UNIVERSITY OF CALIFORNIA SAN DIEGO

Towards Human-Centered NLP Systems: Trustworthiness, Cognition, and Social Good

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy

in

Computer Science

by

Zexue He

Committee in charge:

Professor Julian McAuley, Chair
Professor Taylor Berg-Kirkpatrick
Professor Zhiting Hu
Professor Jingbo Shang

2024

The Dissertation of Zexue He is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2024

DEDICATION

To those whose love and support light my path.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my Ph.D. supervisor, Prof. Julian McAuley, for his unwavering support throughout my Ph.D. journey. Julian is, without a doubt, the best supervisor I could have hoped for. His unique research perspective, which is essential and inspiring, combined with his professional collaboration style, helps every one of my projects forward. His invaluable mentorship has shaped me not only into a better researcher but also into a better person. Reflecting on the past five years in his lab, I feel an immense sense of trust, support, respect, and freedom, which provided a solid foundation for my current and future research journey. Julian has everything I could have wished for in an advisor.

I am also profoundly grateful to my dissertation committee members, Prof. Taylor Berg-Kirkpatrick, Prof. Jingbo Shang, and Prof. Zhiting Hu, for their essential feedback on my research. They are close collaborators from whom I have learned not only the art of conducting research but also valuable life lessons.

I owe my deepest thanks to my family. My parents have believed in me every step of the way, showing their support and encouragement, and standing by me through countless days and nights, even from afar (I know how much they wished they could be here). They are my unwavering strength and confidence to forge ahead. A special, heartfelt thank you goes to my husband, Yuheng Zhi. We met at the start of this Ph.D. journey when we were still young, and gratefully nothing has come between us as we reach this milestone. Yuheng is not only my beloved partner but also my truest friend and most dependable teammate. We have grown together, both in our research and in life. I look forward to embracing all the adventures the future holds together with you.

Throughout my Ph.D., I have had the honor of working with many talented collaborators, and I would like to extend my gratitude to all of them for their contributions: Chun-Nan Hsu (UCSD), Rogerio Feris (MIT-IBM), Fereshte Khani (OpenAI), Marco Tulio Ribeiro (Google), Mengting Wan (Microsoft), Brent Hecht (Microsoft and Northwestern University), Jennifer Neville (Microsoft and Purdue University), Longqi Yang (Microsoft), Rameswar Panda (MIT-

IBM), Graeme Blackwood (IBM), Jianmo Ni (Google), Wang-Cheng Kang (Google), Derek Cheng (Google), Amilcare Gentili (UCSD), Bodhisattwa Majumder (AI2), Zhankui He (Google), An Yan (Salesforce), Yu Wang (UCSD), Yao Liu (CMU), Jessica Echterhoff(UCSD), Canwen Xu (SnowFlake), Jiacheng Li (Meta), Shuyang Li (Meta), Jiarui Jin (SJTU), Abeer Alessa (UCSD). I also wish to thank the mentors and professors who guided my research before my Ph.D.: Kan Ren (ShanghaiTech University) Yiqun Liu (THU), Baobao Chang (PKU), Yanlin Luo (BNU), Haohan Wang (UIUC), and Zachary Lipton (CMU). They opened the door to research and laid a strong foundation for my Ph.D. studies.

I extend my thanks to my co-authors for their permission to include our publications and materials in my dissertation:

Chapter 2, in part, is a reprint of the material as it appears in "Targeted Data Generation: Finding and Fixing Model Weaknesses" by Zexue He, Marco Tulio Ribeiro, and Fereshte Khani, in proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 3, in part, is a reprint of the material as it appears in "Controlling Bias Exposure for Fair Interpretable Predictions" by Zexue He, Yu Wang, Julian McAuley, and Bodhisattwa Prasad Majumder, in Findings of the Association for Computational Linguistics: EMNLP 2022, pp. 5854-5866, 2022. The dissertation author was the primary investigator and author of this paper.

Chapter 4, in part, is a reprint of the material as it appears in "InterFair: Debiasing with Natural Language Feedback for Fair Interpretable Predictions" by Bodhisattwa Prasad Majumder*, Zexue He*, Julian McAuley, in Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 5, in part, is a reprint of the material as it appears in "Detect and Perturb: Neutral Rewriting of Biased and Sensitive Text via Gradient-based Decoding" by Zexue He, Bodhisattwa

Prasad Majumder, and Julian McAuley, in Findings of the Association for Computational Linguistics: EMNLP 2021, pp. 4173-4181. 2021. The dissertation author was the primary investigator and author of this paper.

Chapter 6, in part, is a reprint of the material as it appears in "Synthetic Pre-training Tasks For Neural Machine Translation" by Zexue He*, Graeme Blackwood*, Rameswar Panda, Julian McAuley, and Rogerio Feris, in Findings of the Association for Computational Linguistics: ACL 2023, pp. 8080-8098, 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 7, in part, is a reprint of the material as it appears in "Cognitive Bias in Decision-making with LLMs" by Echterhoff, Jessica, Yao Liu, Abeer Alessa, Julian McAuley, and Zexue He, in the Findings of the 2023 Conference on Empirical Methods in Natural Language Processing, 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 8, in part, is a reprint of the material as it appears in "CAMELoT: Towards Large Language Models with Training-Free Consolidated Associative Memory" by Zexue He, Leonid Karlinsky, Donghyun Kim, Julian McAuley, Dmitry Krotov, and Rogerio Feris, in Workshop on Long-Context Foundation Models at International Conference on Machine Learning, 2024. The dissertation author was the primary investigator and author of this paper.

Chapter 9, in part, is a reprint of the material as it appears in "MedEval: A Multi-Level, Multi-Task, and Multi-Domain Medical Benchmark for Language Model Evaluation" by Zexue He, Yu Wang, An Yan, Yao Liu, Eric Chang, Amilcare Gentili, Julian McAuley, and Chun-nan Hsu, in Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 8725-8744, 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 10, in part, is a reprint of the material as it appears in ""Nothing abnormal": Disambiguating medical reports via contrastive knowledge infusion" by Zexue He, An Yan, Amilcare Gentili, Julian McAuley, and Chun-Nan Hsu in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, no. 12, pp. 14232-14240, 2023. The dissertation author was

the primary investigator and author of this paper.

Final thanks to myself for not giving up. I began my Ph.D. study at the age of knowing nothing, and now, it marks the end of the long and challenging journey. Along the way, I have had lots of troubles, but I decided to write jolly tales. May our future be as brilliant as the stars and the sea.

VITA

2015–2019 B.S., Beijing Normal University

2020–2022 M.S., University of California San Diego

2020–2024 Ph.D., University of California San Diego

PUBLICATIONS

He, Zexue, Bodhisattwa Prasad Majumder, and Julian McAuley. "Detect and Perturb: Neutral Rewriting of Biased and Sensitive Text via Gradient-based Decoding." In Findings of the Association for Computational Linguistics: EMNLP 2021, pp. 4173-4181. 2021.

He, Zexue, Yu Wang, Julian McAuley, and Bodhisattwa Prasad Majumder. "Controlling Bias Exposure for Fair Interpretable Predictions." In Findings of the Association for Computational Linguistics: EMNLP 2022, pp. 5854-5866. 2022.

He, Zexue, An Yan, Amilcare Gentili, Julian McAuley, and Chun-Nan Hsu. ""Nothing abnormal": Disambiguating medical reports via contrastive knowledge infusion." In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, no. 12, pp. 14232-14240. 2023.

He, Zexue, Marco Tulio Ribeiro, and Fereshte Khani. "Targeted Data Generation: Finding and Fixing Model Weaknesses." In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 8506-8520. 2023.

He, Zexue, Yu Wang, An Yan, Yao Liu, Eric Y. Chang, Amilcare Gentili, Julian McAuley, and Chun-Nan Hsu. "MedEval: A Multi-Level, Multi-Task, and Multi-Domain Medical Benchmark for Language Model Evaluation." In The 2023 Conference on Empirical Methods in Natural Language Processing.

He, Zexue*, Graeme Blackwood*, Rameswar Panda, Julian McAuley, and Rogerio Feris. "Synthetic Pre-Training Tasks for Neural Machine Translation." In Findings of the Association for Computational Linguistics: ACL 2023, pp. 8080-8098. 2023.

Majumder, Bodhisattwa*, **Zexue He***, and Julian McAuley. "InterFair: Debiasing with Natural Language Feedback for Fair Interpretable Predictions." In The 2023 Conference on Empirical Methods in Natural Language Processing.

Jin, Jiarui*, **Zexue He***, Mengyue Yang, Weinan Zhang, Yong Yu, Jun Wang, and Julian McAuley. "InfoRank: Unbiased Learning-to-Rank via Conditional Mutual Information Minimization." In Proceedings of the ACM on Web Conference 2024, pp. 1350-1361. 2024.

He, Zexue, Leonid Karlinsky, Donghyun Kim, Julian McAuley, Dmitry Krotov, and Rogerio Feris. "CAMELoT: Towards Large Language Models with Training-Free Consolidated Associative Memory." In ICML 2024 Workshop on Long Context Foundation Models.

Xu, Canwen, **Zexue He**, Zhankui He, and Julian McAuley. "Leashing the inner demons: Self-detoxification for language models. " In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, no. 10, pp. 11530-11537. 2022.

Yan, An, **Zexue He**, Xing Lu, Jiang Du, Eric Chang, Amilcare Gentili, Julian McAuley, and Chun-nan Hsu. "Weakly Supervised Contrastive Learning for Chest X-Ray Report Generation." In Findings of the Association for Computational Linguistics: EMNLP 2021, pp. 4009-4015. 2021.

Wang, Yu, **Zexue He**, Zhankui He, Hao Xu, and Julian McAuley. "Deciphering Compatibility Relationships with Textual Descriptions via Extraction and Explanation." In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, no. 8, pp. 9133-9141. 2024.

Wang, Haohan, **Zexue He**, Zachary C. Lipton, and Eric P. Xing. "Learning Robust Representations by Projecting Superficial Statistics Out." In International Conference on Learning Representations.

Yan, An, Yu Wang, Yiwu Zhong, Chengyu Dong, **Zexue He**, Yujie Lu, William Yang Wang, Jingbo Shang, and Julian McAuley. "Learning concise and descriptive attributes for visual recognition." In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3090-3100. 2023.

Shi, Taiwei, Zhuoer Wang, Longqi Yang, Ying-Chun Lin, **Zexue He**, Mengting Wan, Pei Zhou et al. "WildFeedback: Aligning LLMs With In-situ User Interactions And Feedback." In NeurIPS 2024 Workshop on Behavioral Machine Learning.

Echterhoff, Jessica, Yao Liu, Abeer Alessa, Julian McAuley, and **Zexue He**. "Cognitive bias in high-stakes decision-making with llms." In Findings of the Association for Computational Linguistics: EMNLP 2024.

ABSTRACT OF THE DISSERTATION

Towards Human-Centered NLP Systems: Trustworthiness, Cognition, and Social Good

by

Zexue He

Doctor of Philosophy in Computer Science

University of California San Diego, 2024

Professor Julian McAuley, Chair

In the era of generative artificial intelligence (GenAI), people increasingly rely on these AI-powered systems for daily tasks, ranging from conversational chatbots to healthcare assistance and beyond. Despite their widespread adoption, AI or natural language processing (NLP) models face critical challenges, such as their vulnerabilities to robustness issues across diverse groups, biased behaviors, and harmful outputs. These challenges raise significant concerns about their reliability and real-world applicability, emphasizing the urgent need for human-centered NLP – systems designed to prioritize human values, trust, and socially beneficial outcomes.

This dissertation explores three core aspects of human-centered NLP. First, it addresses the trustworthiness of NLP systems, especially the large language models (LLMs), by examining

critical concerns about their reliability and presenting strategies to enhance their robustness and trustworthiness. Second, it introduces a novel perspective on learning from humans, emphasizing the importance of understanding, modeling, and drawing inspiration from human cognition to align NLP systems more closely with human reasoning and behavior. Third, it highlights the impact of human-centered NLP in socially beneficial applications, such as improving patient care and outcomes in healthcare. By addressing critical challenges and integrating insights from interdisciplinary fields, this dissertation aims to pave a path toward NLP systems that not only perform effectively but also respect human values and advance social good, thereby laying the groundwork for the next generation of responsible, human-centered NLP technologies.

Chapter 1

Introduction

In recent years, natural language processing (NLP) and artificial intelligence models have achieved remarkable success across a wide range of applications, including chatbots like ChatGPT [OpenAI, 2024], machine translators [Zhu et al., 2024], and automatic text summarization [Zhang et al., 2024, Laskar et al., 2023, Chang et al., 2024]. At the core of these advancements are generative models such as large language models (LLMs), developed through a rigorous multi-step process: (1) large-scale data collection, often involving trillions of tokens sourced from web text crawled across the internet; (2) extensive model pre-training on the collected data; and (3) fine-tuning with reinforcement learning from human feedback (RLHF) based on curated human preference data. These steps collectively empower LLMs to interact effectively with users, enabling more natural and context-aware language understanding and generation.

