machine-generated text. All systems submitting runs to the test phase had accuracy results over 0.95, and the baseline based on RoBERTa had an accuracy result of 0.9765. The top system submitted to the shared task had an accuracy of 0.9910, yet the difference with the second best system was statistically significant.

We were pleased to observe such good performance by the participants. This indicates that the task of identifying machine-generated text can be easy when used as a shared task like the one presented here. This task may become more difficult in the future as technology evolves.

References

Nadisha-Marie Aliman and Leon Kester. 2021. Epistemic defenses against scientific and empirical adversarial ai attacks. In *CEUR Workshop Proceedings, 2021 Workshop on Artificial Intelligence Safety*.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Aleš Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58, Baltimore, Maryland, USA. Association for Computational Linguistics.

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 CTˆ2: AI-generated

---

[10] Not all teams who submitted in the test phase had submitted in the development phase

[11] We conducted both McNemar's and Bootstrap tests using `https://github.com/rtmdrr/testSignificanceNLP`

text detection is not as easy as you may think–introducing ai detectability index. *arXiv preprint arXiv:2310.05030*.

Ahmed El-Sayed and Omar Nasr. 2023. An ensemble based approach to detecting synthetic data generated by large language models. In *Proceedings of ALTA 2023, shared task section*.

Tiziano Fagni, Fabrizio Falchi, Margherita Gambini, Antonio Martella, and Maurizio Tesconi. 2021. Tweepfake: About detecting deepfake tweets. *Plos one*, 16(5):e0251415.

Yunhao Fang. 2023. Automatic detection of machine-generated text using pre-trained language models. In *Proceedings of ALTA 2023, shared task section*.

Rinaldo Gagiano and Lin Tian. 2023. A prompt in the right direction: Prompt based classification of machine-generated text detection. In *Proceedings of ALTA 2023, shared task section*.

Xuanli He, Qiongkai Xu, Lingjuan Lyu, Fangzhao Wu, and Chenguang Wang. 2022. Protecting intellectual property of language generation apis with lexical watermark. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10758–10766.

Saman Sarker Joy and Tanusree Das Aishi. 2023. Feature-level ensemble learning for robust synthetic text detection with DeBERTaV3 and XLM-RoBERTa. In *Proceedings of ALTA 2023, shared task section*.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models. *arXiv preprint arXiv:2301.10226*.

Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. 2023. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. *arXiv preprint arXiv:2303.13408*.

Tharindu Kumarage, Paras Sheth, Raha Moraffah, Joshua Garland, and Huan Liu. 2023. How reliable are ai-generated-text detectors? an assessment framework using evasive soft prompts. *arXiv preprint arXiv:2310.05095*.

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*.

Vijini Liyanage and Davide Buscaldi. 2023. An ensemble method based on the combination of transformers with convolutional neural networks to detect artificially generated text. In *Proceedings of ALTA 2023*.

Elizabeth Merkhofer, Deepesh Chaudhari, Hyrum S. Anderson, Keith Manville, Lily Wong, and João Gante. 2023. Machine learning model attribution challenge.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. DetectGPT: Zero-shot machine-generated text detection using probability curvature. *arXiv preprint arXiv:2301.11305*.

Alberto Muñoz-Ortiz, Carlos Gómez-Rodríguez, and David Vilares. 2023. Contrasting linguistic patterns in human and llm-generated text. *arXiv preprint arXiv:2308.09067*.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Duke Nguyen, Khaing Myat Noe Naing, and Aditya Joshi. 2023. Stacking the odds: Transformer-based ensemble for AI-generated text detection. In *Proceedings of ALTA 2023, shared task section*.

OpenAI. 2023. AI text classifier.

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.

Priyanka Ranade, Aritran Piplai, Sudip Mittal, Anupam Joshi, and Tim Finin. 2021. Generating fake cyber threat intelligence using transformer-based models. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–9.

Yury Kashnitsky robodasha. 2022. Detecting generated scientific papers.

Juan Rodriguez, Todd Hay, David Gros, Zain Shamsi, and Ravi Srinivasan. 2022. Cross-domain detection of GPT-2-generated technical text. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1213–1233, Seattle, United States. Association for Computational Linguistics.

Vinu Sankar Sadasivan, Aounon Kumar, Sriram Balasubramanian, Wenxiao Wang, and Soheil Feizi. 2023. Can AI-generated text be reliably detected? *arXiv preprint arXiv:2303.11156*.

Areg Mikael Sarvazyan, José Ángel González, Marc Franco-Salvador, Francisco Rangel, Berta Chulvi, and Paolo Rosso. 2023. Overview of AuTexTification at IberLEF 2023: Detection and attribution of machine-generated text in multiple domains. *Procesamiento del Lenguaje Natural*, 71(0):275–288.

Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, and Jasmine Wang. 2019. Release strategies and the social impacts of language models. *CoRR*, abs/1908.09203.

Edward Tian and Alexander Cui. 2023. GPTZero: Towards detection of ai-generated text using zero-shot and supervised methods.

Qiongkai Xu, Xuanli He, Lingjuan Lyu, Lizhen Qu, and Gholamreza Haffari. 2022. Student surpasses teacher: Imitation attack for black-box NLP APIs. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2849–2860.

Xianjun Yang, Wei Cheng, Linda Petzold, William Yang Wang, and Haifeng Chen. 2023. DNA-GPT: Divergent n-gram analysis for training-free detection of GPT-generated text. *arXiv preprint arXiv:2305.17359*.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. In *Advances in Neural Information Processing Systems 32*.

Haolan Zhan, Xuanli He, Qiongkai Xu, Yuxiang Wu, and Pontus Stenetorp. 2023. G3Detector: General GPT-generated text detector. *arXiv preprint arXiv:2305.12680*.

A Prompt in the Right Direction: Prompt Based Classification of Machine-Generated Text Detection

Rinaldo Gagiano and **Lin Tian**
School of Computing Technologies, RMIT University, Melbourne, Australia
`{rinaldo.gagiano, lin.tian2}@student.rmit.edu.au`

Abstract

The goal of ALTA 2023 Shared Task is to distinguish between human-authored text and synthetic text generated by Large Language Models (LLMs). Given the growing societal concerns surrounding LLMs, this task addresses the urgent need for robust text verification strategies. In this paper, we describe our method, a fine-tuned Falcon-7B model with incorporated label smoothing into the training process. We applied model prompting to samples with lower confidence scores to enhance prediction accuracy. Our model achieved a statistically significant accuracy of 0.991.

1 Introduction

The rapid evolution of Large Language Models (LLMs) has significantly facilitated the generation of complex, human-like text at scale (OpenAI, 2023). These LLMs have found applications in various domains, including AI-assisted writing (Coenen et al., 2021), medical question answering (Yang et al., 2022; Haq et al., 2021, 2022), financial (Lumley, 2023; Haas, 2023; Delocski, 2023), and legal sectors (Trautmann et al., 2022; Blair-Stanek et al., 2023). Leading models like OpenAI's GPT-3 (Brown et al., 2020), Meta's OPT (Zhang et al., 2022), and Big Science's BLOOM (Scao et al., 2022) have the ability to produce content that closely mimics human-created text, making it challenging to distinguish between machine-generated and human-generated content. However, it's important to note that these models lack a genuine understanding of the content they generate.

This limitation can lead to intended negative consequences when this machine-generated content is used in downstream applications. For instance, LLMs have been used to carry out academic fraud (Cotton et al., 2023; Wahle et al., 2022; Elali and Rachid, 2023), disseminate fabricated news stories (Bagdasaryan and Shmatikov, 2022; Groll, 2023; Zellers et al., 2019), and manipulate public opinion (Goldschmidt, 2019; Stella et al., 2018; Bessi and Ferrara, 2016). Given the widespread use of LLMs by the general public (Gault, 2023) and the rapid global dissemination of information, there is a growing risk of disinformation affecting both individuals and organisations.

To address these issues, it is crucial to differentiate between content authored by LLMs and humans. This distinction is essential for ensuring that machine-generated content is used appropriately in various applications while maintaining oversight. Understanding the specific LLM responsible for generating content can help users be aware of potential biases and limitations associated with that model. This interest has led to active research in the area of automatic detection of AI-generated text. Recent work, such as DetectGPT (Mitchell et al., 2023), focuses on techniques for identifying AI-generated content by perturbing text samples and comparing log probabilities. Other approaches involve using LLMs such as DeBERTa (He et al., 2020) or ensemble methods (Przybyła et al., 2023) for multi-class AI detection tasks, illustrating the evolving nature of this research domain.

In this paper, we present our participation in the ALTA 2023 Shared Task (Molla et al., 2023), which centres on the automatic detection of synthetic text produced by LLMs. Participants are challenged with the task of identifying synthetic text across a wide spectrum of sources, spanning different domains and LLMs, including prominent models like T5 (Raffel et al., 2020) and GPT-X (Black et al., 2022). The primary assessment criterion is accuracy, and participants are encouraged to explore diverse methodologies and approaches to construct effective text detection systems.

Our approach involved the fine-tuning of a Falcon-7B (Institute, 2023) model, complemented by the integration of label smoothing during the

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 153–158
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

training process. Furthermore, we leveraged prompting techniques (Liu et al., 2023) for samples exhibiting lower confidence scores, to guide our model, resulting in improved predictions and an overall enhanced system accuracy.

Our participation in this shared task yielded a successful outcome, as our method attained an overall accuracy of 0.991. This achievement underscores the effectiveness of our approach in discerning between human-authored and LLM-generated text, making a substantial contribution to the ongoing endeavours aimed at addressing the challenges associated with synthetic text.

2 Related Work

Text classification is a field that extensively investigates the extraction of features from unprocessed text data to predict text categories. This topic has witnessed substantial research efforts over recent decades, leading to the development of various models tailored for this purpose.

Traditional models like Naive Bayes, Logistic Regression, Support Vector Machines, Random Forest, and K-Nearest Neighbors have been widely explored (Shah et al., 2020; Pranckevičius and Marcinkevičius, 2017). Machine learning boosting techniques, including Extreme Gradient Boosting and Adaptive Boosting, have demonstrated their prowess in delivering high performance (Stein et al., 2019; Qi, 2020; Tang et al., 2020; Minastireanu and Mesnita, 2019; Bloehdorn and Hotho, 2006). Deep learning models, such as Convolutional Neural Networks and Recurrent Neural Networks, have surpassed traditional methods in text classification tasks (Yogatama et al., 2017; Bharadwaj and Shao, 2019; Zhou et al., 2016).

In recent years, Transformer-based language models have risen to prominence for natural language processing tasks due to their enhanced parallelization capabilities and self-attention mechanisms (Vaswani et al., 2017), compared to prior models like RNNs (Medsker and Jain, 1999). However, it's crucial to acknowledge that while Transformer models excel in the domains for which they were trained, they can be less adaptable when dealing with out-of-domain or unseen samples. Their profound understanding of specific contexts, stemming from vast pre-training data, makes them experts in those domains, yet can hinder their ability to generalise effectively (Gagiano et al., 2021; Sarvazyan et al., 2023; Wang et al., 2023; Li et al., 2023). The focus on the knowledge they acquire during fine-tuning might result in a degree of "domain bias," making them less suitable for broader applications.

To mitigate the limitations of domain-specificity in Transformer models, a hybrid approach in text classification is increasingly gaining recognition (Przybyła et al., 2023; Abburi et al., 2023). The concept of ensembling Transformer models with traditional approaches, such as Naive Bayes, Support Vector Machines, or Ensemble Learning, can harness the benefits of both worlds (Przybyła et al., 2023; Abburi et al., 2023). The specialised domain knowledge acquired by Transformer models can be combined with the interpretability, simplicity, and robustness offered by traditional techniques, ultimately leading to more versatile and adaptive text classification models.

3 Dataset

3.1 Description

The dataset for the ALTA 2023 shared task on binary classification, aimed at distinguishing between human-generated and machine-generated text in English, is sourced from a diverse array of text origins. While not specifically annotated, sources mentioned in the task description encompass various domains, such as law and medicine, and utilise text generated by a range of large language models, including T5 and GPT-X. The dataset has a balanced distribution of human and machine-generated labels, with 9000 samples each, totalling 18,000 samples altogether.

3.2 Pre-processing

In the pre-processing phase, we derive our validation set from the original training data. To achieve this, we initiate the process by tokenising each sample within the training set. Subsequently, we sort these tokenised samples by their respective lengths. When creating subsets from the original training set, we ensure a balanced representation of sample lengths and origin labels. The resulting data splits comprise 15,000 samples for training and 3,000 for validation. This approach facilitates robust model evaluation and ensures that the dataset adequately represents the variations present in the training data.