Despite their impressive achievements, LLMs often lack robust performance, particularly when dealing with underrepresented data groups. For instance, a man was wrongly arrested by Arab police after Facebook AI mistranslated "يصبحهم" (a phrase in a low-resource language

meaning "good morning" in English) as “attack them” [1]. This example highlights how models that perform well on majority languages or data groups often fail to generalize robustly to less common or low-resource contexts. In addition to performance issues, LLMs can exhibit irresponsible behaviors due to problematic training data, which may contain explicit or implicit biases and harmful content. These biases can lead to toxic outputs [Xu et al., 2022a, Gehman et al., 2020] and biased or unfair responses toward certain demographic groups [He et al., 2022, Sheng et al., 2019]. For example, LLMs have produced offensive language or perpetuated stereotypes in political discussions [Heikkilä, 2023], raising serious ethical concerns. Furthermore, privacy risks arise when LLMs inadvertently generate sensitive personal information present in their training data [Xu et al., 2019]. Such failures highlight the risks caused by inaccurate, biased and other harmful behaviors embedded in these models, particularly in sensitive domains like healthcare and education, where equitable treatment is paramount.

These challenges underscore the urgent need to address the research question: how can we deploy NLP models **responsibly** so that they can serve as beneficial assistive tools in daily life? To tackle these issues, researchers are increasingly advocating for **human-centered NLP**, defined as the design and development of NLP systems *that prioritize the needs and preferences of human users while carefully considering ethical and social implications.*

Many recent studies have sought to design human-centered NLP systems, focusing primarily on addressing ethical concerns and aligning models with user needs. However, these approaches often overlook a crucial aspect: the importance of understanding and modeling human cognition as a foundation for system development. This dissertation addresses this gap by extending the concept of human-centered NLP – in addition to ethical considerations and user alignment, we argue that incorporating insights into human mental processes and behaviors is equally critical for guiding the design and functionality of NLP systems. This leads to an enriched definition of human-centered NLP, which is built on three foundational principles, as

[1] https://www.theguardian.com/technology/2017/oct/24/facebook-palestine-israel-translates-good-morning-at\tack-them-arrest

illustrated in Figure 1.1.

1. **Preventing, Guarding, and Intervening in NLP Models**: Enhancing trustworthiness requires proactive measures to prevent harmful behaviors, guard against risks, and intervene to address issues when they occur.

2. **Leveraging Human Cognition for Better NLP Models**: Understanding human reasoning and cognitive processes is just as vital as model design. By integrating these insights into system development, we can create NLP systems that align more naturally with human thought processes and behaviors, resulting in improved performance and reliability.

3. **Aiming for Real-World Positive Impact**: The ultimate goal is to move beyond technical benchmarks and prioritize applications that deliver tangible benefits to society, particularly in critical domains such as healthcare, education, and social well-being (i.e., Social Good).

Together, these principles form the foundation of my research approach to human-centered NLP, striving to create AI systems that not only perform robustly but also reliably align with human and society values.

1.1 Robust and Responsible Human-Centered NLP

This dissertation delves into these pillars and examines each in detail, exploring their individual contributions to the broader goal of human-centered NLP.

Trustworthiness Enhancing the trustworthiness of large-scale NLP models is a key focus of my research. This involves detecting and mitigating issues across several critical dimensions:

- **Robustness.** Trustworthy NLP systems must demonstrate consistent performance across various domains, resist adversarial attacks, and maintain effectiveness under shifting data distributions.

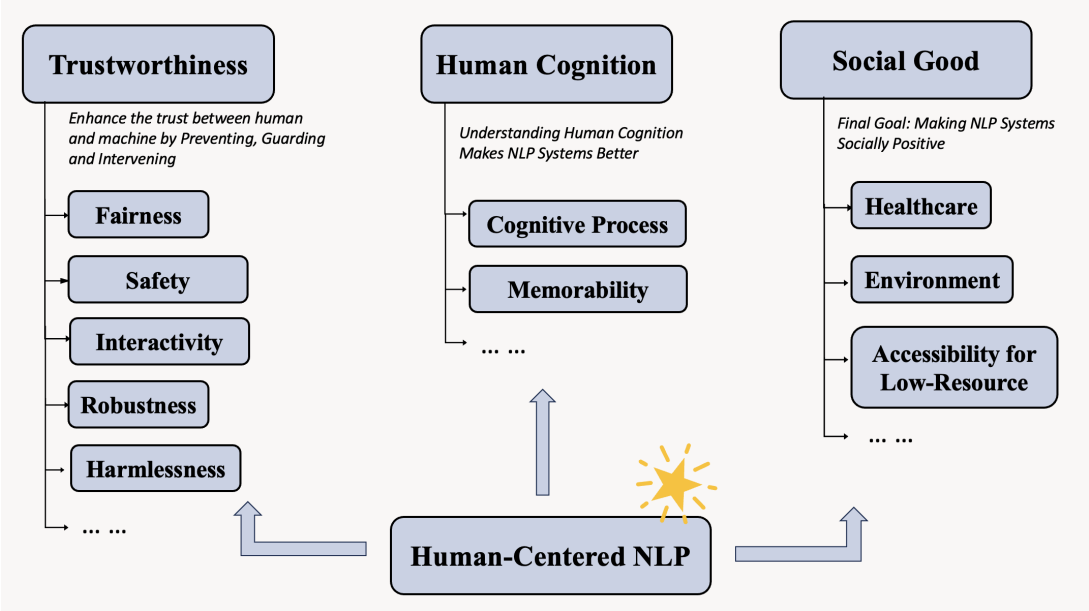


Figure 1.1. Our research framework towards human-centered NLP systems.

- **Fairness.** Building trust between humans and machines requires not only robust performance but also fairness – ensuring that outcomes are equitable across different groups, irrespective of attributes such as race, gender, or age.

- **Interpretability.** Trust can only be achieved when the decision-making processes of an NLP model are transparent and understandable to humans. My research addresses this by promoting explainability and clarity in AI-powered systems.

- **Interactivity.** Effective and safe real-time interaction between NLP systems and humans is crucial for fostering trust and user satisfaction.

- **Harmlessness.** Preventing harmful outputs, such as toxic language and hate speech, is a core aspect of my work on ensuring benign NLP systems.

- **Safety.** Protecting user privacy and safeguarding sensitive information is essential for building trustworthy NLP applications.

While much of the existing work addresses these criteria in isolation, my research

emphasizes their inter-connectedness, arguing that they must work together to truly enhance the trustworthiness of NLP systems.

Human Understanding & Modeling Trustworthy NLP systems cannot be achieved through model design alone; a deeper understanding of human cognition and interactions with AI is equally essential. This requires interdisciplinary research, bridging cognitive science, neuroscience, and human-AI interaction. My research explores two critical dimensions of human understanding and integrates these insights into NLP system design:

- **Cognitive Procedures in Human Decision-Making**: Understanding individual cognitive processes can enhance communication between natural and artificial systems. This alignment fosters more intuitive and effective interactions.
- **Memorability of Human Brain**: The ability to predict whether a novel event will be remembered or forgotten is one of the uniqueness of human brains. People tend to remember and forget similar types of stimuli—images, faces, words, and graphs. My research develops computational models capable of predicting such memorability during or even before the encoding of an event. These memory-inspired algorithms have broad societal applications, including enhancing long-term modeling for medical diagnostics, education, and compensatory tools for cognitive impairments.

This dissertation covers two papers in this direction. One identifies instances where LLMs exhibit cognitive biases similar to those seen in humans (e.g., anchoring bias) when used as interactive assistants in high-stakes decision-making scenarios [Echterhoff et al., 2024]. The other one designs novel Transformer architectures with memory modules inspired by the human brain's memorization mechanisms to efficiently capture long-range language dependencies [He et al., 2024].

AI for Social Good The ultimate goal of building human-centered NLP systems is to make them responsibly assistive in daily life, addressing critical global challenges, promoting positive

social impact, and improving the well-being of individuals and communities. This vision aligns with the broader concept of AI for Social Good. My research explores several key areas within this domain:

- **AI for Healthcare**: such as medical assistance by providing medical advice, facilitating diagnoses, and offering mental health support.
- **AI for Sustainability**: developing efficient AI so that it saves energy and computational resources, improving energy consumption in the end.
- **Accessibility**: expanding access to AI technologies for underprivileged communities such as developing AI models for people speaking low-resource or endangered languages.

While this dissertation primarily focuses on AI for Healthcare as a case study, AI for Social Good also addresses other critical issues beyond those mentioned above, such as climate change, energy conservation, water preservation, and more, which are discussed in the future works section

1.2 Dissertation Organization

Chapter 1 establishes the motivation and research vision for this dissertation, focusing on the challenges faced by advanced NLP systems powered by large language models. It introduces an extended definition of human-centered NLP, emphasizing the integration of insights from human understanding into system design. This chapter lays the foundation for the dissertation by outlining the key topics, objectives, and contributions explored in the subsequent chapters.

The remainder of this dissertation is divided into three main parts, each addressing a component of human-centered NLP:

Part I (Chapter 2-6). This section focuses on improving the trustworthiness of NLP systems by addressing key challenges:

Chapter 2 focuses on improving *robustness* by proposing TDG, a solution for addressing vulnerabilities in large language models. Chapter 3 examines *fairness* issues in NLP systems, critiques the limitations of existing debiasing methods, and introduces a novel debiasing paradigm called Interpretable Debiasing, which is designed to enhance both fairness and *interpretability*. Chapter 4 explores the integration of *interactivity* within the Interpretable Debiasing framework, aiming to increase user satisfaction through a more dynamic and interactive approach. Chapter 5 investigates model *safety* and presents DEPEN, a method designed to mitigate the leakage of sensitive information in human-written data. Chapter 6 concludes this part by discussing model *detoxification* using synthetic data and pre-training tasks.

Part II (Chapter 7-8). This section introduces the component of learning from human cognition, a key extension of the original definition of human-centered NLP.

Chapter 7 investigates human-like *cognitive biases* that emerge in high-stakes decision-making processes involving LLMs and human users, highlighting the parallels between artificial and human reasoning. Chapter 8 explores the concept of *human memorability* and presents CAMELoT, a novel Transformer architecture inspired by the human brain. This model is designed to improve the handling of long-term dependencies in data, leveraging insights from cognitive processes to enhance performance.

Part III (Chapters 9-10). This section focuses on the practical applications of human-centered NLP systems, particularly in the domain of healthcare. Chapter 9 introduces a new *benchmark* for evaluating LLM performance across diverse medical tasks, emphasizing the importance of cautious and responsible deployment in medical applications. Chapter 10 details the development of an *LLM-powered healthcare system* designed to assist in disambiguating medical reports written by healthcare providers, ultimately improving patient healthcare outcomes.

In the end of the dissertation, Chapter 11 summarizes my research contributions, and discusses the future directions. It advocates for increased efforts to build human-centered NLP systems that align with human values and societal needs in the era of LLMs.

Part I

Trustworthiness: Enhancing Trust Between Humans and Machines

Chapter 2

Robustness

In this chapter, we delve into the robustness of NLP systems, focusing on the need for consistent performance across diverse distributions and contexts. While state-of-the-art NLP models often achieve high aggregate accuracy, they frequently fail systematically on specific subgroups of data, leading to unfair outcomes and diminishing user trust. Additional data collection may not help in addressing these weaknesses, as such *challenging subgroups* may be unknown to users, and remain underrepresented in the existing and new data.

To tackle this issue, we propose Targeted Data Generation (TDG) [He et al., 2023b], a framework that automatically identifies challenging subgroups, and generates new data for those subgroups using large language models (LLMs) with a human in the loop. TDG estimates the expected benefit and potential harm of data augmentation for each subgroup, and selects the ones most likely to improve within-group performance without hurting overall performance. In our experiments, TDG[1] significantly improves the accuracy on challenging subgroups for state-of-the-art sentiment analysis and natural language inference models, while also improving overall test accuracy.

[1]Codes and collected data will be released in https://github.com/ZexueHe/TDG.

2.1 Introduction

Despite very high accuracy, state-of-the-art NLP models still exhibit systematic failures on specific subgroups of data. For example, Rajani et al. [2022] found that a 95%-accurate sentiment analysis model did much worse on club reviews (90%) and movie theater reviews (85%), while Stuart-Ulin [2018] notes how a commercial chatbot avoids *any* engagement on topics that even mention Islam or the middle east. The existence of these *challenging subgroups* can lead to unfair outcomes, erode user trust, and ultimately limit deployment of models, even when *aggregate* accuracy is very high.

One possible solution is to collect or generate more data. However, the additional data may still under-sample from specific challenging subgroups, even if data collection is adversarial [Kiela et al., 2021], especially when subgroups are not immediately obvious or salient to humans. Therefore it helps little in addressing these weaknesses. Tools for discovering challenging subgroups still require human creativity and effort [Rajani et al., 2022]. Previous works [Khani and Ribeiro, 2023, Ribeiro and Lundberg, 2022b] show that experts are able to improve existing subgroups via careful data augmentation with large language models (LLMs), but *finding* such challenging subgroups still require human ingenuity. Perhaps more importantly, they find that naively augmenting certain subgroups can drastically *hurt* other subgroups and overall performance [Ribeiro and Lundberg, 2022b]. Hence, the challenge is not only to find challenging subgroups, but also to determine which subgroups are amenable to data augmentation, and how to augment them effectively.

In this work, we propose Targeted Data Generation (TDG), a framework to automatically identify challenging subgroups that can benefit from more data, and then generate that data with LLMs (Figure 2.1). Given a target model, TDG clusters validation data into potential challenging subgroups. We then use held-out data to estimate how much each subgroup would benefit from more data, and how much additional data would hurt performance in other regions. Finally, having identified challenging subgroups amenable to data augmentation, we use GPT-3 [Brown

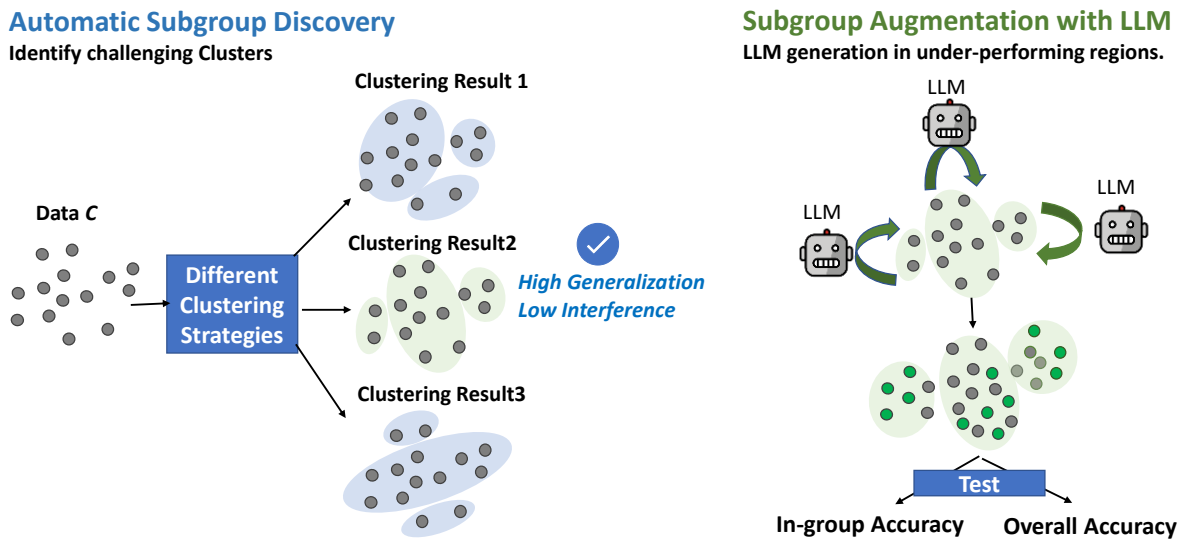


Figure 2.1. Illustration of the Targeted Data Generation (TDG) pipeline. In the automatic subgroup discovery stage, TDG identifies challenging clusters that can benefit from additional data while minimizing potential negative impacts on performance in other regions (i.e., high generalization (GC) and low interference (IC), as defined in Section 2.2.1). In the subgroup augmentation with LLM stage, TDG utilizes GPT-3 to generate additional examples for identified challenging clusters.

et al., 2020] coupled with local subgroup models to generate new data, so as to improve subgroup performance while remaining faithful to the original data distribution.

We evaluate TDG on three tasks: sentiment analysis (SST), paraphrase detection (QQP), and natural language inference (MNLI). We evaluate various clustering techniques, and find that clustering based on the target model's own representation yields the clusters most amenable to data augmentation (with the exception of QQP, where our analysis indicates label noise would make data augmentation ineffective). Finally, augmenting these clusters with GPT-3 results in significant improvements on correspondent test clusters, and also small improvements on overall accuracy.

2.2 Proposed Approach: Targeted Data Generation

Let $\mathcal{M}$ be a target model trained on a training dataset D_{train}, and let D_{test} be a held-out test dataset. We assume access to a validation dataset D_{val}, which we use to identify and evaluate challenging subgroups. We cluster D_{val} into k disjoint clusters, $\mathcal{C} = \{c_1, c_2, \ldots, c_k\}$, using some clustering technique (we explore various options in Section 2.2.2, and drop the subscript when talking about a single cluster, for clarity). We divide D_{val} randomly into two halves, so that each cluster is divided into c_{train} and c_{test} (c_{val} can be further divided from c_{train} if necessary), to simulate the effect of data augmentation and its impact on the same subgroup. We say a cluster c is a *challenging cluster* if the target model $\mathcal{M}$ performs much worse on it than on the overall validation dataset, i.e., $\text{Acc}(\mathcal{M}, c_{\text{train}} \cup c_{\text{val}}) << \text{Acc}(\mathcal{M}, D_{\text{val}})$.

Given a challenging cluster c, our goal is to identify whether it is amenable to data augmentation, i.e., more data would generalize and improve performance on c_{test}, without hurting performance on D_{test}.

2.2.1 Generalization and Interference, in Context

Given the context of $(D_{\text{train}}, \mathcal{M})$ and a target cluster c, we obtain a new model $\mathcal{M}'$ by training on a mixture of D_{train} and c_{train},following Ribeiro and Lundberg [2022a], which effectively upweights examples from c as a surrogate for data augmentation. We use two statistics to evaluate whether c is amenable to data augmentation: Generalization in Context (GC) and Interference in Context (IC).

Definition 2.2.1 (Generalization in Context). We say a cluster c generalizes in the context of the current model $\mathcal{M}$ and dataset D if more training on it leads to better performance on hidden examples from the same cluster. Formally, we define Generalization in Context (GC) as

$$\text{GC}(c) = \text{Acc}(\mathcal{M}', c_{\text{val}}) - \text{Acc}(\mathcal{M}, c_{\text{val}})$$

GC measures how much the target model can learn from more data from the cluster,

and whether that learning transfers to unseen data from the same cluster. A high GC indicates that the cluster is challenging but not hopeless, and that data augmentation could help improve performance. A low GC indicates that the cluster is either already saturated by existing data or too hard for the model to learn, such that more data from the cluster does not help. For example, if the clustering is random, we would expect a low GC, as training on a random subset of data would not improve performance on another random subset. Conversely, if the clustering is based on some meaningful feature that the model struggles with, (such as club reviews [Rajani et al., 2022]), we would expect a high GC, as training on more data from the cluster would help the model overcome its weakness.

Definition 2.2.2 (**Interference in Context**)**.** We say a cluster c interferes with the original data if augmenting it leads to worse performance on the original data. We could similarly evaluate interference with other clusters, but for now we restrict ourselves to having the original model and dataset as the context. Formally, we define Interference in Context (IC) as

$$\text{IC}(c) = \text{Acc}(\mathcal{M}, D_{\text{val}}) - \text{Acc}(\mathcal{M}', D_{\text{val}})$$

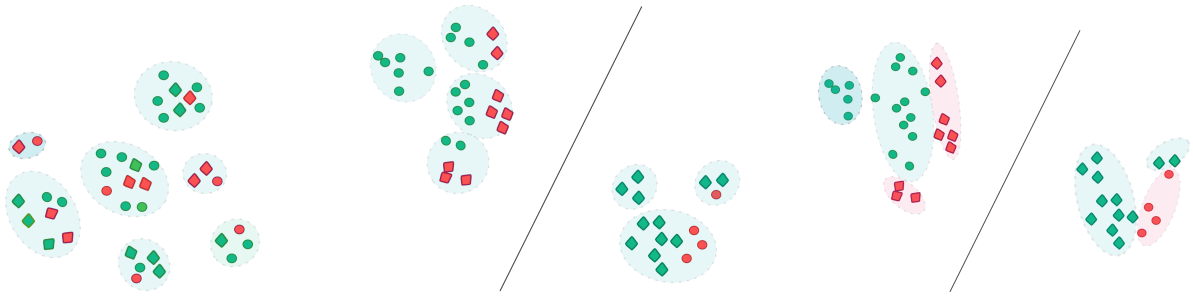
A high IC indicates that the cluster is incompatible with the original data, and that data augmentation would degrade overall performance. A low IC indicates that the cluster is either similar to the original data, or sufficiently different but not conflicting, such that data augmentation would not hurt overall performance. For example, if c is label-imbalanced and D is label-balanced, we would expect a high IC, as training on more data from c might bias the model towards a certain label and hurt performance on D. Conversely, if c and D are from different domains but share some common concepts, we would expect a low IC, as training on more data from c would not confuse the model on D. A negative IC indicates that augmenting c actually improves performance on D, which could happen if D is small and the model has not saturated it yet, or if there is some domain shift between D_{test} and D_{train} which augmentation helps to bridge.

Aggregate statistics To summarize, GC measures whether a cluster benefits from more data,

while IC measures whether augmenting that cluster would hurt performance on the original dataset. We aggregate GC and IC over all clusters by taking the average:

$$\overline{\text{GC}}(C) = \sum_{i=1}^{k} \frac{\text{GC}(c_i)}{k} \tag{2.1}$$

$$\overline{\text{IC}}(C) = \sum_{i=1}^{k} \frac{\text{IC}(c_i)}{k} \tag{2.2}$$



(a) **(b)** **(c)**

Figure 2.2. Example illustration of cluster results on binary classification from different clustering methods. Data points from binary categories are identified by dots and squares. Errors are shown in red. (a) Agnostic clustering where positive and negative data points are mixed together; (b) Task-based clustering where most points of one category are located at one side of the decision boundary of model $\mathcal{M}$ (being separable by $\mathcal{M}$) and positive/negative points are mixed in clusters; (c) Task-based clustering + label information: besides being separable, data points with the same label can be clustered together.

2.2.2 Automatic Subgroup Discovery

We use different representation spaces for clustering, using increasing amounts of information about the task, the model, and the labels. The example is shown in Figure 2.2.

Agnostic clustering We do not use any information about the task, the model, or the labels, and instead use general-purpose embeddings, such as the embeddings extracted from Sentence-BERT implemented in sentence-transformers [Reimers and Gurevych, 2019], to cluster the validation data. This kind of representations might capture some patterns that the target model cannot

currently represent well, and that augmenting these clusters would teach the target model new concepts or relations.

Task-based clustering We use the target model's own representation from the second-to-last layer to cluster the validation data. This kind of representations reflects how the target model perceives the data, and might group together examples that the model considers similar or difficult. We expect that if the model relies on spurious correlations or heuristics, these might show up in the representation and get clustered together. Augmenting these clusters would force the model to learn more robust features or strategies.

Task-based + label information We use the same representation as task-based clustering, but with the constraint that all examples in a cluster must have the same label (similar to Sohoni et al. [2020]). While this creates clusters that are clearly label-imbalanced, we expect that examples close in the target representation will also tend to have the same label, and thus this clustering technique should yield clusters with very low or very high error rate (the latter are good candidates as challenging clusters).

Selecting clusters for augmentation Given a budget of k clusters we can augment, we evaluate the clustering representations using the aggregate GC and IC statistics of their top-k clusters ranked by error rate, resulting a set of clusters C_k. In other words, we choose a representation that yields the most augmentable clusters without hurting overall performance, as formalized in Equation 2.3.

$$\mathcal{C}_k^* = \arg\max_{C_k} [\overline{\text{GC}}(C_k) - \overline{\text{IC}}(C_k)] \tag{2.3}$$

2.2.3 Subgroup Augmentation with LLMs

In order to augment those top challenging clusters $\mathcal{C}_k^*$, we follow the work of Khani and Ribeiro [2023] to use GPT-3 to create similar in-cluster examples, with a human in the loop to provide labels. We finetune a small local model on each cluster's data and use the disagreement

between that model and the current version of $\mathcal{M}'$ to rank GPT-3 generated examples, stopping the process once the current version of the cluster's model mostly agrees with the current version of $\mathcal{M}'$. Intuitively, when $\mathcal{M}'$ and the cluster's model converge on cluster data, $\mathcal{M}'$ has learned to generalize to the data in this cluster (thus fulfilling the requirment of GC), and the original $\mathcal{D}$ used when updating $\mathcal{M}'$ should prevent high interference.

(a) Agnostic clustering; GC=0.0064; IC=0.0000

(b) Task-based clustering; GC=0.011; IC=-0.0002

(c) Task-based + label information; GC=0.1319, IC=0.19298

Figure 2.3. Error distribution of clusters obtained from three clustering methods on SST. Cluster number k=35. For random clustering: GC=-0.0010, IC=0.0000

(a) Agnostic clustering; GC=0.0013; IC=0.0011

(b) Task-based clustering; GC=0.028; IC=-0.0017

(c) Task-based + label information; GC=0.0434, IC=0.0023

Figure 2.4. Error distribution of clusters obtained from three clustering methods on MNLI. Cluster number k=100. For random clustering: GC=-0.0007, IC=0.0002

2.3 Experiments

Setup We evaluate the effectiveness of TDG on three tasks from the GLUE benchmark: The Stanford Sentiment Treebank (SST), MultiNLI Matched (MNLI-m) and Quora Question Pairs (QQP). We train a bert-base model for SST and RoBERTa-large models for MNLI and QQP on the official training corpora released in GLUE benchmark to match the best Transformer

performance.[2] They are regarded as the target model $\mathcal{M}$ in each task. We randomly divide the validation data into two half sets: a *dev* set, used for automatic subgroup discovery, and a *devtest* set, used exclusively for evaluation. Therefore, SST has dev size of 436, MNLI dev has size of 4,908, and QQP has dev size of 20,215. We run each experiment five times with different random seeds and report the average scores.

2.3.1 Automatic Subgroup Discovery

We conduct clustering methods on the dev set of each task. We assign the closest cluster to each instance in the devtest set, such that each cluster in dev has an aligned counterpart for evaluation. We run each clustering method five times using different random seeds and select the clustering results with the best Silhouette scores [Rousseeuw, 1987]. **Comparison of clustering representations** We present the error rates of discovered clusters for SST and MNLI in Figures 2.3 and 2.4. For both tasks, errors were randomly distributed accross clusters produced by agnostic clustering, which indicates that the clusters are not aligned with model behaviors and weaknesses, as also confirmed by the low GC and IC scores. In contrast, task-based clustering (with or without label information) results in a large contingent of clusters with zero or few errors (i.e. most successes are clustered together), and a few clusters with higher error rates. Using label information yields clusters of either all errors or all successes, which results in high Generalization in Context scores, but also high Interference in Context scores. Both are likely due to label imbalance, as we would expect such scores from simply shifting the likelihood of predicting the cluster label. This analysis thus indicates that task-based clustering without labels yields the clusters that are most amenable to augmentation, since clusters have positive generalization and near-zero interference scores. We use these clusters in subsequent results.