4 Methodology

4.1 Proposed Approach

In our approach, we used a multi-step strategy to enhance the performance of our text classification task. First, we fine-tuned the Falcon-7B model with label smoothing regularisation on the training data. We then predict on the validation set, obtaining prediction labels and confidence scores. We extract samples below a chosen confidence threshold and use these to prompt our trained model with a predefined prompt. After prompting we predict on the validation set again, using prediction accuracy to determine the optimal confidence threshold.

4.2 Model

Our approach relied on the *Falcon-7B*[1] built by the Technology Innovation Institute[2]. The model is a causal decoder-only model, trained on 1,5000B tokens from the English dataset *RefinedWeb* (Penedo et al., 2023)

4.3 Label Smoothing

Label smoothing is a common regularisation technique in machine learning, especially in neural network training. Large language models often suffer from overconfidence in prediction tasks. To address this issue, label smoothing introduces a small degree of uncertainty, typically controlled by a small value (epsilon, ϵ), into the ground-truth labels during training. Instead of using 1 for the correct class and 0 for all others in classification, label smoothing assigns slightly lower than 1 to the correct class and slightly higher than 0 to the rest. By encouraging the model to acknowledge alternative possibilities and distribute some probability mass to incorrect classes, label smoothing enhances generalisation, making the model more robust and adaptable to unseen data.

4.4 Prompting

Model prompting is a natural language processing technique that transforms the decision-making process of language models. In traditional classification tasks, models analyse entire text inputs and make predictions based on their understanding of the complete content. However, model prompting introduces a novel approach by providing partial inputs or prompts that guide the model's reasoning towards a specific classification. We use the following prompting structure:

"'*{sample_text}*' this is the wrong classified sample, predicted as *{pred_label}* generated with confidence score *{conf_score}* and the gold prediction is *{true_label}*."

This approach significantly influences the model's thinking, rendering it more focused and contextually attuned to the intended classification task.

5 Experiments

5.1 Implementation Details

The parameters we used for model training, label smoothing, and confidence threshold assessment are as follows:

- The hyper-parameters used for model fine-tuning are shown in Table 1.

Parameter	**Value**
learning_rate	2e-4
fp16	True
max_grad_norm	0.3
max_steps	1000
warmup_ratio	0.03
max_seq_length	512
max_gen_token	1

Table 1: Model fine-tuning hyper-parameters.

- For label smoothing, we set $\epsilon = 0.1$.

- To identify which samples we use for prompting, we search across confidence threshold values of [0.85, 0.92], finding 0.91 optimal.

6 Results

The organisers of the ALTA 2023 shared task provided both a development and a test set for evaluation. While predictions were made on both sets, it's worth noting that the official rankings are determined based on the results from the test set. Accuracy is the metric used to assess the model's performance. For this paper, we exclusively present the results of our test set predictions. The comprehensive leaderboard can be accessed on the ALTA CodaLab Competition website[3].

---

[1] `https://huggingface.co/tiiuae/falcon-7b`

[2] `https://www.tii.ae/`

[3] `https://codalab.lisn.upsaclay.fr/competitions/14327`

Team Name	Accuracy
OD-21	**0.9910**
DetectorBuilder	0.9845
AAST-NLP	0.9835
SamNLP	0.9820
Organizers	0.9765
VDetect	0.9715
cantnlp	0.9675
ScaLER	0.9665
SynthDetectives	0.9555

Table 2: External evaluation of submissions on the test set. Our approach is highlighted in boldface.

Our approach, under the team name *OD-21*, as showcased in Table 2, achieved the highest accuracy score of 0.9910, as indicated by the boldface. The organisers, using McNemar and Bootstrap tools, determined the result as statistically significant when compared to the closest competing score.

All scores presented in Table 2 are above 0.95. This can be attributed to the favourable circumstances of an in-domain problem. In-domain problems, where the test set originates from the same source as the training data, tend to yield high accuracy, as is evident in our results. This alignment between training and test data contributes to the robust performance of language models in such scenarios.

7 Conclusion

In this paper, we have presented our submission to the ALTA 2023 shared task, a binary classification challenge distinguishing generative AI content from human writing. Our proposed approach, using a Falcon-7B language model combined with label smoothing and model prompting, has demonstrated considerable promise. With a top-ranking accuracy score of 0.991, our system has showcased the effectiveness of these techniques in this specific task. Looking forward, there is an opportunity for further research and refinement. Future work should focus on extending our system's capabilities to tackle more challenging scenarios, including out-of-domain problems and multi-class authorship attribution tasks.

8 Acknowledgments

This research is supported in part by the Defence Science and Technology Group, Australia and the Australian Research Council Discovery Project DP200101441.

References

Harika Abburi, Michael Suesserman, Nirmala Pudota, Balaji Veeramani, Edward Bowen, and Sanmitra Bhattacharya. 2023. Generative ai text classification using ensemble llm approaches. *arXiv preprint arXiv:2309.07755*.

Eugene Bagdasaryan and Vitaly Shmatikov. 2022. Spinning language models: Risks of propaganda-as-a-service and countermeasures. In *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE.

Alessandro Bessi and Emilio Ferrara. 2016. Social bots distort the 2016 us presidential election online discussion. *First monday*, 21(11-7).

Pranav Bharadwaj and Zongru Shao. 2019. Fake news detection with semantic features and text mining. *International Journal on Natural Language Computing (IJNLC) Vol*, 8.

Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, et al. 2022. Gpt-neox-20b: An open-source autoregressive language model. *arXiv preprint arXiv:2204.06745*.

Andrew Blair-Stanek, Nils Holzenberger, and Benjamin Van Durme. 2023. Can gpt-3 perform statutory reasoning? *arXiv preprint arXiv:2302.06100*.

Stephan Bloehdorn and Andreas Hotho. 2006. Boosting for text classification with semantic features. In *Advances in Web Mining and Web Usage Analysis: 6th International Workshop on Knowledge Discovery on the Web, WebKDD 2004, Seattle, WA, USA, August 22-25, 2004, Revised Selected Papers 6*, pages 149–166. Springer.

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

Andy Coenen, Luke Davis, Daphne Ippolito, Emily Reif, and Ann Yuan. 2021. Wordcraft: a human-ai collaborative editor for story writing. *arXiv preprint arXiv:2107.07430*.

Debby RE Cotton, Peter A Cotton, and J Reuben Shipway. 2023. Chatting and cheating: Ensuring academic integrity in the era of chatgpt. *Innovations in Education and Teaching International*, pages 1–12.

Boris Delocski. 2023. Natural language processing and its applications in the finance sector.

Faisal R Elali and Leena N Rachid. 2023. Ai-generated research paper fabrication and plagiarism in the scientific community. *Patterns*, 4(3).

Rinaldo Gagiano, Maria Myung-Hee Kim, Xiuzhen Jenny Zhang, and Jennifer Biggs. 2021. Robustness analysis of grover for machine-generated news detection. In *Proceedings of the The 19th Annual Workshop of the Australasian Language Technology Association*, pages 119–127.

Matthew Gault. 2023. Ai spam is already flooding the internet and it has an obvious tell.

Cassio Goldschmidt. 2019. Council post: Ai-generated reviews threaten business reputations.

Elias Groll. 2023. Researchers: Large language models will revolutionize digital propaganda campaigns.

Chain Haas. 2023. Introducing bloomberggpt, bloomberg's 50-billion parameter large language model, purpose-built from scratch for finance.

Hasham Ul Haq, Veysel Kocaman, and David Talby. 2021. Deeper clinical document understanding using relation extraction. *arXiv preprint arXiv:2112.13259*.

Hasham Ul Haq, Veysel Kocaman, and David Talby. 2022. Mining adverse drug reactions from unstructured mediums at scale. In *Multimodal AI in healthcare: A paradigm shift in health intelligence*, pages 361–375. Springer.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Technology Innovation Institute. 2023. Falcon-7b. `https://huggingface.co/tiiuae/falcon-7b`.

Yafu Li, Qintong Li, Leyang Cui, Wei Bi, Longyue Wang, Linyi Yang, Shuming Shi, and Yue Zhang. 2023. Deepfake text detection in the wild. *arXiv preprint arXiv:2305.13242*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Liz Lumley. 2023. Large language models advance on financial services.

Larry Medsker and Lakhmi C Jain. 1999. *Recurrent neural networks: design and applications*. CRC press.

Elena-Adriana Minastireanu and Gabriela Mesnita. 2019. Light gbm machine learning algorithm to online click fraud detection. *J. Inform. Assur. Cybersecur*, 2019:263928.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. *arXiv preprint arXiv:2301.11305*.

Diego Molla, Haolan Zhan, Xuanli He, and Qiongkai Xu. 2023. Overview of the 2023 alta shared task: Discriminate between human-written and machine-generated text. In *Proceedings of the 21st Annual Workshop of the Australasian Language Technology Association (ALTA 2023)*.

OpenAI. 2023. Gpt-4 technical report.

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. *arXiv preprint arXiv:2306.01116*.

Tomas Pranckevičius and Virginijus Marcinkevičius. 2017. Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification. *Baltic Journal of Modern Computing*, 5(2):221.

Piotr Przybyła, Nicolau Duran-Silva, and Santiago Egea-Gómez. 2023. I've seen things you machines wouldn't believe: Measuring content predictability to identify automatically-generated text. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2023). CEUR Workshop Proceedings, CEUR-WS, Jaén, Spain*.

Zhang Qi. 2020. The text classification of theft crime based on tf-idf and xgboost model. In *2020 IEEE International conference on artificial intelligence and computer applications (ICAICA)*, pages 1241–1246. IEEE.

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.

Areg Mikael Sarvazyan, José Ángel González, Marc Franco-Salvador, Francisco Rangel, Berta Chulvi, and Paolo Rosso. 2023. Overview of autextification at iberlef 2023: Detection and attribution of machine-generated text in multiple domains. *arXiv preprint arXiv:2309.11285*.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Kanish Shah, Henil Patel, Devanshi Sanghvi, and Manan Shah. 2020. A comparative analysis of logistic regression, random forest and knn models for the text classification. *Augmented Human Research*, 5:1–16.

Roger Alan Stein, Patricia A Jaques, and Joao Francisco Valiati. 2019. An analysis of hierarchical text classification using word embeddings. *Information Sciences*, 471:216–232.

Massimo Stella, Emilio Ferrara, and Manlio De Domenico. 2018. Bots increase exposure to negative and inflammatory content in online social systems. *Proceedings of the National Academy of Sciences*, 115(49):12435–12440.

Chaofei Tang, Nurbol Luktarhan, and Yuxin Zhao. 2020. An efficient intrusion detection method based on lightgbm and autoencoder. *Symmetry*, 12(9):1458.

Dietrich Trautmann, Alina Petrova, and Frank Schilder. 2022. Legal prompt engineering for multilingual legal judgement prediction. *arXiv preprint arXiv:2212.02199*.

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.

Jan Philip Wahle, Terry Ruas, Frederic Kirstein, and Bela Gipp. 2022. How large language models are transforming machine-paraphrased plagiarism. *arXiv preprint arXiv:2210.03568*.

Yuxia Wang, Jonibek Mansurov, Petar Ivanov, Jinyan Su, Artem Shelmanov, Akim Tsvigun, Chenxi Whitehouse, Osama Mohammed Afzal, Tarek Mahmoud, Alham Fikri Aji, et al. 2023. M4: Multi-generator, multi-domain, and multi-lingual black-box machine-generated text detection. *arXiv preprint arXiv:2305.14902*.

Xi Yang, Aokun Chen, Nima PourNejatian, Hoo Chang Shin, Kaleb E. Smith, Christopher Parisien, Colin Compas, Cheryl Martin, Anthony B. Costa, Mona G. Flores, Ying Zhang, Tanja Magoc, Christopher A. Harle, Gloria Lipori, Duane A. Mitchell, William R. Hogan, Elizabeth A. Shenkman, Jiang Bian, and Yonghui Wu. 2022. A large language model for electronic health records. *npj Digital Medicine*, 5(1):194.

Dani Yogatama, Chris Dyer, Wang Ling, and Phil Blunsom. 2017. Generative and discriminative text classification with recurrent neural networks. *arXiv preprint arXiv:1703.01898*.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. *Advances in neural information processing systems*, 32.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Yujun Zhou, Bo Xu, Jiaming Xu, Lei Yang, and Changliang Li. 2016. Compositional recurrent neural networks for chinese short text classification. In *2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pages 137–144. IEEE.