QQP All clusterings on QQP (not shown) had very high interference scores, and thus were not deemed suitable for augmentation by TDG. Indeed, when we piloted data augmentation procedures on these clusters, we saw no tangible benefits. Manual inspection of clusters indicates

[2]Following Bowman et al. [2015], Yanaka et al. [2019], we use the binarized version of MNLI

that QQP has high label noise (which would explain interference), such that pairs with the same phenomena are often labeled differently, e.g. the pair ("What makes life worth living?", "Is life worth it?") is labeled as not-duplicate, while ("Why is Deadpool so overrated", "Is Deadpool overrated") is labeled as duplicate. In this case, TDG correctly identifies a case where subgroup data augmentation is unlikely to be effective, and other solutions (e.g. data cleaning) should be pursued. We do not report any QQP results from now on.

2.3.2 Subgroup Augmentation with LLMs

Based on the high-GC and low-IC clusters discovered in previous step, we conduct augmentation targeted on those clusters with large language models with human in the loop.

Human Participants We recruited 12 users to label GPT-3 generated data in the subgroup augmentation step. All users are from academia or industry (with IRB approval) and have experience working with AI-based natural language generation systems (e.g. GPT-3). Each user was assigned a high-error cluster discovered in the automatic subgroup discovery step (2 from SST and 10 from MNLI), and asked to label GPT-3 generations. We use the original sentences from the cluster as the initial prompt. Sentences that users labeled differently from the model's prediction were added to the augmented set. We allocated 90 minutes for user labeling.

Baselines We compare TDG to the following previous works that aim to improve subgroup performance: (1) **Reweighing** [Sohoni et al., 2020], which addresses hidden stratification caused by dataset imbalance by optimizing the per-cluster worst-case performance. In our experiments, we use the same Group Distributionally Robust Optimization (GDRO) introduced in their work on each cluster as the fine-tuning objective. (2) **Paraphrasing** where we use Parrot [Damodaran, 2021], a T5-based paraphrase model, to generate similar examples of data points in clusters as an augmentation. The size of the final fine-tune set is the same as TDG for a fair comparison.

One cluster at a time v.s. simultaneous augmentation Each participant augmented a single cluster, and we report these results as **TDG(single)**, noting that for these we only measure

in-cluster performance. We further pool the data from all participants (**TDG(all)**) to test the improvements on each cluster as well as performance on the overall test set (devtest). In each experiment, in order to avoid the issue of catastrophic forgetting [McCloskey and Cohen, 1989], we randomly sampled training data with the same frequency as TDG augmented data in the fine-tuning process[3].

Table 2.1. Accuracy of TDG v.s. baselines tested on top-2 error clusters and left-out devtest set of SST. BERT-base is the target model $\mathcal{M}$.

Model	SST			
	1st	2nd	Avg Cluster	devtest
BERT-base	81.74	81.13	81.45	93.77
Reweighing	78.7	82.03	80.37	93.49
Paraphrasing	77.61	82.42	80.02	92.26
TDG (single)	**83.8**	**83.39**	**83.60**	-
TDG (all)	82.61	**83.39**	83.00	**94.32**

Table 2.2. Accuracy of different models tested on top-10 high-error clusters and left-out devtest set of MNLI.

Model	MNLI											
	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	Avg Cluster	devtest
RoBERTa-Large	51.85	**53.57**	53.85	54.84	55.56	58.82	65.71	66.56	**68.75**	76.19	60.57	93.46
Reweighing	51.85	**53.57**	30.77	58.06	55.56	58.82	68.57	65.91	**68.75**	73.81	58.57	93.46
Paraphrasing	51.85	42.86	53.85	54.84	44.44	58.82	65.71	65.91	**68.75**	26.19	53.32	86.45
TDG (single)	51.85	**53.57**	61.54	**67.74**	**66.67**	**64.71**	65.71	**75.68**	66.67	76.19	**65.03**	-
TDG (all)	**59.26**	**53.57**	**64.28**	61.29	55.56	**64.71**	**74.28**	68.18	**68.75**	**78.57**	64.85	**93.62**

Improvement in challenging subgroups Table 2.1 and Table 2.2 show the results of all baselines, as well as TDG(single) and the aggregated TDG(all), on the SST and MNLI tasks, respectively. For both tasks, augmenting individual clusters with TDG tends to be more effective than all baselines and ablations, as the average in-cluster accuracy has been increased from 81.45% to 83.60% on SST and from 60.57% to 65.03% on MNLI, which is higher than any baseline models. Additionally, we also observed that adding TDG data from all clusters can

[3]In MNLI experiment, due to the high interference among clusters, we adjust the weights of training samples and collected responses when combining all data points for TDG(all) in fine-tuning (i.e., we set portions of original samples:user responses = 2:1). In SST, all responses are combined without any adjustment.

improve all clusters by an average of 4.28% (from 60.57% to 64.85%) on MNLI and an average of 1.55% (from 81.45% to 83.00%) on SST, which is also higher than all baseline models. Note that the accuracy of every single cluster in TDG(all) is better than the target model. For some challenging clusters, augmentation on their own (TDG(single)) may yield better results, due to potential interference between clusters.

Improvement in overall devtest We observed an improvement in overall performance on the devtest set with TDG(all), with an increase of 0.55% on SST and 0.16% on MNLI. This suggests that improving challenging clusters has the potential to improve the model at a global level, while neither baselines were able to achieve this. We notice the improvement on the devtest set is not as significant as the improvement on individual low-performed groups. This is likely due to the fact that these vulnerable groups are usually minorities and their representation in the devtest set is small (e.g., the average size of the 10 clusters in MNLI experiment is just 88 whereas the devtest has size of 4,908), diluting the impact of the improvement.

Ablation Analysis We evaluate the following variations of TDG to test the effectiveness of each step:

- **Automatic Subgroup Discovery Only** in which the fine-tuning data is created by using the same clusters as TDG but without augmentation and adding the same number of random samples from the training data, to test the error discovery step.
- **Subgroup Augmentation with LLM Only** in which the fine-tuning data is created by using n random samples from the dev set (n is the number of total sentences in challenging clusters used in TDG) and applying subgroup augmentation with GPT-3, to test the effectiveness of the augmentation. Augmentation ends once the same number of augmented data as TDG is reached.

We see that fine-tuning with clusters alone can improve performance on certain clusters when the size is sufficient (e.g., 2nd in SST), but it can also lead to over-fitting and reduced

Table 2.3. Accuracy of different ablations of TDG on top-2 high-error clusters in SST. BERT-base is the target model $\mathcal{M}$.

Model	SST			
	1st	2nd	Avg Cluster	devtest
BERT-base	81.74	81.13	81.45	93.77
Automatic Subgroup Discovery only	78.70	82.20	80.45	93.89
Subgroup Augmentation with LLM only	79.42	78.42	78.91	93.17
TDG (single)	**83.80**	**83.39**	**83.60**	-
TDG (all)	82.61	**83.39**	83.00	**94.32**

performance (e.g., 1st in SST). Additionally, subgroup augmentation on randomly sampled clusters results in a decrease in performance not only in low-performing areas, but also overall on the devtest set. Without the automatic subgroup discovery, the GPT-3 augmented sentences may introduce more noise rather than benefits, which verifies the bottleneck of previous work [Ribeiro and Lundberg, 2022a] and emphasizes the importance of the automatic subrgoup discovery.

Interpretation of low-performed groups In this section, we present some examples from the high-error groups discovered in automatic subgroup discovery. We also provide readable interpretations for the clusters as shown in Table 2.4. Our automatic subgroup discovery is able to identify meaningful errors, such as mis-identifying the dominant sentiment from a mixture of sentiments in SST, or errors related to different language tones in MNLI. Furthermore, we also notice complex patterns in reasoning is identified, such as Factivity and Monotonicity, which are recognized challenges in SuperGLUE Diagnostic tasks.

2.4 Conclusion

In this work, we presented a thorough analysis of error distribution among different groups and introduced Targeted Data Generation (TDG), a framework that automatically identifies challenging groups that are amenable to improvement through data augmentation using large language models (LLMs) without negatively impacting overall accuracy. Our experiments with

state-of-the-art models demonstrate that TDG is able to improve in-group performance by 2-13% while also increasing overall accuracy. Furthermore, TDG was able to improve performance for every single selected cluster without interference, indicating its potential as a reliable approach for a new data collection framework. As LLMs continue to advance and are trained on more diverse and large corpora, TDG represents a promising approach for addressing the weaknesses of simpler models.

Chapter 2, in part, is a reprint of the material as it appears in "Targeted Data Generation: Finding and Fixing Model Weaknesses " by Zexue He, Marco Tulio Ribeiro, Fereshte Khani, referenced as [He et al., 2023b], in proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2023. The dissertation author was the primary investigator and author of this paper.

Table 2.4. Interpretation about discovered high-error clusters. Each cluster is shown with two errors.

	Cluster: Having multiple sentiments and one is dominating than the rest	**Label**	**Prediction**
SST	On the heels of the ring comes a similarly morose and humorless horror movie that, although flawed , is to be commended for its straight-ahead approach to creepiness .	positive	negative
	Another one of those estrogen overdose movies like "divine secrets of the ya ya sisterhood ” except that the writing , acting and character development are a lot better .	positive	negative
	Cluster: Having same meaning. Formal Tone v.s. Casual Tone	**Label**	**Prediction**
MNLI	**Sentence1**: Do you think I should be concerned? **Sentence2**: Do you think it is a problem	entailment	not enatilment
	Sentence1: He seemed too self-assured. **Sentence2**: He is very cocky	entailment	not enatilment
	Cluster: One v.s. All	**Label**	**Prediction**
	Sentence1: Pray be seated, mademoiselle. **Sentence2**: Please, everyone be seated.	not enatilment	entailment
	Sentence1: Similar conclusions have been reached by legal studies in a dozen states including Florida. **Sentence2**: Similar conclusions have been seen across the world.	not enatilment	entailment
	Cluster: Suspicion v.s. Fact	**Label**	**Prediction**
	Sentence1: The analysis also addresses the various alternatives to the final rule which were considered, including differing compliance or reporting requirements, use of performance rather than design standards, and an exemption for small entities from coverage of the rule. **Sentence2**: The rule is subject to change."	not enatilment	entailment
	Sentence1: In the depths of the Cold War, many Americans suspected Communists had infiltrated Washington and were about to subvert our democracy. **Sentence2**: Communists infiltrated Washington during the Cold War.	not enatilment	entailment

Chapter 3

Fairness & Interpretability

In this Chapter, we discuss a novel method that reduces the model biases in an interpretable way. Human-written language contains implicit or explicit biases and stereotypes, which make their way into deep NLP systems through the learning procedure. Emerging works show that biases may have worrisome influence and even lead to unfair outcomes in various NLP tasks like text classification [Park et al., 2018, Kiritchenko and Mohammad, 2018, De-Arteaga et al., 2019], coreference resolution [Rudinger et al., 2018], toxicity detection [Zhou et al., 2021, Xia et al., 2020, Xu et al., 2022a], language modeling [Lu et al., 2020, Bordia and Bowman, 2019, Sheng et al., 2019], etc. Recent work on reducing bias in NLP models usually focuses on protecting or isolating information related to a sensitive attribute (like gender or race). However, when sensitive information is semantically entangled with the task information of the input, e.g., gender information is predictive for a profession, a fair trade-off between task performance and bias mitigation is difficult to achieve. Existing approaches perform this trade-off by eliminating bias information from the latent space, lacking control over how much bias is necessarily required to be removed.

Instead, we argue that a favorable debiasing method should use sensitive information 'fairly', rather than blindly eliminating it. [Caliskan et al., 2017, Sun et al., 2019] . In this work, we provide a novel debiasing algorithm called Interpretable Debiasing [He et al., 2022] by adjusting the predictive model's belief to (1) ignore the sensitive information if it is not

useful for the task; (2) use sensitive information *minimally* as necessary for the prediction (while also incurring a penalty). Experimental results on two text classification tasks (influenced by gender) and an open-ended generation task (influenced by race) indicate that our model achieves a desirable trade-off between debiasing and task performance along with producing debiased rationales as evidence.

3.1 Preliminaries

3.1.1 Debiasing: Sensitive Attribute Protection

Recently, several works have attempted to address bias issues in NLP tasks. One stream of approaches is sensitive attribute protection [Zhang et al., 2018, Jentzsch et al., 2019, Badjatiya et al., 2019, Heindorf et al., 2019, He et al., 2021c], which mitigates bias by isolating or protecting certain sensitive attributes like race or gender from decision making. However, real-world human-written language is complicated and there are often cases where sensitive information is entangled tightly with the semantics of the sentence [Caliskan et al., 2017]. In this situation, protecting the attribute will unavoidably affect the model's performance. For example, isolating all the underlined words in

Example 1. *He is a congressman and he is good at singing.*

might misguide a 'profession' classifier to get a result of a *singer* (instead of a *congressman*). The balance between bias mitigation and other desired goals is challenging in current debiasing scenarios [Sheng et al., 2021]. Conceptually, debias methods that protect sensitive attributes in some latent space may achieve such a delicate equilibrium if bias is reduced to some precise degree. However, controlling the degree of debiasing in a transparent fashion is challenging [Gonen and Goldberg, 2019] as these methods [Zhang et al., 2018, Ravfogel et al., 2020, Gonen and Goldberg, 2019] operate in a black-box style, providing no evidence for bias mitigation or task performance. Hence, it remains hard for human users to understand and trust the underlying debiasing mechanism.

3.1.2 Interpretability: Model Rationales

Recent works [Lei et al., 2016, Bastings et al., 2019] have shown that *rationales* are an effective way to justify the reasoning behind a prediction from a neural model. Rationales are defined as pieces of the input text that are tailored t be short and coherent, yet sufficient for making the same prediction using the pieces of input as using the entire input. It usually is extracted with an generator-encoder architecture, where generator learns a distribution over text fragment as candidate rationales (a list of Bernoulli distribution) whereas the encoder is optimized to maximize the probability of making the same prediction after getting the rationales candidate from encoder. The framework is regularized by desiderata of being short and coherent. for rationales

3.2 Proposed Approach: Interpretable Debiasing

In this section, we introduce our interpretable debiasing algorithm that uses a 'fair' amount of sensitive information in the important parts of input (a.k.a. rationale). We aim to perform a predictive task (e.g., predicting a profession based on a biography) while minimizing the impact of sensitive information (e.g., gender) with minimally affecting the performance of the original task. Given an input, there are tokens that are predictive of the task output (we call them task rationales) and there are tokens that carry the sensitive information (we call them bias rationales). With energy functions, we measure how important a token is for the task output or how sensitive it is. By constraining the use of biased input tokens, we control the task energy so that the model is allowed to be exposed to a minimum of bias that is necessary to the task.

3.2.1 Extracting Bias Rationale

We first identify input tokens that carry sensitive information. To be more specific, for an input text $\mathbf{x} = \{x_1, x_2, x_3, \cdots, x_n\}$ with n tokens (e.g., biography of a person), we predict the bias label y_b (e.g. gender of the person, having K_b categories) based on $\mathbf{x}$ with model $f_b(\mathbf{x}; \theta_b)$

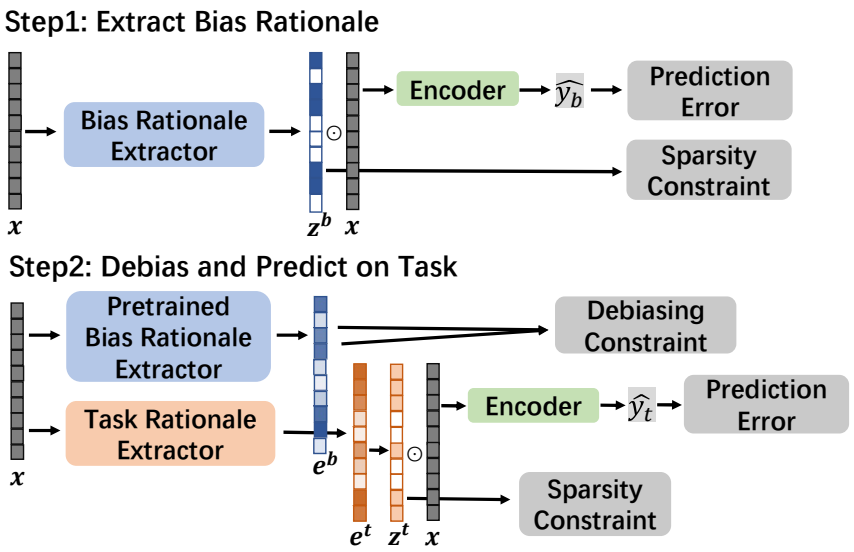


Figure 3.1. Pipeline of Interpretable Debiasing Framework. We first pretrain a bias rationale extraction framework and obtain bias energy for each input token. Then we train a *fair* task prediction model where the task rationales are regulated by a debiasing constraint based on bias energy. A token with high bias energy will be penalized for being in task rationale with a decrease in its original task importance.

parameterized by θ_b, so that the predicted bias label $\hat{y_b}$ is close to ground truth y_b

$$\hat{y_b} = \arg\max_{k_b \in K_b} f_b(\hat{y_b} = k_b | \mathbf{x}; \theta_b),$$

which is optimized by minimizing the cross-entropy error $\mathcal{L}_{bias}(f(\mathbf{x}), y_b; \theta_b)$. We are interested in identifying the tokens that are most predicted $\hat{y_b}$, i.e. bias rationales.

Rationale is defined as a short yet sufficient snippet of an input responsible for the prediction [Bastings et al., 2019]. Here, we obtain the bias rationale using an extractive framework that includes two modules – an extractor that identifies parts of input as the rationale, and an encoder that makes a prediction only based on the rationale. The extractor and encoder together compose the rationale extraction framework (REF). The proposed rationale comes in the form of a sequence of binary variables, indicating if a particular input token is informative to the task. The extractor and the encoder are jointly trained to minimize the prediction error.

Therefore, to extract bias rationale, we augment f_b with the sequence of latent binary

variables $\mathbf{z^b} = \{z_1^b, z_2^b, z_3^b, \cdots, z_n^b\}$, $z_i^b \in \{0, 1\}$ [Lei et al., 2016], which is optimized to maximize the predictive probability of the correct bias label by regulating the contribution of each token:

$$\mathbf{z^b} \sim g_b(\mathbf{x}|\phi_b)$$

$$\hat{y_b} = \arg\max_{k_b \in K_b} f_b(y_b = k_t|\mathbf{x} \odot \mathbf{z^b}; \theta_b)$$

where g_b is a bias rationale extractor parameterized by ϕ_b, that predicts the probability of how much each token contributes to predict the bias label. We sample the binary vector $\mathbf{z^b}$ from g_b and $\mathbf{x} \odot \mathbf{z^b}$ is treated as the *bias rationale*. We model g_b such that the output of g_b satisfies Kuma distribution [Bastings et al., 2019] to avoid $\mathbf{z^b}$ being non-differentiable.

Bias REF is trained with the following objective and important tokens for predicting bias are selected as bias rationales:

$$\mathcal{C}_b = \mathcal{L}_b(f_b(\mathbf{x} \odot \mathbf{z^b}); \theta_b) + \lambda_b \Omega_b(\phi_b)$$

where λ_b is hyperparameter and Ω_b is a sparsity constraint penalizing the number of selections and translations, making learned rationale concise and sufficient.

3.2.2 Task Prediction

Based on the bias rationale obtained so far, we want to influence a predictive model to use input tokens in a debiased way. Elaborately, we want the contribution of the biased tokens to be as minimum as possible for the predictive task. To achieve this, we encourage the predictive model for a task (e.g., profession classification with K_t classes) to use informative tokens (task rationales) with minimal bias.

Similar to bias rationale extraction, we train a task REF consists of an extractor g_t that generates $\mathbf{z_t} = [z_1^t, z_2^t, \cdots, z_3^t]$, and an encoder f_t that makes prediction with extracted rationale

$\mathbf{x} \odot \mathbf{z}^t$

$$\mathbf{z^t} \sim g_t(\mathbf{x}|\phi_t)$$

$$\hat{y}_t = \arg\max_{k \in K_t} f_t(\hat{y}_t = k_t|\mathbf{x} \odot \mathbf{z}^t; \theta_t)$$

where $\hat{y}_t$ is the task prediction and y_t is the ground truth label ($y_t \in C_t$). Task rationale is extracted by minimizing the task cross-entropy loss $\mathcal{L}_t$ and maintaining the sparsity Ω_t, as

$$\mathcal{C}_t = \mathcal{L}_t(\mathcal{F}(\mathbf{x} \odot \mathbf{z^t}); \theta_t) + \lambda_t \Omega_{task}(\phi_t)$$

However, we would like to modify the task REF to consider bias rationale, and optimize task rationale in such a way that they contain minimal bias. For this, we introduce a debiasing constraint that adds a penalty if a biased token is used as the part of the task rationale, and optimize the task rationale to incur minimal penalty.

3.2.3 Debiasing with Energy-Based Constraint

Our debiasing constraint should regulate the importance of the biased tokens towards the predictive task. We capture the importance of each token for being biased and being important for the predictive task, using *energy* scores[1]. *Energy* is defined as the negative log-likelihood of the non-selection probability of each token [LeCun et al., 2006]. Higher energy indicates stronger importance.

We obtain the task energy for the i-th token as:

$$\begin{aligned} e_i^t &= -\log\text{-likelihood}(p(z_i^t = 0)) \\ &= -\log\text{-likelihood}(1 - g_t(x_i|\phi_t)), \end{aligned}$$

[1]We did not use direct probabilities from REFs since they produce unstable performance as $p(z_i^b = 0)$ and $p(z_i^t = 0)$ may not be independent and may not be summable. See Section 3.3 for the experimental evidences.

where $g_t(x_i|\phi_t)$ is the probability for selecting the i-th token x_i for the task prediction. Similarly, the bias energy for the i-th token would be:

$$e_i^b = -\log\text{-likelihood}(1 - g_b(x_i|\phi_b))$$

We construct the debiasing constraint using both task and bias energy for a token. For an i-th token that has a high bias energy, we will penalize its importance for the predictive task by decreasing its task energy. In contrast, for tokens with low bias energy, we keep task their energy as it is. This is realized by a debiasing constraint as:

$$D(i) = \begin{cases} e_i^t + (e_i^b - A) & \text{if } e_i^b > A, \\ 0 & \text{otherwise} \end{cases}$$

where A is a hyperparameter indicating the bias tolerance threshold. This constraint will eventually get rid of highly biased token for being important to the task and use low-bias energy replacements instead, in order to boost the task performance. This modifies our task objective as:

$$\mathcal{C} = \mathcal{C}_t + \gamma \sum_{i}^{|\mathbf{x}|} D(i)$$

where γ is the hyperparameter.

3.2.4 Training

The pipeline of our algorithm is shown in Figure 3.1. We first pretrain a bias REF f_b by minimizing $\mathcal{C}_b$. During the debiasing process, this model is served as a fixed reference model. During debiasing, we then train the task model f_t by minimizing $\mathcal{C}$. For classification tasks, $\mathcal{L}_t$ is a cross-entropy loss and for generation task, $\mathcal{L}_t$ is a language-modeling loss.

Table 3.1. Evaluation of rationale-based debiasing methods on classification tasks.

Task	Variants	Toxicity F1 Score ↑	Gender F1 Score ↓	Comprehensive-ness Score ↑	Sufficiency Score ↓	Selection↓
Toxicity Detection	Full Text	0.73	0.56	-	-	100%
	Reranking	0.64	0.39	0.01	0.01	34.7%
	Probability	0.65	0.37	0.00	0.00	**63.42%**
	Ours	0.73	**0.37**	**0.00**	**0.00**	**63.34%**
Task	**Variants**	**Profession Accuracy ↑**	**Gender F1 Score ↓**	**Comprehensive-ness Score ↑**	**Sufficiency Score ↓**	**Selection ↓**
Profession Classification	Full Text	0.81	0.98	-	-	100%
	Reranking	0.70	0.45	0.23	0.32	36.40%
	Probability	0.73	0.50	0.44	0.13	**65.42%**
	Ours	0.80	**0.38**	**0.52**	**0.01**	**65.26%**

Table 3.2. Comparison between our method and other debiasing baselines without rationales on toxicity detection.

Models	Toxicity F1 ↑	Gender F1 ↓
Full Text	0.73	0.56
Adv	0.46	0.22
Embed	0.49	0.30
Ours	0.73	0.37

Table 3.3. Comparison between our method and other debiasing baselines without rationales on profession classification.

Models	Profession Acc. ↑	Gender F1 ↓	RMS TPR-GAP↓
Full Text	0.813	0.984	0.184
Adv	0.361	0.358	0.057
INLP	0.752	-	0.095
Embed	0.236	0.914	0.179
Ours	0.796	0.375	0.054

3.3 Experimental Setup

3.3.1 Scenarios and Datasets

We evaluate our debiasing algorithm on two text classification tasks influenced by *gender* bias –toxicity detection and profession classification, and an open-ended text generation task influenced by *racial* bias. We use the Jigsaw Toxicity dataset [2] for toxicity detection, BioBias

[2]https://www.kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification

dataset [De-Arteaga et al., 2019] for profession classification, and BOLD dataset [Dhamala et al., 2021] for open-ended generation.

Jigsaw Toxicity is a dataset for the Kaggle Toxic Comment Classification Challenge that detects toxicity (toxic or non-toxic) from a conversational response influenced by multiple sensitive attributes. A datapoint has an input as a textual comment associated with annotated toxicity labels and various identity attributes about the entity mentioned, such as gender, race, etc. We take gender identification as the unintended bias and filter out the examples annotated as 'no gender mentioned.' The gender categories in our dataset are female, male, transgender, and other gender. We have 125,071 examples out of which 80%, 10% and 10% are used for training, validation, and testing respectively.

BiosBias is a dataset derived from a large-scale user study of gender in occupation classification [De-Arteaga et al., 2019]. It consists of short biographies annotated with gender and occupation information. De-Arteaga et al. [2019] found possible influence of gender behind the annotated profession labels. We consider a profession classification task without the influence of gender. We follow the experimental settings in [Ravfogel et al., 2020], that contains 393,423 biographies labeled with binary gender (male/female) and 28 professions (e.g. professor, software engineer, model, etc.). 255,710 examples (65%) are used for training, 39,369 (10%) for validation, and 98,344 (25%) for testing.