Automatic Detection of Machine-Generated Text Using Pre-Trained Language Models

Yunhao Fang
School of Computer Science and Information Systems
The University of Melbourne
`yunhfang@student.unimelb.edu.au`

Abstract

In this paper, I provide a detailed description of my approach to tackling the ALTA 2023 shared task whose objective is to build an automatic detection system to distinguish between human-authored text and text generated from Large Language Models. By leveraging several pre-trained language models through model fine-tuning as well as the multi-model ensemble, the system managed to achieve second place on the test set leaderboard in the competition.

1 Introduction

Large Language Models (LLMs) have experienced a drastic advancement over the past few years and brought a revolution to the domain of Natural Language Processing (Gordijn and Have, 2023). Through the expansion of model parameters and the intensive pre-training on a large corpus, recent LLMs such as GPT-4 (OpenAI, 2023) and Llama2 (Touvron et al., 2023) have shown their capability to understand the human language and generate high-quality text.

However, the growing attention to LLMs and their increasing availability to the public nowadays has inevitably led to some concerns as these models can be used in an inappropriate manner to cause harm to society. This includes fake news generation (Zellers et al., 2019), fake product reviews generation (Adelani et al., 2020) and plagiarism (Dehouche, 2021). Therefore, this calls for the construction of a reliable machine-generated text detection system to regulate the use of LLMs so that we can make the most of them. To explore the effective ways that can achieve this objective, ALTA 2023 (Molla et al., 2023) organised a shared task with the goal of constructing an automatic detection system to distinguish between the human-authored text and text generated by the LLMs. The task is formed as a binary classification problem.

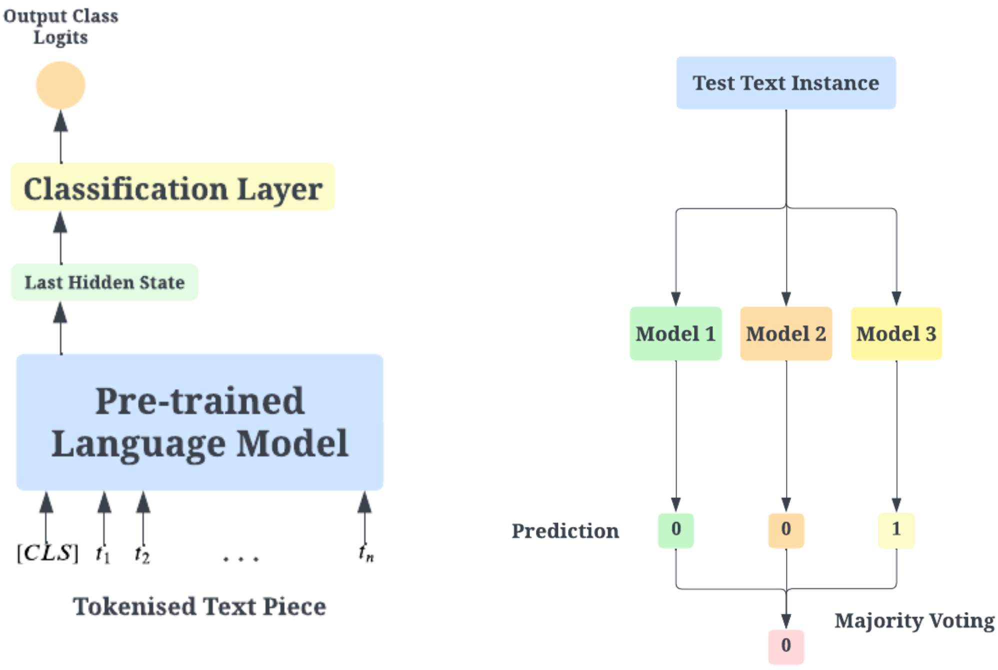
My team handled this task through the utilisation of some representative pre-trained models to tackle the classification problem for machine-generated vs human-authored text given the fact that they have already exhibited their strength in various Natural Language Processing tasks. The models I experimented with include the vanilla BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019) and DeBERTaV3 (He et al., 2022a) which represent the chain of improvement for the BERT-based models. I also implemented an ensemble model via majority voting over the best models to further enhance the performance. The rest of the paper will provide a detailed explanation of the design of my system as well as the performance with respect to the task.

2 Related Work

2.1 Machine-Generated Text Detection

Recent studies related to the construction of automatic machine-generated text detection systems focus on the utilisation of the source generator to assist the detection. One area of research intended to rely on internal information from the generative models, such as the probability distribution of tokens or text sequences assigned by the generator, to construct the detector (Mitchell et al., 2023). The other group of researchers proposed the incorporation of the watermarking technique into the generative models by introducing some signals inside the text that cannot be perceived by humans but are detectable by machines. (Kirchenbauer et al., 2023; He et al., 2022b). However, these approaches suffer from their practicality since there exist numerous proprietary LLMs in the industry where the developers are reluctant to expose the internal details of their models, and it is also difficult to guarantee that every LLM developer agrees on the incorporation of watermarking into their models. Therefore, my detection system aims to obtain a good performance under the “black-box” scenario where only the generated text from the generative models is accessible.

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 159–163
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics



(a) The architecture of the detection system based on Pre-trained Language Models

(b) Multi-model ensemble through majority voting

Figure 1: Illstration of the automatic detection system

2.2 Pre-Trained Language Models

My detection system took advantage of several pre-trained language models by constructing the classifiers upon these models to differentiate machine-generated and human-authored text. This section will provide a description of the models that have been applied during the model development phase.

2.2.1 BERT

BERT (Devlin et al., 2019), which stands for Bidirectional Encoder Representation from Transformer, aims to learn the deep bidirectional contextual representation of the language through pre-training on a large text corpus. It attains this objective through the conduction of unsupervised tasks during pre-training to learn the language patterns from the text, which includes the Masked Language Model (MLM) and Next Sentence Prediction (NSP). MLM intends to predict the tokens that are masked randomly in the text to capture the bidirectional information of the token, while NSP attempts to understand the relationship between two sentences by predicting whether one sentence follows the other.

2.2.2 RoBERTa

RoBERTa (Liu et al., 2019) is an extension of the vanilla BERT with the goal of optimising the design choices and training strategies of BERT to boost the performance on the downstream tasks. It replaced the static masking in BERT with dynamic masking to avoid duplicated masks and removed the NSP objective from BERT. In addition to this, RoBERTa is also pre-trained on a higher volume of data for a longer time and over a larger batch size compared to BERT.

2.2.3 DeBERTaV3

The original DeBERTa model (He et al., 2020) managed to make a further enhancement on both BERT and RoBERTa through the introduction of two novel techniques: disentangled attention and enhanced mask decoder. A recently upgraded version of DeBERTa called DeBERTaV3 (He et al., 2022a) was proposed by the authors to replace the MLM objective from BERT with Replaced Token Detection (RTD), where a generator is employed to generate corrupted tokens inside the text and the model is trained as a discriminator to determine whether the token is the original one or has been corrupted. It also proposed a method called gradient-disentangled embedding sharing (GDES) to handle the embeddings from the generator and the discriminator in an effective way.

3 Dataset

The dataset provided by the ALTA 2023 shared task (Molla et al., 2023) consists of text pieces of human-authored and machine-generated text across a wide range of domains. The machine-generated text inside the dataset originates from different types of LLMs. The statistics of the dataset are presented in Table 1. The labels are only contained in the training set where the label assigned to each text piece is either 1 or 0, with 0 indicating that the text is generated by the machine and 1 indicating that the text is written by the human. The distribution of the labels inside the training dataset is 50% for machine-generated and 50% for human-authored which is well-balanced.

Category	Size
Training	18,000
Development	2,000
Test	2,000
Total	22,000

Table 1: Statistics of the dataset for ALTA 2023 shared task

The training set and the development set are released at the same time for model development and the test set is used for the final evaluation of the models and the determination of the rank in the competition.

4 Methodology

Following the process explained in BERT (Devlin et al., 2019), the pre-trained language models discussed in Section 2.2 are adopted to build binary classifiers by adding a single classification layer on top of the last hidden state of the first token (the special '[CLS]' token added by these pre-trained language models) for each of them, which is the contextual representation of the full text. The model architecture is shown in Figure 1a. The original text pieces are tokenised using the corresponding tokeniser for each model and the tokens are input into the classifier. The classifiers are then fine-tuned on the provided training set so that they can learn the language patterns inside the data. The resulting models will be applied to make predictions about the development and test set to gain insight into their performance.

Besides the employment of each single pre-trained language model to perform classification and obtain the results, I've further performed the multi-model ensemble through majority voting over the prediction results from the 3 models that express the best performance. The process is demonstrated in Figure 1b. The voting is conducted as a hard voting where for each instance of the text pieces inside the test set, the label that is assigned to the text by most of the classifiers will be selected as the final label. The logic behind this is to improve the robustness of the detection system by combining the results from multiple models.

Hyperparameter	Value
Learning rate	2e-5
Batch size	64
Training epochs	5
Max length	100

Table 2: Hyperparameter Setting in the experiment

5 Experiments

5.1 Experimental Settings

During the experimental stage, I utilised the pre-trained language models from huggingface to build the classifiers and perform fine-tuning, which includes the models discussed in Section 2.2 with varied size: 1) *bert-base-cased* [1], 2) *bert-large-cased* [2], 3) *roberta-base* [3], 4) *roberta-large* [4], 5) *microsoft/deberta-v3-base* [5], 6) *microsoft/deberta-v3-large* [6]. I used BCEWithLogitsLoss [7] as the loss function and AdamW (Loshchilov and Hutter, 2019) as the optimizer during the model training phase. The setting of the hyperparameters used for the experiment is indicated in Table 2. All the implemented models applied the same experimental settings to compare the performance between each other.

---

[1] `https://huggingface.co/bert-base-cased`

[2] `https://huggingface.co/bert-large-cased`

[3] `https://huggingface.co/roberta-base`

[4] `https://huggingface.co/roberta-large`

[5] `https://huggingface.co/microsoft/deberta-v3-base`

[6] `https://huggingface.co/microsoft/deberta-v3-large`

[7] `https://pytorch.org/docs/stable/generated/torch.nn.BCEWithLogitsLoss.html`

Model	Version	Development set	Test set
BERT	*bert-base-cased*	0.986	0.976
	bert-large-cased	-	0.980
RoBERTa	*roberta-base*	0.985	0.981
	roberta-large	0.991	0.985
DeBERTaV3	*microsoft/deberta-v3-base*	0.984	0.978
	microsoft/deberta-v3-large	**0.992**	0.982
Ensemble	-	-	**0.990**

Table 3: Classification accuracy of different models on development and test set

The performance of the resulting models is evaluated using the accuracy_score [8] from scikit-learn as specified by the ALTA 2023 shared task.

5.2 Results

Table 3 shows the classification accuracy of all the fine-tuned pre-trained language models as well as the ensemble model involved in the experiment over the development set and test set. As indicated in the table, for all types of pre-trained language models, the large version of the models obtain a better performance compared to the base ones on both the development and the test set. This illustrates the fact that larger models with more parameters have the ability to learn more language patterns from the text to distinguish between human-authored and machine-generated text. Additionally, all versions of BERT underperform RoBERTa and DeBERTaV3 on the test set, while RoBERTa and DeBERTaV3 express a comparable performance between each other. This suggests that the evolution of the BERT model makes contributions to the classification of machine-generated and human-authored text similar to most of the NLP tasks. The results from the table also demonstrate the effectiveness of the multi-model ensemble as the ensemble model using majority voting outperforms all the single models by a certain amount on the test set.

6 Conclusion

In this paper, I've presented my automatic detection system for the ALTA 2023 shared task that classifies machine-generated and human-authored text. The capability of pre-trained language models in handling the task is demonstrated by fine-tuning them on the dataset and constructing the classifiers. The benefits that the multi-model ensemble brings to the performance of the detector are also indicated by the experiment results. As a result, the best system achieves second place in the ALTA 2023 shared task.

Acknowledgements

This research was supported by The University of Melbourne's Research Computing Services and the Petascale Campus Initiative for providing computational resources and I also thank the ALTA 2023 shared task organisers for providing the dataset and organising the task.

References

David Ifeoluwa Adelani, Haotian Mai, Fuming Fang, Huy H Nguyen, Junichi Yamagishi, and Isao Echizen. 2020. Generating sentiment-preserving fake online reviews using neural language models and their human-and machine-based detection. In *Advanced Information Networking and Applications: Proceedings of the 34th International Conference on Advanced Information Networking and Applications (AINA-2020)*, pages 1341–1354. Springer.

Nassim Dehouche. 2021. Plagiarism in the age of massive generative pre-trained transformers (gpt-3). *Ethics in Science and Environmental Politics*, 21:17–23.

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.

Bert Gordijn and Henk ten Have. 2023. Chatgpt: evolution or revolution? *Medicine, Health Care and Philosophy*, 26(1):1–2.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2022a. Debertav3: Improving deberta using electra-style pre-

---

[8] `https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html`

training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Xuanli He, Qiongkai Xu, Lingjuan Lyu, Fangzhao Wu, and Chenguang Wang. 2022b. Protecting intellectual property of language generation apis with lexical watermark. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10758–10766.

John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. 2023. A watermark for large language models. *arXiv preprint arXiv:2301.10226*.

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*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature.

Diego Molla, Haolan Zhan, Xuanli He, and Qiongkai Xu. 2023. Overview of the 2023 alta shared task: Discriminate between human-written and machine-generated text. In *Proceedings of the 21st Annual Workshop of the Australasian Language Technology Association (ALTA 2023)*.

OpenAI. 2023. Gpt-4 technical report.

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

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. *Advances in neural information processing systems*, 32.

An Ensemble Based Approach To Detecting LLM-Generated Texts

Ahmed El-Sayed, Omar Nasr
Arab Academy for Science and Technology
`{ahmedelsayedhabashy,omarnasr5206}@gmail.com`

Abstract

Recent advancements in Large Language models (LLMs) have empowered them to achieve text generation capabilities on par with those of humans. These recent advances paired with the wide availability of those models have made Large Language models adaptable in many domains, from scientific writing to story generation along with many others. This recent rise has made it crucial to develop systems to discriminate between human-authored and synthetic text generated by Large Language models (LLMs). Our proposed system for the ALTA shared task, based on ensembling a number of language models, claimed first place on the development set with an accuracy of 99.35% and third place on the test set with an accuracy of 98.35%.

1 Introduction

In the realm of human-computer interactions, the recent advancements in AI-generated texts are hallmarked by the introduction of Large Language Models (LLMS), such as GPT4 (OpenAI, 2023), GPT3 (Brown et al., 2020), T5 (Raffel et al., 2020), LLAMA (Touvron et al., 2023) and much more. This has resulted in AI's ability to generate text of high quality and fluency comparable to that of humans. These language models have had widespread integration and adaptations across many different fields including but not limited to, law, medicine and education. Nonetheless, similar to any revolutionary technology, LLMs possess both positive and negative aspects for our society. Apart from spreading misleading information, the potential misuse of LLMs could lead to numerous social and ethical challenges, such as academic misconduct (Yun et al., 2023) and spread of misinformation (Else, 2023). The recent growth in adaption of Large Language Models in many domains and their unprecedented ability to generate high quality fluent text similar to that of humans have caught researchers' attention. This lead to the development of systems with the goal of being able to differentiate between human-generated texts and machine-generated ones. Those systems vary according to their scope of operation, ranging from domain specific ones that detect deep fakes based on specific models to more generalized ones, yet there have been efforts to build a unified model able to operate on different domains and generalize to novel LLMs despite not being trained on their respective data. Large Language Models are expected to fundamentally change many aspects of life and with the trend in the number of Large Language Models introduced each year (Naveed et al., 2023), The challenges of detecting text generated by Large Language Models are expected to reach new heights in the upcoming years. The ALTA 2023 shared task (Molla et al., 2023) focuses on this important topic, offering a dataset for evaluation and training. The dataset addresses several issues and supports the creation of a single, readily generalizable model. Our proposed model uses an ensemble-based approach paired with fine-tuning a number of language models. The structure of this research paper will unfold as follows: The related work section will provide an overview of various solutions explored by different researchers in the context of this problem. Subsequently, the data section will detail the properties of the provided dataset and any preprocessing steps undertaken. In the system description section, we will go through the architecture of our proposed model. The results section will then offer a detailed analysis of the outcomes generated by the proposed system, complemented by a comprehensive evaluation of the model's overall performance. Finally, the summary section will synthesize the paper's content, briefly touch on potential future research directions, and consider possible improvements to the model.

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 164–168
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

2 Related Work

We will touch on the most recent developments in identifying data produced by Large Language models in the section that follows. Because of the widespread use of LLMs and their possible drawbacks, academics have been particularly interested in this area in recent years. Many researchers have proposed systems that use both deep learning techniques and traditional machine learning models. One interesting approach was when (Solaiman et al., 2019) built a logistic regression based detector which made use of TF-IDF unigram and bigram features. The model was trained on GPT-2 outputs and WebText samples and yielded an accuracy up to 97% at 124 million parameters and up to 93% at 1.5 billion parameters. (Fröhling and Zubiaga, 2021) experimented with a number of conventional machine learning approaches, mainly Support Vector Machines, Random Forests and Logistic Regression. In the realm of deep learning, many models were proposed to tackle the problem of AI-generated text, yet many of the proposed systems either focused on specific domains, or they were model specific (Yang et al., 2023; Mitchell et al., 2023). One interesting system was proposed by (Li et al., 2023) which consisted of training 3 detection models; a language model based on Longformer (Beltagy et al., 2020), FastText (Joulin et al., 2016) and GLTR (Gehrmann et al., 2019) and testing the model on multiple settings to ensure its success ranging from domain-specific & model-specific to unseen domains and unseen models. Many studies have also shown that text written by LLM is more objective and less emotional than human-generated text (Webber et al., 2020). Another factor has to do with the fact that LLMs have a condition called hallucinations, which results from the generation of material that is nonsensical or inconsistent (Ji et al., 2023). Something that makes it possible to apply fact-verification procedures. A different strategy is known as "white box detection," where the detector can monitor any unauthorized or suspicious behavior by inserting hidden watermarks into its outputs and having complete access to the target language model (Abdelnabi and Fritz, 2021).

3 Data

The dataset used is the dataset provided in the ALTA 2023 shared task. Below is an illustration of the dataset distribution. The dataset is derived from a number of sources, including several LLMs (e.g., T5, GPT-X) and domain sources (e.g., legal, medical). The labels are AI-generated and Human-generated, represented as 1 and 0 respectively, which formulate a Binary Classification problem. There were 9000 samples in the training set for each of the corresponding labels, spread evenly. Other than the language model-specific preprocessing, no further preprocessing was used.

Dataset	Train	Dev	Test
Texts	18000	2000	2000

Table 1: Data distribution for the task.

4 System Description

In the subsequent section, we will outline our experimentation on the dataset, highlighting the key stages involved in the development of the previously mentioned system.

4.1 Conventional Machine Learning Models

Our approach commenced with word embedding utilizing diverse pretrained word embedding, incorporating padding, and iterative experimentation with various models such as Support Vector Machines and Logistic Regression. While initially productive, these models did not produce satisfactory results. Consequently, we pivoted towards exploring Deep Learning methodologies, focusing primarily on Language Models to enhance the outcomes.

4.2 Language Models

Language models have demonstrated outstanding results on a variety of tasks in recent years. Other researchers have expanded on this accomplishment by creating other models based on BERT (Devlin et al., 2018). Using the dateset we were given, we fine-tuned many BERT-based models. After evaluating the fine-tuning of DistilBERT (Sanh et al., 2019) on our given dataset, achieving an accuracy of 98.5% on the development set, we decided to adopt Roberta (Liu et al., 2019) as our primary model due to its strong performance in similar scenarios (Zhan et al., 2023). Specifically, fine-tuning Roberta resulted in an impressive accuracy of 99.15%. Additionally, XLMRoberta (Conneau et al., 2019) demonstrated a high accuracy of 98.75%, affirming our decision to select Roberta as the foundational model for our development. While experimenting with different hyperparameters for

Roberta, we maintained a consistent accuracy of 99.15%, indicating that higher results were not attainable. Nevertheless, a notable finding was that, despite identical prediction accuracy to our initial model, there were disparities in the predictions. This realization prompted us to implement an ensemble approach.

4.3 Ensembling

An ensemble of machine learning models is a method that combines many different machine learning models, often of different kinds or versions, to enhance robustness, generalization, and predictive performance. By utilizing the combined intelligence of several models, this method outperforms utilizing a single model in terms of prediction accuracy and stability. Our approach involved employing hard voting, a technique where multiple individual models are trained and make predictions on a given dataset. The final prediction is determined through a "voting" mechanism, where each model in the ensemble "votes" for a specific class (in classification tasks). The final output of the ensemble is based on the majority of votes for a particular class or prediction. We experimented with ensembling multiple learners; DistillBERT, XLMRoberta and Roberta Base, then we ensembled multiple Roberta base models. This resulted in the highest performance of the development set. One approach that was only used in an unofficial submission is ensembling Roberta large models, which was found to outperform our previously mentioned models.

4.4 Experiment settings

The training procedure was conducted using the Google Colab platform for training our pipeline, which has 12.68 GB of RAM, a 14.75 GB NVIDIA Tesla T4 GPU, and Python language. We used ktrain's (Maiya, 2020) fit one cycle, which applies a one cycle policy (Smith, 2018). The learning rate was determined via the lr_plot function, which experiments with a range of learning rates and suggests multiple possible learning rates. The parameters set for our experiment are mentioned in Table 2.

Parameter	Value
Epochs	10
Learning Rate	Varying
Batch Size	Varying
Max Length	128
Optimizer	AdamW
Loss Function	Binary Cross Entropy

Table 2: Training parameters.

We experimented with 3 different learning rates for Roberta of 1e-5, 2e-5 and 8.675e-6 as well as different batch sizes of 32, 64 and 128.

5 Results

This section examines how well our suggested AAST-NLP system performed in the ALTA-2023 shared task related to the identification of data produced by big language models. Table 3 presents our results, some of which were not evaluated because of submission limit restrictions.

Model Used	Validation	Test
BASELINE	50.3%	_
DistillBERT	98.5%	_
XLMRoberta	98.75%	_
Roberta BASE	99.15%	98.25%
Ensemble 1	99.3%	98.35%
Ensemble 2	99.35%	98.35%
Ensemble 3	99.3%	98.6%

Table 3: Accuracy of the models on the respective datasets. Ensemble 1 refers to an ensemble of DistilBERT, Roberta and XLMRoberta. Ensemble 2 refers to an ensemble of 3 Roberta-base models. Ensemble 3 refers to an ensemble of 3 Roberta-large models.

Our ensembled models performed the best of the suggested systems, placing first on the test set and third on the development set, suggesting some progress on ensembling multiple learners. Due to computational power constraints, we initially conducted our experiments using Roberta-base. However, after experimenting with Roberta-large, we discovered that when three Roberta-large models were ensembled, they outperformed our top ranking system on the test set.

6 Discussion

The results of these experiments showed that an ensembling-based approach is worth further exploring in the pursuit of a generalized model for classifying synthetic text data generated by LLMS. Some potential further improvements include adding larger models to the ensemble, such as xlm-roberta-XL. Other improvements include supplementing the development data set with more training data such as the one used in (Li et al., 2023). Another approach could be to further tune the hyperparameters of the individual members of the ensemble, which could lead to marginal improvements in the overall performance of the ensemble. Overall, the system has promising implications and, with more research, could prove very fruitful in combating the spread of fake data in the modern world. Addressing this problem is a very pressing matter as this spread of fake synthetic text data could spread far and wide and have catastrophic effects on the journalism industry,the education industry, along with several other industries.

7 Summary

The presented system, utilizing an Ensemble approach through Hard Voting, was thoroughly described in this study. The conducted experiments were comprehensively addressed. Incorporating pretrained language models, along with ensembling, effectively addresses the challenge of identifying text generated by extensive language models, though there remains room for enhancement. Our forthcoming research will concentrate on evaluating our model in analogous settings, utilizing data generated by recently developed Large Language Models across diverse domains to assess its performance. Another compelling avenue for future investigation involves conducting additional experiments with larger language models, particularly emphasizing the adaptation of Roberta-large to our specific problem. This aspect warrants further exploration in subsequent research endeavors.

References

Sahar Abdelnabi and Mario Fritz. 2021. Adversarial watermarking transformer: Towards tracing text provenance with data hiding.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The Long-Document Transformer. *arXiv (Cornell University)*.

T. 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, A. Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric J. 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. *arXiv (Cornell University)*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv (Cornell University)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv (Cornell University)*.

Holly Else. 2023. Abstracts written by ChatGPT fool scientists. *Nature*, 613(7944):423.