BOLD or Bias in Open-ended Language Generation Dataset is proposed by Dhamala et al. [2021] to measure the fairness in open-ended language generation. This dataset contains 23,679 text generation prompts related to five domains: profession, gender, race, religious ideologies, and political ideologies, with corresponding ground-truth sentences taken from English Wikipedia. We divide the finetune/development/test set of examples in each domain with a 0.7/0.1/0.2 ratio, which is used to finetune a GPT2 language model. We then consider the four races (European Americans, African Americans, Asian Americans, and Latino/Hispanic Americans) as unintended bias. This subset consists of 7,657 prompts and ground truth, of which 5,359 (70%)

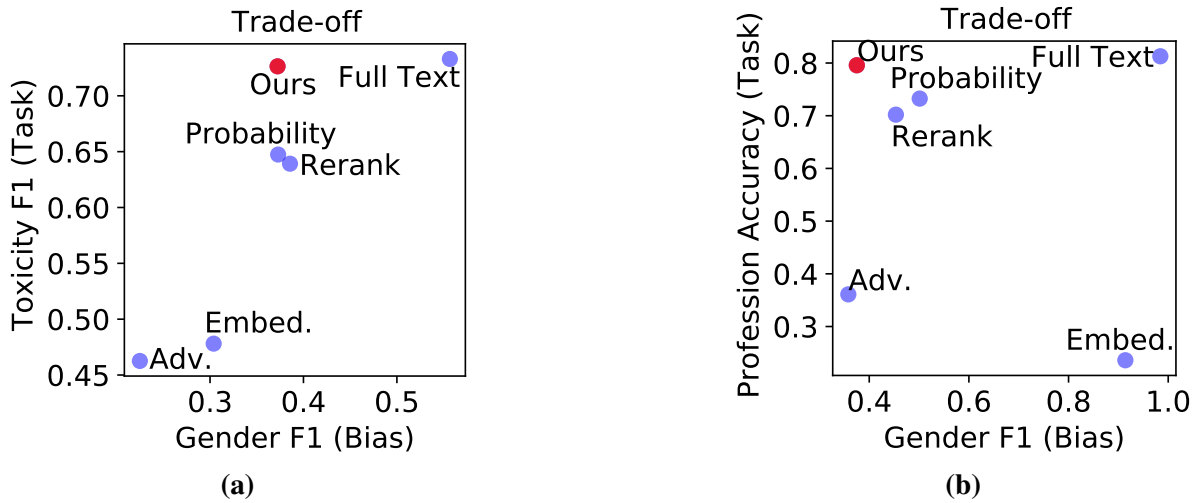


Figure 3.2. Trade-off between bias and task performance for (a) Toxicity Detection (b) Profession Classification. More upper left means a better model.

are finetuning examples, 765 (10%) are validation examples, and 1530 (20%) are test examples.

3.3.2 Baselines and Ablations

Toxicity detection. We first consider a baseline with full text input for toxicity detection. This provides us the upper bound for the task performance while it being the most biased. We also consider two other debiasing methods as baselines: a model with adversarial training (Adv.) [Zhang et al., 2018] that performs debiasing on the model's latent space, and a model [Bolukbasi et al., 2016] that performs debiasing on the embedding space (Embed).

Profession classification. Similar to toxicity detection, we also have the baseline with full text input that gives the upper bound of task performance but with maximum bias. For debiasing baselines we have Adv [Zhang et al., 2018] and INLP [Ravfogel et al., 2020], a method[3] that removes bias with an iterative null-space projection.

Open-ended Generation. We consider a language model (GPT2) trained on the original data to provide the upper bound of generation performance but with maximum bias. For debiasing

[3]Due to unavailability of the codes for INLP, gender prediction performance is not reported in Table 3.3. We use similar data settings as INLP to make other results comparable.

Table 3.4. Toxicity and gender prediction with various inputs.

Input	Toxicity F1	Gender F1
Full Text	0.73	0.56
Toxicity Rationale	0.73	0.55
Difference Δ	0.00	0.01

Table 3.5. Profession and gender prediction with various inputs.

Input	Profession Acc.	Gender F1
Full Text	0.81	0.98
Toxicity Rationale	0.80	0.98
Difference Δ	0.01	0.00

baseline, we compare with PPLM [Dathathri et al., 2019], a controllable text generation algorithm which generates output by steering the generation away from the sensitive information.

Ablations. To investigate the impact of different parts of our algorithm, we also considered two variants for comparison: (1) *Rerank* where the task rationale is selected based on a reversed order of bias energy. This is an inference-time debiasing method, which is used to investigate the necessity of debiasing constraint during training (2) *Probability* where we use probability directly obtained from REFs instead of energy for token importance.

Backbone Models. In implementation, we use LSTM as the backbone for REFs in toxicity detection and profession classification, and use GPT-2 transformer as the backbond model in open-ended generation.

3.3.3 Evaluation Metrics

To ensure the optimal trade-off between bias removal and task performance we evaluate our model based on three desiderata: (1) task performance, (2) bias mitigation, and (3) rationale faithfulness.

Task Performance. To evaluate task performance, we use F1 scores for toxicity prediction due to the imbalanced output label proportions and use accuracy for profession classification. For the

Table 3.6. Comparison of our method with debiasing baselines on open-ended generation task.

	Models	PPL↓	BertScore Precision ↑	BertScore Recall ↑	BertScore F1 ↑	Race Accuracy ↓	Sufficiency Score ↓	Selection ↓
Open-ended Generation	Ground Truth	27.69	1.00	1.00	1.00	0.63	-	100.0%
	GPT2	69.61	0.86	0.86	0.86	0.62	41.92	60.2%
	PPLM	66.97	0.81	0.81	0.81	0.61	39.28	100.0%
	Rerank	69.73	0.84	0.85	0.85	0.62	42.04	37.7%
	Probability	77.69	0.88	0.87	0.87	0.62	50.00	53.7%
	Ours	67.22	0.86	0.86	0.86	0.62	39.51	51.9%

open-ended generation task, the goal is to generate a high-quality sentence following a prompt. We use language model perplexity and BertScore [Zhang et al., 2019] w.r.t. the ground-truth text.

Bias Mitigation. Following Zhang et al. [2018], for classification tasks, we pretrain a gender classifier and report the F1 score for gender prediction before and after debiasing to measure the degree of bias mitigation. For the generation task, we also report the accuracy gap between a pretrained race classifier before and after debiasing. Additionally, for profession classification, [Ravfogel et al., 2020] showed that the root-mean-square difference in the True Positive Rates between individuals (RMS TPR-GAP) with different genders is closely related to the Equal Opportunity fairness notion [Hardt et al., 2016]—hence we report this too.

Rationale Faithfulness. To ensure that extracted rationales are trustworthy, we evaluate faithfulness in rationale-based debiasing methods using comprehensiveness and sufficiency [DeYoung et al., 2020]. Sufficiency measures the degree to which a rationale is adequate for making a prediction, while comprehensiveness indicates whether all selections are necessary for making a prediction.

A smaller decrease in sufficiency and a larger decline in comprehensiveness indicate a high degree of faithfulness. We refer readers to [DeYoung et al., 2020] for more details. We also report the rationale selection ratio to measure conciseness of the extracted rationales.

Table 3.7. Debiasing Example in Toxicity Detection. Task rationales are in green, bias rationales are in red, and overlap is in yellow. [-] indicates rationale generated before debiasing, and [+] indicate rationales after debiasing.

[-] Task Rationale	Correct , Anderson . Plowing through groups of innocent civilians is practiced by islamic terror groups such as ISIS . It is also used by Palestinians to kill babies waiting at bus stops in the arms of their mother .
Bias Rationale	Correct , Anderson . Plowing through groups of innocent civilians is practiced by islamic terror groups such as ISIS . It is also used by Palestinians to kill babies waiting at bus stops in the arms of their mother .
[+] Task Rationale (ours)	Correct , Anderson . Plowing through groups of innocent civilians is practiced by islamic terror groups such as ISIS . It is also used by Palestinians to kill babies waiting at bus stops in the arms of their mother .
[-] Task Rationale	Showing solidarity with countries inundated with refugees by taking only homosexuals , families and orphans . One slip of the lip and its over .
Bias Rationale	Showing solidarity with countries inundated with refugees by taking only homosexuals , families and orphans . One slip of the lip and its over
[+] Task Rationale (rerank)	Showing solidarity with countries inundated with refugees by taking only homosexuals, families and orphans . One slip of the lip and its over .
[+] Task Rationale (ours)	Showing solidarity with countries inundated with refugees by taking only homosexuals , families and orphans . One slip of the lip and its over .

3.4 Results and Analysis

3.4.1 Classification Tasks

Dependence on sensitive information for task prediction. First, we evaluate the appropriateness of the classification tasks by measuring how important tokens for task prediction are strong indicators of the sensitive information or bias. For toxicity detection, we observe in Table 3.4 that when prediction models use only task rationales as input, they remain highly predictive for both the predictive task as well the bias prediction—showing minimal decrease in task and bias prediction performance when we switch from using full text input to only using task rationales as input (only 0.0005 points drop for toxicity detection, 0.0032 points drop for gender prediction). A similar phenomenon for profession classification, as seen in Table 3.5, indicates that both of these tasks might benefit from our debiasing method.

Performance of rationale-based debiasing methods. Table 3.1 shows the comparison

between our methods and other baseline along the dimensions of task performance, bias mitigation and rationale faithfulness. We achieve the maximum bias mitigation with the largest F1 score drop for gender (bias) prediction on both tasks (F1 drop of 0.1844 in toxicity detection and 0.6091 in profession classification). Secondly, debiasing affects minimally the task performance. We observed a minimal performance drop (0.00 for toxicity F1 and 0.01 for profession accuracy) after debiasing for our method whereas other methods with deabised rationales suffer from larger performance loss. We see that debiasing constraint plays an important role during training to achieve better faithfulness, as we see our method achieves best comprehensiveness and sufficiency score. Finally, our method achieves the best bias-performance trade-off by selecting sparser rationales as compared most of the other baselines. Rerank selects fewest tokens for rationales but such a sparse selection eventually hurts task performance. This also indicates a necessity of debiasing constraint at the training time rather than using it directly during inference.

Performance of debiasing methods that do not produce rationales. We compare our algorithm with debaising algorithms that do not use rationales in Table 3.2 and Table 3.3 for both classification tasks. We observe Adversarial Debiasing (Adv) achieves the maximum bias mitigation in both tasks. We argue that it debiases too much, to an extent that eventually hurts the task performance as we see large drops in toxicity F1 and profession accuracy. It is indicative that debiasing on the latent space leaves us with less room to control the balance between bias mitigation and task performance. Debiasing on embedding space (Embed) performs worse in the profession classification than other baselines that it not only harms task performance but also incorporates little debiasing. Upon investigation, we found that Embed uses word embeddings pre-trained on Google News. While the domain mismatch could lead the performance degradation for profession classification task (biographies being different than Google News); for toxicity detection the domain of online context matches with Embed pretraining and hence it attributes to the poor performance of the model itself. INLP is a strong baseline however it cannot produce any rationales hence lack transparency and control as compared to our method.

Bias-performance trade-off. We visualize the trade-off between the degree of debiasing and task performance across various competing methods in Figure 3.2. The upper-left corner indicates the optimal operational point. Among all other methods, we see that for both classification tasks, our method resides closest to the upper-left corner which confirms despite having stronger debiasing methods, we maintain the fair balance between task performance and the degree of debiaising.

3.4.2 Open-ended Generation Task

We present the comparative performances of the baselines and our method for the open-ended generation task in Table 3.6. While we see that debiasing in generation task is challenging as perplexity (PPL) for all methods are far from that of the ground-truth human-written answers, our method achieves the best bias mitigation as well as best perplexity and BertScore as compared to other debiasing methods. While PPLM is fluent with a good perplexity and mitigates bias reasonably, it has low BertScore indicating low generation quality. We achieve better generation results by using sparser rationales as compared to GPT2 and Probability baselines. While Rerank selects fewest input words as rationales it eventually have poor generation quality showing lack of control on bias exposure to maintain task performance. While the Probability model acted as a strong baseline for classification tasks, for generation task, it performs worse than the GPT2 baseline. We attribute this to the lack of independence assumption between $p(z_i^b = 0)$ and $p(z_i^t = 0)$, as task labels and bias labels appears to be closely related and hence directly minimizing their sum in D might suffer from confounding in some cases. We also notice that both PPLM and our method achieve best faithfulness in terms of sufficiency but we achieve that using sparser rationales and better generation quality.

3.4.3 Case Study

We compare extracted rationales with two different inputs across different rationale-based debiasing methods for toxicity detection task in Table 3.7.

In the first example, ‘mother’ appears to be in the task rationales for toxicity as often offensive expressions and slangs include the word ‘mother’. On the other hand, ‘mother’ is also highly predictive of gender women. However, in the current context, mother is not indicative of toxicity but only acts as a sensitive token, hence our method penalizes its importance and does not use it for the task prediction after debiasing.

In the second example, ‘lip’ (frequently appears as a part of *lipstick*) and ‘homosexuals’ appear as indicator for gender as well as predicting toxicity. It is understandable that ‘homosexuals’ strongly indicates toxicity as it regularly appears in homophobic comments. While removing both them will decrease gender bias greatly, something that happens for Rerank baseline, it is not *fair* to not include ‘homosexuals’ in task rationales. While our method drops ‘lip’ from task rationales after debiasing it still keeps (and fairly so) ‘homosexuals’ in its task rationales thus controlling the bias exposure for a fair and interpretable toxicity prediction.

3.5 Conclusion

We proposed a fair and interpretable debiasing method that can control bias exposure by balancing bias mitigation and task performance. While previous methods often debias too strongly or with lesser control and transparency, we show, on three different tasks, that our method achieves the best trade-off between task performance and bias mitigation, while producing the most faithful rationales for the debiased task prediction. We also indicate cases where it is even necessary to keep sensitive information that is useful for task output. Our model provides fair control on bias exposure, especially in such cases, instead of blindly debiasing the input with minimal interpretation.