Leon Fröhling and Arkaitz Zubiaga. 2021. Feature-based detection of automated language models: tackling GPT-2, GPT-3 and Grover. *PeerJ*, 7:e443.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander Rush. 2019. GLTR: Statistical detection and visualization of generated text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 111–116, Florence, Italy. Association for Computational Linguistics.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification.

Yafu Li, Qintong Li, Leyang Cui, Wei Bi, Longyue Wang, Linyi Yang, Shuming Shi, and Yue Zhang. 2023. Deepfake text detection in the wild.

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 (Cornell University)*.

Arun S. Maiya. 2020. ktrain: A low-code library for augmented machine learning. *arXiv preprint arXiv:2004.10703*.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature.

Diego Molla, Haolan Zhan, Xuanli He, and Qiongkai Xu. 2023. Overview of the 2023 alta shared task: Discriminate between human-written and machine-generated text. In *Proceedings of the 21st Annual Workshop of the Australasian Language Technology Association (ALTA 2023)*.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2023. A comprehensive overview of large language models.

OpenAI. 2023. GPT-4 Technical Report. *arXiv (Cornell University)*.

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.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv (Cornell University)*.

Lauren Smith. 2018. A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay. *arXiv (Cornell University)*.

Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, and Jasmine Wang. 2019. Release strategies and the social impacts of language models. *arXiv (Cornell University)*.

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, Édouard Grave, and Guillaume Lample. 2023. LLAMA: Open and Efficient Foundation Language Models. *arXiv (Cornell University)*.

Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors. 2020. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Online.

Xianjun Yang, Wei Cheng, Yue Wu, Linda Petzold, William Yang Wang, and Haifeng Chen. 2023. Dna-gpt: Divergent n-gram analysis for training-free detection of gpt-generated text.

Hye Sun Yun, Iain J. Marshall, Thomas A Trikalinos, and Byron C. Wallace. 2023. Appraising the potential uses and harms of LLMs for medical systematic reviews. *arXiv (Cornell University)*.

Haolan Zhan, Xiaoqiong He, Qiongkai Xu, Yuxiang Wu, and Pontus Stenetorp. 2023. G3Detector: General GPT-Generated Text Detector. *arXiv (Cornell University)*.

Feature-Level Ensemble Learning for Robust Synthetic Text Detection with DeBERTaV3 and XLM-RoBERTa

Saman Sarker Joy and **Tanusree Das Aishi**
Department of Computer Science and Engineering
BRAC University, 66 Mohakhali, Dhaka-1212, Bangladesh
`{saman.sarker.joy, tanusree.das.aishi}@g.bracu.ac.bd`

Abstract

As large language models, or LLMs, continue to advance in recent years, they require the development of a potent system to detect whether a text was created by a human or an LLM in order to prevent the unethical use of LLMs. To address this challenge, ALTA Shared Task 2023 introduced a task to build an automatic detection system that can discriminate between human-authored and synthetic text generated by LLMs. In this paper, we present our participation in this task where we proposed a feature-level ensemble of two transformer models namely DeBERTaV3 and XLM-RoBERTa to come up with a robust system. The given dataset consisted of textual data with two labels where the task was binary classification. Experimental results show that our proposed method achieved competitive performance among the participants. We believe this solution would make an impact and provide a feasible solution for detection of synthetic text detection.

1 Introduction

In recent years, the remarkable advancements in Large Language Models (LLMs) have showcased an unprecedented revolution in the field of Natural Language Processing (Raiaan et al., 2023). These models e.g. GPT-X and T5, demonstrate the ability to generate text that closely resembles content created by humans. However, there is a risk of abuse that makes this a double-edged sword and raises moral questions. The spread of synthetic text produced by LLMs carries the risk of spreading false information (Bian et al., 2023), interfering in elections (Schneier, 2023), and jeopardize the credibility of scientific knowledge (Birhane et al., 2023).

In this context, the ALTA Shared Task 2023[1] introduced a task where researchers have to develop automated detection systems with the capacity to discriminate between human-written text and text generated by Large Language Models (LLMs). The aim of this task is to mitigate the unethical application of LLMs and promote their conscientious and responsible utilization in various domains.

The dataset provided by the ALTA Shared Task 2023 is used in building and assessing the automatic text detection systems. The task is fundamentally a binary classification problem. Each text is labeled as either 0 (AI-generated) or 1 (human-generated). The text samples are derived from diverse domains, including law and medicine, and span a spectrum of LLMs. The efficacy of the models will be assessed based on their accuracy and their resilience in detecting synthetic text. This paper presents our approach to this task, where we at first performed some data analysis on the dataset. Then using those analysis, we have proposed a feature-level ensemble model that utilizes the strengths of two state-of-the-art transformer models: DeBERTaV3 and XLM-RoBERTa. We believe that our proposed method, refined through rigorous experimentation, has achieved competitive and robust performance, positioning it as a promising solution among the participating models.

Text	**Label**
I am asking you this because the fans of the band are completely devoted. They experience the days leading up to the concert very intensely.	0
A Reston man has been charged with abduction after police say he dragged a woman from the sidewalk and tried to remove her clothes.	1

Table 1: Example of ALTA Shared Task 2023. Here, the two labels are 0 (for AI-generated) or a 1 (for human-generated).

---

[1] `https://www.alta.asn.au/events/sharedtask2023/description.html`

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 169–172
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

Text	Label	ID	Language Found
Rektor på Gammel Hellerup Gymnasium, Jørgen Rasmussen, ønsker ikke at udtale sig om sagen.	1	36	German
En réalité, la superteam qui semble se profiler est composée de :	1	5628	French
E le parole “programma di aggiustamento strutturale”, “ristrutturazione” e “default” si possono benissimo tradurre in genocidio sociale.	1	15541	Italian

Table 2: Examples of text in the train set containing other languages.

2 Dataset Description

The dataset statistics are summarized in Table 3. Each text in the dataset is tagged with either a 0 (for AI-generated) or a 1 (for human-generated). The training dataset is evenly balanced, ensuring an equal number of both categories.

In our analysis, we found that text length in the training set typically range from 10 to 50 and never exceed 200 which can be seen in Figure 1. Also, we came across that some words of the texts are in languages other than English, e.g. German, Afrikaans, Romanian, French, etc., in about 124 rows of the train dataset. Some examples are shown in Table 2. This indicates a bit of multilingual content needs to be considered as well.

Category	Data
Train	18000
Validation	2000
Test	2000

Table 3: Dataset Splits.

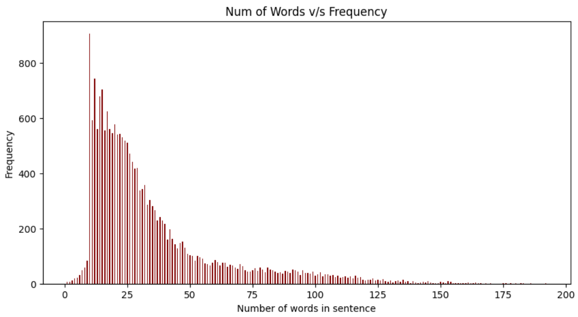


Figure 1: Frequency of all the text in the training set. It shows that all the text lengths are less than 200.

3 System Overview

3.1 Transformer Model

Transformers (Vaswani et al., 2017) are widely used in NLP tasks because they excel at various tasks. They offer high performance, scalability, and flexibility, making them a popular choice for many applications.

3.1.1 DeBERTaV3

DeBERTaV3 (He et al., 2023) is a new pretrained language model that improves upon the original DeBERTa (He et al., 2021) model. It does so by using a pretraining task called replaced token detection (RTD) instead of the traditional mask language modeling (MLM) task, which is more sample-efficient.

3.1.2 XLM-RoBERTa

XLM-RoBERTa (Conneau et al., 2019), a large-scale multilingual language model based on Facebook’s RoBERTa (Liu et al., 2019). XLM-RoBERTa undergoes pretraining on an extensive 2.5TB dataset of filtered CommonCrawl data.

3.2 Training Strategies

3.2.1 Ensamble Learning

Our model utilizes feature-level ensemble learning by independently extracting valuable information from two pre-trained models, DeBERTaV3 and XLM-RoBERTa. We extract essential details from each of these models and then effectively combine them to enhance our model’s performance. This approach is known as feature-level ensemble learning.

3.2.2 Model Architecture

We used the base versions (12 layers) of both DeBERTaV3 and XLM-RoBERTa. We made this

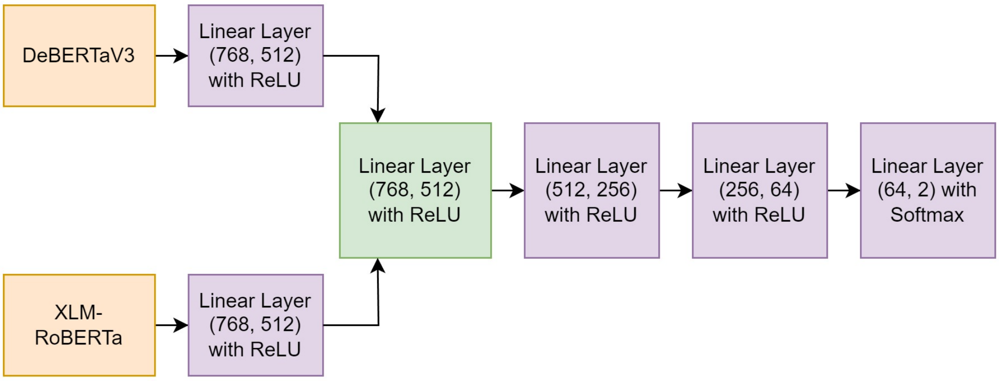

DeBERTaV3
Linear Layer (768, 512) with ReLU
XLM-RoBERTa
Linear Layer (768, 512) with ReLU
Linear Layer (768, 512) with ReLU
Linear Layer (512, 256) with ReLU
Linear Layer (256, 64) with ReLU
Linear Layer (64, 2) with Softmax

Figure 2: DeBERTaV3 and XLM-RoBERTa concatenation of the last layer + MLP Architecture.

choice due to computation limitations, as these base versions are more computationally efficient while still delivering effective results.

In this setup, the last layer of the DeBERTaV3 and XLM-RoBERTa models is used as the input. These representations are then passed through additional linear layers and fine-tuned on the specific task or dataset during the training phase. This allows the model to adapt to the task while benefiting from the pretrained language representation capabilities of DeBERTaV3 and XLM-RoBERTa. From Figure 2, we can see an overview of our model.

3.2.3 Data Augmentation

We had a validation set for which we initially did not have the correct labels. However, when we trained our initial model, we achieved a high level of accuracy in the validation set. So, we decided to include the validation data along with the predicted labels it generated into our original training dataset. This adjustment improved our models' performance.

4 Experiments and Evaluation

4.1 Experimental Settings

For hyperparameters, we have taken the number of epochs for training as 20, the learning rate is 1e-5, maximum length is 200, batch size of 8, the loss function is Cross Entropy Loss and the optimizer is AdamW (Loshchilov and Hutter, 2017).

4.2 Evaluation Metric

The ALTA Shared Task 2023 organizers employed a standard evaluation metric accuracy to evaluate the participants' system. They calculated the accuracy score using scikit-learn's (Pedregosa et al., 2011) accuracy_score package.

4.3 Results and Analysis

From Table 4, the highest accuracy in this competition was achieved by "OD-21" securing the top position with a score of 0.9910.

Our team, "SamNLP" submitted which achieved an accuracy of 0.9820, securing the 4th position in the original contest. This initial model achieved validation dataset accuracy of 0.9930, signifying that 1986 out of 2000 samples were accurately classified. Consequently, we integrated this high-performing dataset into our training data through augmentation. The model we later developed with the validation data added to the training set, was not initially submitted during the contest but was submitted after the contest had concluded. The rank that is listed 2nd is not the original rank it achieved during the contest but rather represents the rank it would have attained if it had been submitted as part of the competition.

A noteworthy observation is the marginal differences in accuracy among the top-performing teams. The variations in accuracy between the top-ranking teams are quite low, suggesting that the competition was highly competitive and challenging.