Chapter 3, in part, is a reprint of the material as it appears in “Controlling Bias Exposure for Fair Interpretable Predictions” by Zexue He, Yu Wang, Julian McAuley, and Bodhisattwa Prasad Majumder, referenced as [He et al., 2022], in Findings of the Association for Computational Linguistics: EMNLP 2022, pp. 5854-5866. 2022. The dissertation author was the primary

investigator and author of this paper.

Chapter 4

Interactivity

In this chapter, we present InterFair [Majumder et al., 2022], an extension of our previous work on interpretable debiasing, by incorporating interactivity into the debiasing framework. This approach is motivated by the belief that the definitions of fairness and bias can vary on a case-by-case basis, making the balance between fairness and task completion inherently subjective. By introducing interactivity, InterFair enables a more flexible and adaptive framework for addressing fairness in diverse contexts.

In InterFair, we explore two interactive setups with a *frozen* predictive model and show that users able to provide feedback can achieve a better and *fairer* balance between task performance and bias mitigation. In one setup, users, by interacting with test examples, further decreased bias in the explanations (5-8%) while maintaining the same prediction accuracy. In the other setup, human feedback was able to disentangle associated bias and predictive information from the input leading to superior bias mitigation and improved task performance (4-5%) simultaneously.

4.1 Introduction

InterFair shares the same motivation with Interpretable Debiasing [He et al., 2022] that instead of eliminating the biased information from the model's internal representations or from the input itself, disregarding the task performance during the process, in an ideal situation, a

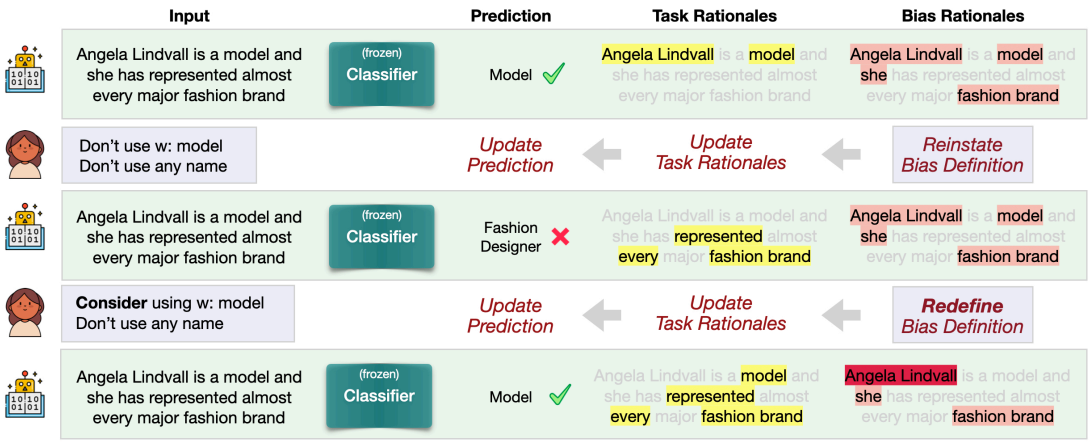


Figure 4.1. Pipeline of INTERFAIR. An algorithmically debiased model still suffers from generating biased rationales. Users interact with the final model states and perturb them using language feedback further to decrease bias and/or improve task performance.

model should use only the necessary amount of information, irrespective of bias, to achieve an acceptable task performance. Differently, in this work, we believe this trade-off between task performance and bias mitigation is subjective or varies between users [Yaghini et al., 2021] and is often hard to achieve via learning from data [Zhang et al., 2018, He et al., 2022]. Figure 4.1 shows the limit of an algorithmic approach where ignoring all gendered information can lead to a wrong result.

A user can potentially further tune the model's belief on the bias, leading to a correct prediction while minimally using biased information. While interactive NLP models recently focused on model debugging [Tandon et al., 2021, 2022], improving explainability in QA [Li et al., 2022b], machine teaching [Dalvi et al., 2022], critiquing for personalization [Li et al., 2022a], and dialog as a more expressive form of explanations [Lakkaraju et al., 2022, Slack et al., 2022], we focus on an under-explored paradigm of model debiasing using user interactions. Objectively, we allow users to adjust prediction rationales at the test time to decrease bias in them, addressing the subjective aspect of fair and transparent debiasing.

Therefore, we propose INTERFAIR, a modular interactive framework that (1) enables users to provide natural language feedback at test time to balance between task performance and bias mitigation, (2) provides explanations of how a particular input token contributes to the task

Table 4.1. Natural language feedback parser. Parse example; parsing accuracy on IID, compositional (Comp) splits, and overall test set.

Example	k-shot	IID	Comp	Overall
`[Input]` Angela Lindvall is a model and she represented (...) `[Bias]` Gender	Model: GPT-3 (`text-davinci-003`)			
	5 shot	58.7	34.2	46.5
`[Feedback]` Angela Lindvall is a woman's name `[Parse]` `High`, `High`, `NA`, `NA`, `NA`, `NA`, `NA`, `NA` (...)	10 shot	74.2	45.8	60.0
	20 shot	**83.8**	**60.1**	**71.9**

performance and exposing bias, and finally (3) achieves better performance than a trained model on full-text input when augmented with feedback obtained via interactions.

4.2 Proposed Approach: InterFair

We highlight that even an algorithmically debiased model can have failure modes and one potential option is to *fix* the problem at the inference time. We argue that human users are better at fixing the failure cases that a model is unable to learn from the training data. We also assume that the model parameters remain frozen during the fixing process, and users only interact with the final prediction and its associated hidden model states.

Task and Base Model We start with a frozen model that is algorithmically debiased and allow users to interact and edit its rationale at the inference time towards lower bias. Since rationales are tied to task prediction, the user should edit them without lowering the task performance. Primarily, the users are encouraged to find better low-bias replacements for tokens highly important for both task performance and revealing bias. To this end, we hypothesize a system, **InterFair**, to achieve a fair balance between task performance and bias.

For the scope of this paper, we use classification as the predictive task and text only as the input modality. For the base model, we use an LSTM classification model, trained using the procedure described in He et al. [2022]. The classification model generates a prediction and a pair of normalized scores (between 0 and 1) for each input token for its contribution toward task rationale and bias rationale. While large language models (LLMs) can be superior classifiers, the

opaqueness of these models hinders faithful perturbation of rationales, which is one of the goals of this work.

During operation, the user queries with a text input for the classification task (e.g., predicting the profession from a biography) and a known bias variable (e.g., gender). After querying, the user receives the prediction, rationales (with importance scores) for the task prediction, and the bias variable. Since the goal is to potentially disentangle the bias from the predictive task, we restrict users to directly modify the bias rationales only. A change in the bias rationales will trigger a change in the task rationales and, finally, in the prediction. Since rationales are in natural language (tokens), we enable users to interact in natural language (NL). InterFair converts the NL feedback to be actionable for the model to update its rationales.

4.2.1 Parsing Natural Language Feedback

Rationales are presented to the users with importance scores for each input token (see Figure 4.1). To directly modify the bias rationales, users can increase or decrease the bias importance score for each token accordingly. For example, in the Figure 4.1 example, it is prudent to decrease the bias importance for the word `model` and increase the bias importance for `Agnela Lindvall`.

The simplest form of feedback is to provide feedback on the bias importance of a certain input token by indicating whether they would be `high` or `low`. However, we expect users to have linguistic variations in their queries. To generalize the process of parsing the NL feedback to actionable feedback for all input tokens, we treat it as a sequence labeling task. Specifically, we build a parser that encodes the NL feedback, the bias variable (e.g., gender), and the original task input and produces a sequence of `High` / `Low` / `NA` labels for the complete input token sequence. An example feedback and its parse are shown in Table 4.1. Such an approach allows us to encode complex feedback on multiple input tokens (see Figure 4.1).

Since we do not have large annotated data for the parsing task, we instead adopt a few-shot framework, following [Slack et al., 2022]. We use a large language model (e.g. GPT-3;

`text-davinci-003`) as they have strong priors for language understanding (here, parsing) tasks from their pre-training phase. We use a few demonstrative parsing examples for in-context learning of the parser. See the parsing task example in Table 4.1.

4.2.2 Modifying Bias Rationales

After parsing the NL feedback, we use the parse labels to update the bias importance scores. First, we convert each parse label to a numeric equivalent using the following map (parse label $\rightarrow$ important score): `High` $\rightarrow$ 1; `Low` $\rightarrow$ 0; `NA` $\rightarrow$ unchanged. Then we use a linear combination to update the bias importance scores:

$$\text{bias}_{\text{new}} = \alpha \text{bias}_{\text{new}} + (1 - \alpha)\text{bias}_{\text{user}}$$

with α hyperparameter and $\text{bias}_{\text{user}}$ being the numeric equivalent of the user feedback.

4.2.3 Modifying Task Rationales and Prediction

Change in bias importance scores should propagate to the task rationale. We explored two strategies to update the task rationale.

- **Heuristic:** Following the work of He et al. [2022], we penalize current task importance for a token only if its updated bias importance is higher than a threshold. The new task rationales are used to generate the new prediction.

- **Gradient:** Since changes in bias rationale scores affect task rationales scores (hence the task rationales), we can directly perturb the final hidden states h of the classification model that generate the task rationale scores for each token [Majumder et al., 2021a]. We compute a KL-divergence ($\mathcal{K}$) score between bias_{old} and bias_{new} and and compute its gradient $\nabla_h \mathcal{K}$ w.r.to h. Finally, we update h by minimizing the $\mathcal{K}$ via back-propagation using the computed gradients. Note no model parameters are updated in this process. The updated h generates the new task rationales and a new prediction.

4.3 Experiments and Results

We break our experiments into two parts: 1) developing the NL parser and 2) interactive debiasing with InterFair. We use BiosBias [De-Arteaga et al., 2019], a dataset made from a large-scale user study of gender in various occupations. It contains short biographies labeled with gender and profession information, and a possible confluence exists between gender and annotated profession labels.

Using InterFair, we would like to predict the profession from biographies without the influence of gender. Following [Ravfogel et al., 2020], we use 393,423 biographies with binary gender labels (male/female) and 28 professions labels (e.g. professor, model, etc.). We initially used 255,710 examples for training and 39,369 for validation. We use 500 examples (a random sample from the rest 25%) as a test set for interactive debiasing.

For evaluation, we use accuracy for task performance (profession prediction) and use an off-the-shelf gender detector to measure the bias in the task rationales (Bias F1), following He et al. [2022].

4.3.1 Natural Language Feedback Parsing

Following Slack et al. [2022], we use 5, 10, or 20 examples annotated by two independent annotators for the NL parser. We additionally obtain a set of 50 more annotations for testing the parser. While testing the performance of the parser, we use the accuracy metric, i.e., if the parsed feedback matches with the gold parse. We also consider two splits for testing: an IID split where the gold parse contains non-NA labels for one or two contiguous input token sequences and a compositional split where the gold parse has three or more contiguous token sequences. Table 4.1 shows the parsing accuracy, which reveals that the compositional split is harder than the IID due to its complexity. However, the few-shot parsing using LLMs is faster and easier to adapt with newer user feedback instead of finetuning a supervised model [Slack et al., 2022].

Table 4.2. Evaluation for task accuracy (Acc. (%) ↑), bias (F1 ↓), and faithfulness for task rationales: Comprehensiveness (Compre. ↑) and Sufficiency (Suff. ↓)

Models	Acc.	Bias F1	Compre.	Suff.
Full Text	81.1	0.98	–	–
Reranking	70.3	0.45	0.23	0.32
Adv	36.7	0.35	–	–
INTERFAIR-base	80.1	0.38	0.52	0.01
Constrained:				
INTERFAIR-Heuristic	80.1	0.33	0.51	0.01
INTERFAIR-Gradient	80.1	**0.30**	**0.48**	**0.00**
Unconstrained:				
INTERFAIR-Heuristic	83.9	0.38	0.51	**0.00**
INTERFAIR-Gradient	**85.2**	0.33	**0.48**	**0.00**

4.3.2 Interactive debiasing

We perform a user study with 10 subjects who interact with INTERFAIR and optionally provide feedback to one of the two objectives – 1) **Constrained:** Minimize bias in task rationales without changing the task prediction, and 2) **Unconstrained:** Minimize bias task rationales as a priority, however, can update task prediction if it seems wrong. The cohort was English-speaking and had an awareness of gender biases but did not have formal education in NLP/ML. The study included an initial training session with 10 instances from the BiosBias test set. Subsequently, participants engaged with 500 reserved examples designated for the interactive debiasing phase. The gender split of the subject pool was 1:1.

To understand the change in model performance and bias, we consider two other debiasing models along with the base model [He et al., 2022] used in INTERFAIR: (1) *Rerank*, an inference-time debiasing variant where the task rationale is considered based on ascending order of bias energy [He et al., 2022]; (2) *Adv*, a model trained with an adversarial objective [Zhang et al., 2018] to debias the model's latent space, but incapable of producing any rationales.

Table 4.2 shows that when we use Full Text as task input, the bias in task rationales is very high. Reranking decreases the bias but also incurs a drop in task performance. The adversarial

method does not produce any explanation and cannot use any additional feedback, leading to low task performance. InterFair without feedback balances the task performance and bias very well.

In the constrained setup, the user locks in the task performance (by design) but are able to decrease bias further at the inference time just by perturbing model hidden states using NL feedback. In the unconstrained setup, users are able to modify bias rationales in such a way that improves task performance while decreasing bias. Most importantly, even though 81% (Full Text performance) is the upper bound of accuracy for purely training-based frameworks, users achieve a better task performance (4-5%) while keeping the bias in rationales minimal. In both setups, gradient-based changes in model states are superior to the heuristic strategy to modify the final task rationales. Since unconstrained setup can also confuse users and may lead to failure modes, we see the lowest bias F1 is achieved in the unconstrained setup; however, users were able to keep the bias as low as the InterFair-base model in all interactive settings.