5 Conclusion

In conclusion, the growing capabilities of Large Language Models (LLMs) have brought both opportunities and challenges in the field of Natural Language Processing. The rise of synthetic text generated by LLMs has raised ethical concerns, including the spread of misinformation and potential misuse in various domains. To address this, the ALTA Shared Task 2023 was introduced. In this paper, we presented our approach to this task, where we focused on building a feature-level ensemble

Team Name	Accuracy	Position
SamNLP (Ours)**	0.9820	4th**
SamNLP (Ours with validation data added)*	0.9855	2nd*
Competitive performance of top-ranked methods		
OD-21	**0.9910**	1st
DetectorBuilder	0.9845	2nd
AAST-NLP	0.9835	3rd
Organizers	0.9765	5th
VDetect	0.9715	6th
cantnlp	0.9675	7th

Table 4: Comparative performance of our proposed method along with top-performing participants' method. The double asterisk (**) represents the actual position for the test dataset, while the single asterisk (*) denotes the model and accuracy achieved in the test dataset after the conclusion of the contest.

model using two state-of-the-art transformer models. We conducted a comprehensive analysis of the dataset, which revealed the need to handle multilingual content. Our approach leveraged feature-level ensemble learning, utilizing the strengths of both models, and included data augmentation to enhance performance. While we secured the 4th position in the original contest, the inclusion of validation data improved our model's accuracy, bringing it to the 2nd position when submitted after the contest's conclusion. Notably, the top-performing teams in the competition exhibited marginal differences in accuracy, emphasizing the high level of competitiveness in the task. We believe that our proposed method provides a promising solution for the detection of synthetic text, contributing to the responsible and conscientious use of LLMs in various applications. As LLMs continue to evolve, robust detection systems like the one presented in this paper become increasingly important to address the ethical challenges associated with AI-generated text.

References

Ning Bian, Hongyu Lin, Peilin Liu, Yaojie Lu, Chunkang Zhang, Ben He, Xianpei Han, and Le Sun. 2023. Influence of external information on large language models mirrors social cognitive patterns.

Abeba Birhane, Atoosa Kasirzadeh, David Leslie, and Sandra Wachter. 2023. Science in the age of large language models. *Nat. Rev. Phys.*, 5(5):277–280.

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.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention.

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. *CoRR*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85):2825–2830.

Mohaimenul Raiaan, Md. Saddam Hossain, Kaniz Fatema, Nur Fahad, Sadman Sakib, Most. Marufatul Jannat Mim Mim, Jubaer Ahmad, Mohammed Eunus Ali, and Sami Azam. 2023. A review on large language models: Architectures, applications, taxonomies, open issues and challenges.

Bruce Schneier. 2023. Ai disinformation is a threat to elections learning to spot russian, chinese and iranian meddling in other countries can help the us prepare for 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Stacking the Odds: Transformer-Based Ensemble for AI-Generated Text Detection

Duke Nguyen **Khaing Myat Noe Naing** **Aditya Joshi**
University of New South Wales, Sydney, Australia
{duke.nguyen, khaingmyatnoenaing}@student.unsw.edu.au, aditya.joshi@unsw.edu.au

Abstract

This paper reports our submission under the team name 'SynthDetectives' to the ALTA 2023 Shared Task. We use a stacking ensemble of Transformers for the task of AI-generated text detection. Our approach is novel in terms of its choice of models in that we use accessible and lightweight models in the ensemble. We show that ensembling the models results in an improved accuracy in comparison with using them individually. Our approach achieves an accuracy score of 0.9555 on the official test data provided by the shared task organisers.

1 Introduction

Transformer (Vaswani et al., 2017) is a sequence-to-sequence model that has enabled the training of large language models (LLMs). LLMs such as GPT enable text generation in response to user-defined prompts, allowing for wide applicability. As a result, they have proliferated into several aspects of society, both for good and for bad. Text generated from LLMs, when used unethically, can have several detrimental implications: they can cause widespread fake news, dispense away with all notions of academic honesty and authorship, and threaten to replace human-generated information with AI-generated data at large.

Motivated by these existential concerns, many models have been developed to distinguish AI-generated content from human's. ALTA is participatory in tackling this issue by announcing the ALTA 2023 Shared Task, whose goal is to build 'automatic detection systems that can discriminate between human-authored and synthetic text generated by Large Language Models (LLMs)' (Molla et al., 2013). The text comes from a variety of sources in terms of domains (e.g. medical, law), and source model (e.g. GPT-X, T5). Technically, participating teams are required to build an automated system to solve a binary classification task, distinguishing between human and AI-generated text. Models are evaluated based on robustness and accuracy. There is no requirement on the efficiency and run-time performance. Our team participated in the said shared task. The code is available here[1]. We stack multiple Transformer-based models in an ensemble and show that the ensemble performs better than the individual models. In this paper, we will discuss existing works in the domain, our analysis of the original training data, our proposed pipeline and architecture, our experimental results, and suggested future work.

2 Related Work

AI-generated text detection has a long history. The sources of our AI-generated text are LLMs, which constrain our task to 'authorship attribution (AA) for neural texts', also known as Neural Text Detection (NTD). It is a subclass of the task of binary classification (and sometimes multi-class, when we are detecting the source model). We will summarise briefly the current literature in this domain. Our main source comes from two major surveys by Jawahar et al. (2020) and Uchendu et al. (2023). The latter classifies automated NTD as follows:

Stylometric attribution detects Neural Text Generator (NTG) using ensembles of classical machine learning (ML) models trained on stylometric features such as LIWC (Linguistic Inquiry & Word Count), POS tags, n-grams, Readability score, WritePrints, Empath. These models work best on a small dataset. However, as we increase the data size, they are outperformed by deep learning models rapidly.

Deep learning: GLoVe-based attribution: GLoVe (Pennington et al., 2014) is an unsupervised learning algorithm that aggregates global word-word co-occurrence statistics from text to

---

[1] `https://github.com/dukeraphaelng/synth_detectives`

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 173–178
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

build word representation. GLoVe-based models use these embeddings with RNN and LSTM, which was considered SOTA before BERT (Devlin et al., 2018).

Deep learning: Energy-based attribution: Energy-based models (EBMs) (LeCun et al., 2007) are 'un-normalized generative models' using some energy function to generate high-quality data by modelling the probability distribution of the training data. Adapted for NTD (Bakhtin et al., 2019), they perform well on unseen data, however, they do not scale as well, and are very expensive to train.

Deep learning: Transformer-based attribution: is Transformer-based models fine tuned to perform NTD. These models surpass stylometric and GLoVe-based models and are cheaper than EBMs. RoBERTa and BERT are two models that frequently achieve high performance on NTD benchmarks (Uchendu et al., 2023). Other Transformer-based models that are used in NTD include ELECTRA, XLNet, and DeBERTa. These inspire our choice of weak learners.

Statistical attribution: was developed to combat top-p and top-k decoding strategies which Transformers are not well-equipped against. It has been shown that 'human language is stationary and ergodic as opposed to neural language' (Varshney et al., 2020) suggesting the validity of this approach. Four different algorithms have been proposed which detect AI-generated text through statistical distributions. These are: GLTR (Gehrmann et al., 2019), MAUVE (Pillutla et al., 2021), Distribution detector (Gallé et al., 2021), and DetectGPT (Mitchell et al., 2023), the last three of which perform competitively.

Hybrid attribution: is ensembles using several previously described detectors. These include TDA-based detector (Kushnareva et al., 2021), which extracts attention matrices of BERT's word representations and process them through TDA-based methods as features for a logistic regression model, Fingerprint detector (Diwan et al., 2021), which ensembles fine-tuned RoBERTa embeddings and CNN classifier), FAST (Zhong et al., 2020), which uses RoBERTa with a Graph Neural Network), and CoCo (Liu et al., 2022), a coherence-based contrastive learning model. Our work is an ensemble-based approach to the task. However, we use an ensemble of Transformer models.

id	text	label
0	'Have you ever heard of the Crusades? A time in which Christians went on a 200 year rampage throughout Europe and on their path to Isreal in which they slaughtered innocent people in the name of your God?'	1
4	'The Circuit Court of Appeals of New Jersey had jurisdiction of the controversy between these parties, and its decree was affirmed. But as the court had jurisdiction under the original act of Congress, the jurisdiction in this case was also, under the act of Congress, a bar to the suit.'	0

Table 1: Samples from the training set.

3 Dataset

Three subsets of the dataset are presented: training, validation, and testing. The training set contains 18,000 entries, and the validation and testing each contains 2,000 entries. Evaluation is based on the testing set which was not released until the testing phase of the competition. The training set contains three columns 'id', 'text', and 'label' (1 if human-generated, 0 if AI-generated). The validation and testing set each contains two columns 'id', and 'text'. Samples of the training set are shown in Table 1.

When analyzing the dataset, we find that the AI-generated and human-generated text is evenly split into 9000-9000 entries respectively in the training set. We also find that the average word count per text is relatively low. The mean length is in the 34-35 range in the three subsets, with a standard deviation in the 26.7-27.9 range, a maximum of 172-193, and a minimum of 1, making this a short sequence task.

To find the main domains of the text, we remove all stop words from each set and find the frequency of n-gram phrases from the cleaned corpus, and pick the top-k elements from each set. We look at the n-gram range of $(3, 4)$, with $k = 10$. We find that overwhelmingly all the phrases are in the domain of law across the three sets. The following list is the union of the three sets with the above configuration: {'court of appeals', 'of the court', 'of the united', 'of the united states', 'opinion of the', 'the court of', 'the court of appeals', 'the district court', 'the opinion of', 'the united states'}.

4 Approach

Our approach uses a stacking ensemble of classifiers (as shown in Figure 1) to perform our training, validation, and testing. A stacking ensemble of classifiers acts similarly to a weighted voting classifier. Our choice of architecture is inspired by Maloyan et al. (2022), which achieved high performance in the RuATD Shared Task 2022 on Artificial Text Detection in Russian (Shamardina et al., 2022).

We train each weak classifier using the above dataset split, and then we concatenate the raw predictions on the training set together and feed them to the meta-learner. We use a simple Logistic Regressor as our meta-learner. Our criteria for picking models are ease of use, short-sequence-task-based models, and variety in model architecture. We also choose only encoder-only models, since they are built for regression/ classification tasks, and we can conveniently extract the `[CLS]` token from their last hidden state to perform Logistic Regression. As a result, we use ALBERT, ELECTRA, RoBERTa and XLNet as the Transformer-based models.

To optimise the training cycle, we tokenise the entire dataset (with the respective model's tokeniser), and pass them through their respective pre-trained model to obtain the `[CLS]` token from the last hidden state. We consider this to be our dataset and do our splitting, training, and testing on this processed dataset. To train, we pass the `[CLS]` token through a single fully connected layer, with the input dimension equivalent to the model's `[CLS]`'s dimension, and the output dimension of 2, then we softmax the output. After fine-tuning the weak models, we perform inference on the training split and concatenate the predictions which are fed for the meta-learner to train.

5 Experiment Setup

5.1 Setup

We do not perform any data preprocessing on the dataset. We have a train-validation-test split of 0.8, 0.1, 0.1. All training was done on Google Cloud Platform's Vertex Colab GPU for GCE usage on NVIDIA A100 (40 GB).

5.2 Pipeline

For both our weak learners and our meta-model, we use the AdamW optimiser with the default settings, i.e. $lr = 0.001, \beta = (0.9, 0.999), \epsilon = 1e-08, weight\_decay = 0.01$. All models are trained with $epochs = 300$ and $batch\_size = 128$.

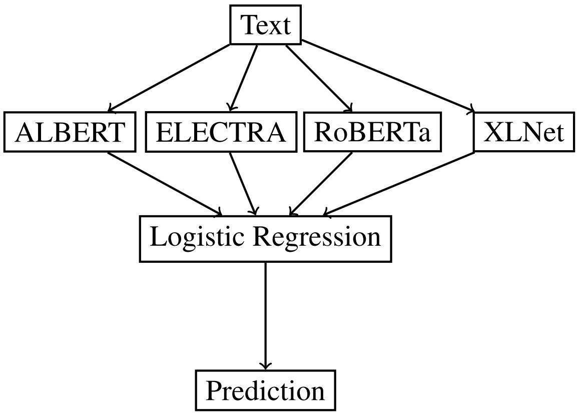


Figure 1: Our Stack Ensembling Architecture for AI-generated text detection.

All models in the ensemble are pre-trained models available on HuggingFace (as of 25th October 2023). For each model, we include their architecture name and the unique HuggingFace model identifier associated with their pre-trained weights.

ALBERT (albert-base-v2) (Lan et al., 2020) is a modification of BERT (Devlin et al., 2018) which reduces its memory consumption and increases the training speed by repeating layers split among groups and splitting the embedding matrix into smaller matrices, whilst being more performative than BERT in GLUE, RACE, and SQuAD.

ELECTRA (google/electra-small-discriminator) (Clark et al., 2020) is another modification of BERT that changes the pretraining objective, as inspired by GAN where ELECTRA acts as the discriminator which predicts whether a token in a randomly masked text is original or generated by the generator (which we train simultaneously). This approach makes ELECTRA perform comparably to larger models whilst using a lot less compute.

RoBERTa (roberta-base) (Liu et al., 2019) optimises BERT in four aspects of training: using full-sentences without Next Sentence Prediction (NSP) loss, with dynamic masking, with larger mini-batches, and with a larger byte-level Byte-Pair Encoding (BPE).

XLNet (xlnet-base-cased) (Yang et al., 2020) uses a generalised autoregressive pretraining method that maximises the 'expected likelihood over all permutations of the input sequence factorisation order' enabling bidirectional contexts and overcoming BERT's pretrain-finetune discrepancy due to neglecting masked positions dependency. XLNet also builds on Transformer-XL (Dai et al., 2019), outperforming BERT on 20 tasks.

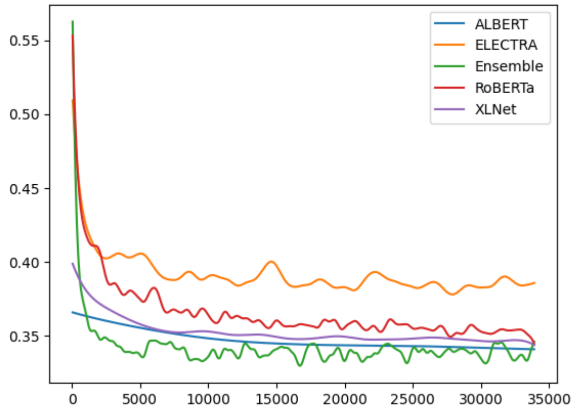


Figure 2: Training Loss.

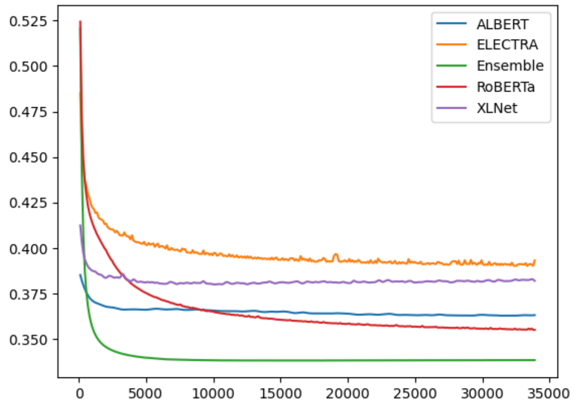


Figure 3: Validation Loss.

Model	Accuracy
ELECTRA	0.9311
XLNet	0.9361
ALBERT	0.9567
RoBERTa	0.9572
Ensemble	0.9694

Table 2: Model Accuracy on the Test Set.

6 Results

Figure 2 shows our smoothed training loss using the Gaussian kernel (since the original training loss displays too wide short-cycle variation, obfuscating the overall trend), Figure 3 shows our validation loss which follows a similar pattern. Table 2 shows our accuracy on the test set as described in the experiment setup. Among our weak learners, RoBERTa performs the best, followed by ALBERT, XLNet, and finally ELECTRA. As expected, our meta-model (Ensemble) outperforms even RoBERTa by more than 0.012. The final testing accuracy model ranking is reflected in the validation loss, and to a lesser extent in the training loss. This agrees with much of the literature indicating that RoBERTa is the best learner in AI-generated text prediction (Jawahar et al., 2020). We also note that XLNet and ALBERT start with extremely low loss, suggesting their pre-training procedure might be conducive to AI-generated text detection.

Finally, the ALTA shared task organisers provided us with a shared task test set *i.e.*, the official test set. We achieve an accuracy of 0.9555 with our stacking ensemble on the official test set.

7 Conclusion & Future Work

In this paper, we describe our system for the ALTA Shared Task 2023. We show how an ensemble of Transformer-based models can be combined using a logistic regression classifier to predict if a text was generated by AI. We achieved an accuracy of 0.9555 using a stacking ensemble of basic encoder-only Transformer models.

Our work presents a novel approach to ensemble Transformer-based models to approach the ALTA shared task. However, this work identifies several potential directions for future work. Ensembling models usually benefit from a variety of learners specialised in different types of inputs. We only implemented an ensemble of Transformer classifiers, but it would be beneficial to integrate other non-Transformer-based weak learners as detailed in Section 2. Especially useful would be to integrate contrastive learning in our training procedure. In addition, it would also be useful to perform data augmentation which can help generalise the model. One suggested technique is ‘text continuation’, where given a human-generated text, we slice the first n words and have an LLM finish the sentence. Furthermore, the scope of the shared task does not imply the possibility of an adversarial attack. It has been shown that the RoBERTa detector can be attacked easily through misspelling (Wolff and Wolff, 2022). It would also be helpful to build detectors that are resilient in this regard.

Acknowledgements

We thank Jake Renzella and the School of Computer Science and Engineering at the University of New South Wales, Sydney for providing us with GPU cloud credits, without which we would not be able to run our experiments.

References

Anton Bakhtin, Sam Gross, Myle Ott, Yuntian Deng, Marc'Aurelio Ranzato, and Arthur Szlam. 2019. Real or fake? learning to discriminate machine from human generated text. *arXiv preprint arXiv:1906.03351*.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

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*.

Nirav Diwan, Tanmoy Chakravorty, and Zubair Shafiq. 2021. Fingerprinting fine-tuned language models in the wild. *arXiv preprint arXiv:2106.01703*.

Matthias Gallé, Jos Rozen, Germán Kruszewski, and Hady Elsahar. 2021. Unsupervised and distributional detection of machine-generated text. *arXiv preprint arXiv:2111.02878*.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander M Rush. 2019. Gltr: Statistical detection and visualization of generated text. *arXiv preprint arXiv:1906.04043*.

Ganesh Jawahar, Muhammad Abdul-Mageed, and Laks Lakshmanan, V.S. 2020. Automatic detection of machine generated text: A critical survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2296–2309, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Laida Kushnareva, Daniil Cherniavskii, Vladislav Mikhailov, Ekaterina Artemova, Serguei Barannikov, Alexander Bernstein, Irina Piontkovskaya, Dmitri Piontkovski, and Evgeny Burnaev. 2021. Artificial text detection via examining the topology of attention maps. *arXiv preprint arXiv:2109.04825*.

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.

Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and Fujie Huang. 2007. Energy-based models. *Predicting structured data*, 1(0).

Xiaoming Liu, Zhaohan Zhang, Yichen Wang, Yu Lan, and Chao Shen. 2022. Coco: Coherence-enhanced machine-generated text detection under data limitation with contrastive learning. *arXiv preprint arXiv:2212.10341*.

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.

Narek Maloyan, , Bulat Nutfullin, Eugene Ilyshin, and and. 2022. DIALOG-22 RuATD generated text detection. In *Computational Linguistics and Intellectual Technologies*. RSUH.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. *arXiv preprint arXiv:2301.11305*.

Diego Molla, Haolan Zhan, Xuanli He, and Qiongkai Xu. 2013. Overview of the 2023 alta shared task: Discriminate between human-written and machine-generated text. In *Proceedings of the 21st Annual Workshop of the Australasian Language Technology Association (ALTA 2023)*.

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.

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

Tatiana Shamardina, Vladislav Mikhailov, Daniil Chernianskii, Alena Fenogenova, Marat Saidov, Anastasiya Valeeva, Tatiana Shavrina, Ivan Smurov, Elena Tutubalina, and Ekaterina Artemova. 2022. Findings of the the RuATD shared task 2022 on artificial text detection in russian. In *Computational Linguistics and Intellectual Technologies*. RSUH.

Adaku Uchendu, Thai Le, and Dongwon Lee. 2023. Attribution and obfuscation of neural text authorship: A data mining perspective. *SIGKDD Explor. Newsl.*, 25(1):1–18.

Lav R Varshney, Nitish Shirish Keskar, and Richard Socher. 2020. Limits of detecting text generated by large-scale language models. In *2020 Information Theory and Applications Workshop (ITA)*, pages 1–5. IEEE.

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.

Max Wolff and Stuart Wolff. 2022. Attacking neural text detectors.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2020. Xlnet: Generalized autoregressive pretraining for language understanding.

Wanjun Zhong, Duyu Tang, Zenan Xu, Ruize Wang, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. 2020. Neural deepfake detection with factual structure of text. *arXiv preprint arXiv:2010.07475*.

Natural Language Processing for Clinical Text

Vlada Rozova[§]
Centre for Digital Transformation of Health
University of Melbourne
Parkville VIC 3052
AUSTRALIA
vlada.rozova@unimelb.edu.au

Jinghui Liu[§]
Australian e-Health Research Centre
CSIRO
Herston QLD 4029
AUSTRALIA
jinghui.liu@csiro.au

Mike Conway
School of Computing & Information Systems
University of Melbourne
Parkville VIC 3052
AUSTRALIA
mike.conway@unimelb.edu.au

1 Introduction

Learning from real-world clinical data has potential to promote the quality of care, improve the efficiency of healthcare systems, and support clinical research. As a large proportion of clinical information is recorded only in unstructured free-text format, applying NLP to process and understand the vast amount of clinical text generated in clinical encounters is essential. However, clinical text is known to be highly ambiguous, it contains complex professional terms requiring clinical expertise to understand and annotate, and it is written in different clinical contexts with distinct purposes. All these factors together make clinical NLP research both rewarding and challenging.

In this tutorial, we will discuss the characteristics of clinical text and provide an overview of some of the tools and methods used to process it. We will also present a real-world example to show the effectiveness of different NLP methods in processing and understanding clinical text. Finally, we will discuss the strengths and limitations of large language models and their applications, evaluations, and extensions in clinical NLP.

2 Learning Objectives

This three hour tutorial has several related learning objectives:

1. Develop insight into the range of clinical text data available
2. Develop insights into a range of clinical NLP application areas
3. Understand the landscape of methods used in clinical NLP
4. Identify potential obstacles associated with working with clinical text
5. Understand privacy, legal, and ethical issues associated with working with clinical text
6. Understand publication practices in clinical NLP

Note that given regulatory constraints and ethical sensitivities regarding the sharing of clinical data, we are unable to distribute clinical corpora discussed in this session to tutorial participants.

3 Target Audience and Prerequisites

This tutorial targets NLP researchers (students and more experienced researchers) with an interest in, or curiosity about working with clinical text. The tutorial is designed to be accessible for anyone with an interest in NLP.

4 Outline

The tutorial consists of three consecutive one hour sessions, described below.

4.1 Introduction to Clinical NLP

The first session will introduce the broad area of clinical NLP, focusing on the special characteristics of clinical text and some of the challenges associated with the application of NLP methods to clinical notes (Nadkarni et al., 2011; Dalianis,

---

[§] The first two authors contributed equally to this work.

Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association, pages 179–182
November 29 - December 1, 2023 ©2023 Association for Computational Linguistics

2018; Wang et al., 2018). First, we will describe the role of linguistic variation and technical clinical vocabularies in clinical text, particularly regarding issues related to polysemy, synonymy, misspellings, and acronyms. Second, we will discuss the importance of contextual attributes in the context of clinical information extraction, particularly negation, uncertainty detection, and temporality detection. Third, we will discuss typical processes involved in developing clinical NLP systems, including challenges related to corpus development and annotation. Fourth, we will briefly outline some of the major clinical NLP datasets available for research. Fifth, we will summarise some of the regulatory, legal, and ethical issues related to clinical NLP, with a particular focus on privacy protection. Finally, we will make some brief comments regarding publication practices and grant funding in clinical NLP.

Suggested Reading

1. Dalianis (2018). Clinical Text Mining: Secondary Use of Electronic Patient Records. *Springer* (Dalianis, 2018)
2. Nadkarni et al. (2011) Natural Language Processing: an introduction. *Journal of the American Medical Informatics Association* (Nadkarni et al., 2011)
3. Lederman et al. (2022). Tasks as needs: reframing the paradigm of clinical natural language processing research for real-world decision support. *Journal of the American Medical Informatics Association* (Lederman et al., 2022)

4.2 Clinical NLP in Practice

In the second session, we will compare several approaches to named-entity recognition (NER) using real-world data. For this, we will use a small dataset of 283 pathology reports from The Royal Melbourne Hospital and Peter MacCallum Cancer Centre, Melbourne, Australia (Rozova et al., 2023). Phrases in the reports were annotated for invasive fungal infection (IFI), a rare but dangerous condition for immunocompromised patients.

We will start by exploring the dataset: we will look at the reports themselves to see if there is any structure that we could leverage in our analysis. The audience will be presented with a report and asked to determine what information is relevant to IFI. We will take note of specific terminology, the importance of negation and context dependency.

Next, we will look into the provided manual annotations and run summary statistics noting the number of concept categories, how common each category is and its lexical diversity. Based on this information, we will discuss what performance can be reasonably expected from a NER model.

Finally, we will compare three common approaches to NER: a simple dictionary-based approach, conditional random fields (CRF), and BERT, a transformer-based model (Devlin et al., 2018). We will consider the strengths and weaknesses of each approach, especially given the application context. We will then compare the performance of the models and discuss what additional steps can be undertaken for future improvement.

Suggested Reading

1. Liu and Panagiotakos (2022) Real-world data: a brief review of the methods, applications, challenges and opportunities. *BMC Medical Research Methodology* (Liu and Panagiotakos, 2022)
2. Velupillai et al. (2018) Using clinical Natural Language Processing for health outcomes research: overview and actionable suggestions for future advances. *Journal of Biomedical Informatics* (Velupillai et al., 2018)

4.3 Large Language Models & Clinical NLP

The third session introduces the use of Large Language Models (LLMs) in the context of clinical NLP, primarily focusing on their applications, domain adaptation, and evaluation. We first discuss the categories of LLMs by considering encoders and decoders and how they are applied to various clinical NLP tasks. For encoders, we introduce using the models for standard NLP tasks involving clinical text and clinical prediction tasks at the point of care (Lewis et al., 2020; Jiang et al., 2023). For decoders, we discuss the applications enabled by the general-domain LLMs (Lee et al., 2023; Thirunavukarasu et al., 2023) such as medical question answering (Singhal et al., 2023) and zero- and few-shot learning (Agrawal et al., 2022). Then we show whether adaptation to the clinical domain is still necessary for LLMs that have already been pretrained on vast amounts of general-domain text by summarising relevant results from recent work (Lehman et al., 2023). We go on to discuss evaluation issues relevant for LLMs in the clinical context, as the application of clinical LLMs extends beyond mere predictive accuracy. We talk about the other perspectives that need to be considered when measuring the effectiveness and usefulness of LLMs for healthcare (Wornow et al., 2023).

In addition to this core content, we briefly touch on other related topics surrounding LLMs for clinical applications, including multimodal modelling, retrieval-augmented generation (RAG), and implementation issues in the clinical context. For multimodal modelling, we discuss the interaction between various modalities from patient data, such as text, image, and structured data, and how LLMs enable new modelling approaches (Moor et al., 2023). For RAG, we talk about its potential benefits in the clinical setting, such as for open-ended QA (Zakka et al., 2023). We also discuss issues and challenges in implementing and monitoring current LLMs in the clinical environment (Finlayson et al., 2021).

Suggested Reading

Suggested readings for this section include:

1. Lehman et al. (2023) Do we still need clinical language models? *Proceedings of the Conference on Health, Inference, and Learning*. (Lehman et al., 2023)
2. Thirunavukarasu et al. (2023) Large language models in medicine. *Nature Medicine* (Thirunavukarasu et al., 2023)
3. Wornow et al. (2023) The shaky foundations of large language models and foundation models for electronic health records. *NPJ Digital Medicine* (Wornow et al., 2023)

5 Presenter Information

Vlada Rozova is a Postdoctoral Research Fellow with the Centre for Digital Transformation of Health at the University of Melbourne. She is a data scientist and a machine learning practitioner passionate about developing automated systems that can facilitate clinical decision-making. Vlada works with stakeholders of diverse backgrounds to build solutions that address user needs and enjoys seeing the development and implementation of tools from start to end.[1]

Jinghui Liu is a Postdoctoral Research Fellow at the Australian e-Health Research Centre, Commonwealth Scientific and Industrial Research Organisation (CSIRO). He is interested in studying and applying natural language processing and machine learning techniques to healthcare data and how these models can contribute to realising the potential of digital health.[2]

Mike Conway is a Senior Lecturer in Digital Health at the University of Melbourne's School of Computing & Information Systems and the Centre for Digital Transformation of Health. His research interests are centred on the application of computational methods — particularly natural language processing — to public health research questions, with much of his research output focused on mental health and substance use.[3]

Ethics Statement

While we do not anticipate any specific ethical concerns arising directly from this tutorial, there are a number of more general ethical issues associated with NLP that are particularly acute with respect to clinical NLP. These issues include *dual use* (NLP-supported epidemiological studies can be used to identify and support at-risk groups in the community, but could also be used to stigmatise these same groups); *bias* (NLP models trained on existing clinical text may amplify existing biases); *privacy* (NLP algorithms may risk compromising the privacy of individuals with rare medical conditions), and *reproducibility* (there is some tension between the need to protect patient privacy and the ethical imperative to support reproducibility via data sharing).

Acknowledgements

We would like to thank Professor Wendy Chapman (Centre for Digital Transformation of Health, University of Melbourne) for providing valuable teaching materials for this tutorial. The authors did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors to support the activities described in this paper.

References

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large language models are few-shot clinical information extractors. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1998–2022, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

---

[1] `https://scholar.google.com/citations?user=3P5aMMcAAAAJ&hl=en`

[2] `https://people.csiro.au/l/j/jinghui-liu`

[3] `https://maconway.github.io/`

Hercules Dalianis. 2018. *Clinical Text Mining: Secondary Use of Electronic Patient Records*. Springer International Publishing.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Samuel G Finlayson, Adarsh Subbaswamy, Karandeep Singh, John Bowers, Annabel Kupke, Jonathan Zittrain, Isaac S Kohane, and Suchi Saria. 2021. The clinician and dataset shift in artificial intelligence. *The New England journal of medicine*, 385(3):283–286.

Lavender Yao Jiang, Xujin Chris Liu, Nima Pour Nejatian, Mustafa Nasir-Moin, Duo Wang, Anas Abidin, Kevin Eaton, Howard Antony Riina, Ilya Laufer, Paawan Punjabi, Madeline Miceli, Nora C Kim, Cordelia Orillac, Zane Schnurman, Christopher Livia, Hannah Weiss, David Kurland, Sean Neifert, Yosef Dastagirzada, Douglas Kondziolka, Alexander T M Cheung, Grace Yang, Ming Cao, Mona Flores, Anthony B Costa, Yindalon Aphinyanaphongs, Kyunghyun Cho, and Eric Karl Oermann. 2023. Health system-scale language models are all-purpose prediction engines. *Nature*.

Asher Lederman, Reeva Lederman, and Karin Verspoor. 2022. Tasks as needs: reframing the paradigm of clinical natural language processing research for real-world decision support. *Journal of the American Medical Informatics Association: JAMIA*, 29(10):1810–1817.

Peter Lee, Sebastien Bubeck, and Joseph Petro. 2023. Benefits, limits, and risks of GPT-4 as an AI chatbot for medicine. *The New England journal of medicine*, 388(13):1233–1239.

Eric Lehman, Evan Hernandez, Diwakar Mahajan, Jonas Wulff, Micah J Smith, Zachary Ziegler, Daniel Nadler, Peter Szolovits, Alistair Johnson, and Emily Alsentzer. 2023. Do we still need clinical language models? In *Proceedings of the Conference on Health, Inference, and Learning*, volume 209 of *Proceedings of Machine Learning Research*, pages 578–597. PMLR.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020. Pretrained language models for biomedical and clinical tasks: Understanding and extending the State-of-the-Art. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 146–157, Online. Association for Computational Linguistics.

Fang Liu and Demosthenes Panagiotakos. 2022. Real-world data: a brief review of the methods, applications, challenges and opportunities. *BMC Medical Research Methodology*, 22(287).

Michael Moor, Qian Huang, Shirley Wu, Michihiro Yasunaga, Cyril Zakka, Yash Dalmia, Eduardo Pontes Reis, Pranav Rajpurkar, and Jure Leskovec. 2023. Med-Flamingo: a multimodal medical few-shot learner. *arXiv*.

Prakash M Nadkarni, Lucila Ohno-Machado, and Wendy W Chapman. 2011. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5):544–551.

Vlada Rozova, Anna Khanina, Jasmine Teng, Joanne Teh, Leon Worth, Monica Slavin, Karin Thursky, and Karin Verspoor. 2023. Cytology and histopathology invasive fungal infection reports (version 1.0.1). *PhysioNet*.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Abubakr Babiker, Nathanael Schärli, Aakanksha Chowdhery, Philip Mansfield, Dina Demner-Fushman, Blaise Agüera Y Arcas, Dale Webster, Greg S Corrado, Yossi Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle Barral, Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. 2023. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180.

Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. 2023. Large language models in medicine. *Nature medicine*, 29:1930–1940.

Sumithra Velupillai, Hanna Suominen, Maria Liakata, Angus Roberts, Anoop D. Shah, Katherine Morley, David Osborn, Joseph Hayes, Robert Stewart, Johnny Downs, Wendy Chapman, and Rina Dutta. 2018. Using clinical natural language processing for health outcomes research: Overview and actionable suggestions for future advances. *Journal of Biomedical Informatics*, 88:11–19.

Yanshan Wang, Liwei Wang, Majid Rastegar-Mojarad, Sungrim Moon, Feichen Shen, Naveed Afzal, Sijia Liu, Yuqun Zeng, Saeed Mehrabi, Sunghwan Sohn, et al. 2018. Clinical information extraction applications: a literature review. *Journal of biomedical informatics*, 77:34–49.

Michael Wornow, Yizhe Xu, Rahul Thapa, Birju Patel, Ethan Steinberg, Scott Fleming, Michael A Pfeffer, Jason Fries, and Nigam H Shah. 2023. The shaky foundations of large language models and foundation models for electronic health records. *NPJ digital medicine*, 6(1):135.

Cyril Zakka, Akash Chaurasia, Rohan Shad, Alex R Dalal, Jennifer L Kim, Michael Moor, Kevin Alexander, Euan Ashley, Jack Boyd, Kathleen Boyd, Karen Hirsch, Curt Langlotz, Joanna Nelson, and William Hiesinger. 2023. Almanac: Retrieval-Augmented language models for clinical medicine. *arXiv*.

Author Index

Ackerman, Mark S., 112
Aishi, Tanusree Das, 1, 169

Beck, Daniel, 125
Brown, Nathan J., 78
Buscaldi, Davide, 107

C-LARA-Instance, ChatGPT, 20
Chang, Chia-Hui, 11
Chen, Ming-Bin, 136
Chiera, Belinda, 20
Chu, Kevin, 78
Chua, Cathy, 20
Clermont, Frantz, 40
Conway, Mike, 179

El-Sayed, Ahmed, 164

Fang, Yunhao, 159
Fataliyev, Kamaladdin, 67
Fataliyev, Steven Nguyen, 125
Frermann, Lea, 136

Gagiano, Rinaldo, 153
Gouravajhala, Sai R., 112

Haffari, Gholamreza, 88
He, Xuanli, 148
Holtta-Otto, Katja, 125
Huang, Yishan, 100

Ishihara, Shunichi, 40

Joshi, Aditya, 173
Joy, Saman Sarker, 1, 169

Koopman, Bevan, 78
Kummerfeld, Jonathan K., 112

Lau, Jey Han, 136
Lee, Yu-Kai, 11
Li, Zihan, 112
Liu, Jinghui, 78, 118, 179
Liu, Wei, 67
Liyanage, Vijini, 107

Maruf, Sameen, 88
Meng, Yang, 130
Molla, Diego, 30, 148

Naing, Khaing Myat Noe, 173
Nasr, Omar, 164
Ng, King Tao Jason, 30
Nguyen, Anthony, 78, 118
Nguyen, Duke, 173
Nodi, Naima Tahsin, 1

Qu, Lizhen, 88

Raheb, Chadi, 20
Rasel, Annajiat Alim, 1
Rayner, Manny, 20
Reid, Kathy, 51
Rozova, Vlada, 130, 179
Rusak, Filip, 78

Shi, Yiming, 112
Simonsen, Annika, 20

Tian, Lin, 153

Vernier, Andrew M., 112
Verspoor, Karin, 130

Wang, Yufei, 88
Williams, Elizabeth T., 51

Xiang, Zhengkang, 20
Xu, Qiongkai, 148

Zhan, Haolan, 88, 148
Zukerman, Ingrid, 88
Zviel-Girshin, Rina, 20