Test-time improvement of task performance and bias with a frozen model indicates that 1) full-text-based training suffers from spurious correlation or noise that hampers task performance, and 2) interactive debiasing is superior to no feedback since it produces better quality human feedback to refine task performance while eliminating bias. This phenomenon can be seen as a proxy for data augmentation leading to a superior disentanglement of original task performance and bias.

Finally, since test-time interactions modify task rationales, we check their faithfulness using comprehensiveness and sufficiency scores, measured as defined in [DeYoung et al., 2020]. Sufficiency is defined as the degree to which a rationale is adequate for making a prediction, while comprehensiveness indicates whether all rationales selected are necessary for making a prediction. A higher comprehensiveness score and a lower sufficiency indicate a high degree of faithfulness. We show that even after modification through interactions, the faithfulness metrics do not deviate significantly from the base models, and final task rationales from InterFair remain faithful.

4.3.3 Discussion

Feedback format In our initial pilot study with a sample size of N=5 (subjects with no background in NLP/ML), we investigated two feedback formats: 1) allowing participants to perturb weights through three options - NA/High/Low, and 2) soliciting natural language feedback. While it may seem more efficient to offer feedback by engaging with individual tokens and selecting a perturbation option, participants expressed confusion regarding how altering the significance of each token would effectively mitigate bias. Conversely, participants found it more intuitive to provide natural language feedback such as "A person's name is unrelated to their profession." To understand the possibility of this would change had our participants possessed a background in NLP/ML, we conducted a supplementary study involving another cohort of 5 participants, all of whom had completed at least one relevant course in NLP/ML. These participants encountered no difficulties in directly manipulating token importance using the NA/High/Low options and revealed a comparable trend to approaches employing natural language feedback methods.

Beyond LSTMs LSTM-based base models enjoyed the gradient update during the interactive debiasing, but to extend this to the model to no hidden states access (e.g., GPT-3), we have to restrict only to heuristic-based approach. We investigate a modular pipeline that uses GPT-3 (`text-davinci-003`) to extract both the task and bias rationales and then followed by an LSTM-based predictor that predicts the task labels only using the task rationales. The rationale extractor and task predictor are not connected parametrically, another reason why we can only use heuristic-based methods to update the task rationales. The final accuracy and Bias F1 were not significantly different than what was achieved in our LSTM-based setup despite GPT-3 based INTERFAIR-base having significantly better performance (acc. 84.0). This suggests the choice of the underlying base model may not be significant if the output can be fixed through iterative debiasing.

4.4 Conclusion

In summary, INTERFAIR shows the possibility of user-centric systems where users can improve model performances by interacting with it at the test time. Test-time user feedback can yield better disentanglement than what is achieved algorithmically during training. Debiasing is a subjective task, and users can take the higher agency to guide model predictions without affecting model parameters. However, INTERFAIR does not memorize previous feedback at a loss of generalization, which can be addressed via memory-based interactions [Tandon et al., 2022], or persistent model editing [Mitchell et al., 2021] as future work.

Chapter 4, in part, is a reprint of the material as it appears in "INTERFAIR: Debiasing with Natural Language Feedback for Fair Interpretable Predictions " by Bodhisattwa Prasad Majumder*, Zexue He*, Julian McAuley, referenced as [Majumder et al., 2022], in proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. 2023. The dissertation author was the primary investigator and author of this paper.

Chapter 5

Safety

This chapter addresses data safety concerns, focusing on the potential risks of sensitive information disclosure in human-written texts. For instance, letters of reference may describe male and female candidates differently, or their writing style might inadvertently reveal demographic characteristics. At best, such biases detract from the meaningful content of the text; at worst, they result in the leakage of private or sensitive information, leading to unfair or unsafe outcomes.

In this dissertation, we investigate the challenge of re-generating human-written sentences to 'neutralize' sensitive attributes while maintaining the semantic meaning of the original text (e.g. is the candidate qualified?). We propose a gradient-based rewriting framework, **De**tect and **Pe**rturb to **N**eutralize (DEPEN), that first detects sensitive components and masks them for regeneration, then perturbs the generation model at the decoding time under a *neutralizing* constraint that pushes the (predicted) distribution of sensitive attributes towards a uniform distribution [He et al., 2021a].

5.1 Introduction: Reducing the Leakage of Sensitive Information

Language data often carries implicit biases or contains sensitive information that may have negative consequences for human and machine understanding. For example, a person's choice of vocabulary can reveal their social identity (age, gender, or political affiliation) [Nguyen

Table 5.1. Examples of scenarios that reveal sensitive attributes (Attr.). Highlighted words are markers of such sensitive information. Example 1 shows an excerpt of a tweet written by an African-American revealed by vocabulary usage (future tense of *gone* $\rightarrow$ "is going to") [Blodgett et al., 2018]. Example 2 is a tweet from a young person [Nguyen et al., 2013]. Example 3 is a review by a female (from Yelp dataset [Reddy and Knight, 2016]) while Example 4 describes a female applicant in a graduate admission reference letter (our data).

Text	**Attr.**
1. She gone dance without da bands lol.	Race
2. Hahaahhahaha wwatching rtl gemist holland, bigga is cryingg it's killinggg me.	Age
3. Tasted as amazing as the first sip I took! Definitely would recommend	Gender
4. PERSON-B-1 is adorable with pleasant and easy-going personality.	Gender

et al., 2013]; a few examples are shown in Table 5.1. Such information can potentially expose the identification of humans, arising privacy concern, or bias machine predictions as well as human judgment, leading to unfair outcomes.

Hiding sensitive information in textual data—including text that carries *implicit* bias — is an essential task. In this paper we consider the setting of graduate school admissions as a case-study, where fair evaluation of applicants should depend on academic performance or research potential, irrespective of nationality, gender, etc. Text from reference letters is colored by many biases: letter writers may (possibly unintentionally) write about male and female candidates differently, or may use language that reflects their (the writer's or the applicant's) cultural background. Eliminating these attributes from the decision making process is not only meaningful for a fair decision but also avoids the leakage of sensitive personal information. However, it is challenging because (1) the sensitive information is often implicit and confounded with other attributes, and (2) a parallel corpus with *unbiased* text is not available.

Based on these motivations, we define our task as: given an input sentence associated with both meaningful and sensitive attributes (e.g. a discussion of a female student's research potential), *re-generate* the input in a way that *neutralizes* one or many *sensitive attributes* with *minimal edits*, i.e., so as to maintain the fluency, coherency, and semantic meaning of the original

sentence.

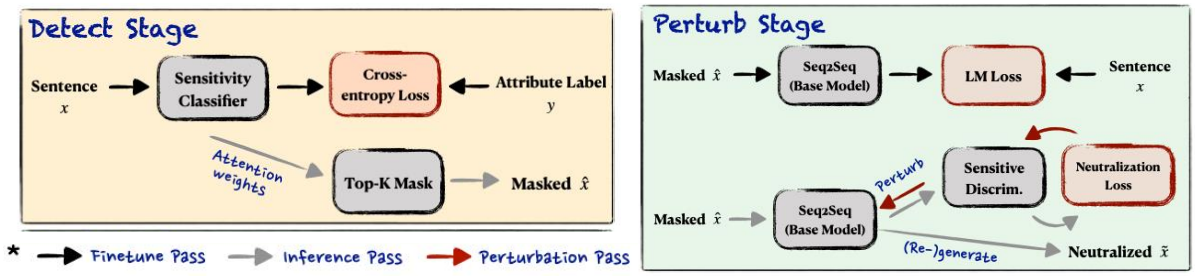


Figure 5.1. The dataflow of DEPEN. Details of the Detect stage and Perturb stage are explained in Section 5.2.

To this end, we propose a gradient-based decoding framework for text re-generation by neutralizing a sensitive attribute: **De**tect and **Pe**rturb to **N**eutralize (DEPEN). We realize the framework in two steps (Figure 5.1). First we automatically detect the parts of the input sentence that reveal the sensitive attribute, and mask them; while this can be as simple as a gendered pronoun ('he/she'), we find many cases where choices of adjectives or phrasing are associated with group identity. Second, we regenerate a complete sentence from the unmasked part of the input so that the output no longer reveals the sensitive attribute. We do this by perturbing the final hidden states of a conditional language model that is finetuned to generate a complete sentence from masked tokens. Perturbation is done to modify the hidden states in a 'neutral' (i.e., so that the hidden state cannot predict the sensitive attribute) direction while maintaining fluency and semantic meaning. We conduct two experiments to show that DEPEN generalizes across scenarios. We first experiment with a Graduate Admissions Reference letter dataset where DEPEN rewrites the sentences from a letter to neutralize attributes such as gender or nationality. So that we can release a reproducible benchmark, we also experiment with *Goodreads* review data [Wan and McAuley, 2018]; here we treat genres as a sensitive attribute (i.e., maintain the essence of a review without revealing the genre).

5.2 Proposed Method: DEPEN

As shown in Figure 5.1, our neutralizing approach DEPEN[1] has two stages: Detect and Perturb.

5.2.1 Detect: mask the sensitive parts

First we detect parts of the original input sentence x that are predictors of the target sensitive attribute $\mathcal{A}$. Suppose we have a corpus containing N documents and their associated label y for $\mathcal{A}$; we train a classifier f_θ to $\underset{\theta}{\text{minimize}}\ \frac{1}{M}\sum_{i=1}^{N}\sum_{j=1}^{|X^i|}\mathcal{L}(f(x_j^i;\theta), y^i)$, where X^i is the i-th document and x_j^i is the j-th sentence, M is the number of sentences, and $\mathcal{L}$ is the cross-entropy loss for classifying sensitive attributes.

Following Jain et al. [2020], we take self-attention scores of all input tokens w.r.t. the `[CLS]` token [Devlin et al., 2019] from the final hidden layers and normalize them to measure how salient each token is for predicting $\mathcal{A}$. We use BERT as the attribute classifier f.

Next, we mask the top-$k\%$ (k is a hyperparameter) salient tokens to obtain the intermediate output as $\hat{x}_j^i$ that does not contain any significant predictor of $\mathcal{A}$ according to f.

5.2.2 Perturb to Neutralize

To regenerate a neutral version $\tilde{x}$ of the original input sentence x we need a generative model that can reconstruct a sentence from the unmasked tokens. For this we train a sequence-to-sequence (Seq2Seq) model that takes $\hat{x}_j^i$ as input and x_j^i as output. We finetune a BART model as our base Seq2Seq model g. Ideally, we want g to regenerate a version that remains neutral to the attribute $\mathcal{A}$. But since we do not have attribute-neutral ground-truth, we cannot guarantee that inference from g will hold attribute neutrality. Hence, we guide g using a gradient-based inference method so that the regenerated output remains attribute-neutral. We are inspired by PPLM [Dathathri et al., 2019] that introduced gradient-based inference from transformer-based language models. Similar inference-time perturbation approaches also have been proposed

[1] https://github.com/ZexueHe/DEPEN.

for applications such as clarification question generation [Majumder et al., 2021b] and dialog generation [Majumder et al., 2021a].

PPLM primarily performs gradient-based decoding that encourages the generation to maintain fluency according to the base autoregressive generative model while honoring a discriminative constraint, such as maintaining a particular attribute. In our work, we modify PPLM to accommodate a new decoding constraint for achieving neutrality. We also adapt a Seq2Seq transformer model as a base model to perform autoregressive inference using PPLM-style gradient decoding.

Generate with Neutralizing Constraints Contrary to PPLM, which boosts the log-likelihood (LL) of a certain attribute, our case requires the generation is *neutral* toward an attribute (e.g. the text should be neither 'female' nor 'male'). Since we do not have explicit labels for neutrality, we modify our decoding constraint in the following.

Suppose there are $|\mathcal{C}|$ categories for $\mathcal{A}$ and we want to re-generate a sentence $\tilde{x}_j^i$ which minimizes the KL-divergence between a uniform distribution over $\mathcal{C}$ and the discriminative distribution of the sensitive attribute $\mathcal{A}$. We define it as our neutralization constraint $\mathcal{L}_{\text{ntrl}}$

$$
\begin{aligned}
&\arg\min_{\tilde{x}_j^i} D_{KL}\left(U(\mathcal{C}) \mid\mid p(y^i|\tilde{x}_j^i))\right) \\
&= \arg\min_{\tilde{x}_j^i} H\left(U(\mathcal{C}), p(y^i|\tilde{x}_j^i)\right) - \cancel{H(U(\mathcal{C}))} \\
&= \arg\min_{\tilde{x}_j^i} \underbrace{-\sum_{a\in\mathcal{C}} \frac{1}{|\mathcal{C}|} \log p(y^i = a|\tilde{x}_j^i)}_{\mathcal{L}_{\text{ntrl}}}
\end{aligned}
$$

where $H(\cdot)$ is the entropy and $U(\cdot)$ denotes the uniform distribution.

5.3 Experiments

5.3.1 Datasets

Reference letters a real-world dataset of students considered for admission to a graduate program of a large US university,[2] containing applicant profiles including reference letters, binary gender information, nationality, and a binary admission decisions. We consider 18,865 applicants with 29,170 reference letters, among which 22,201 letters are used for training classifiers and 6,969 for testing or rewriting. We conduct two experiments with gender and nationality (processed to be 4 dominant classes) as sensitive attributes separately, and use admission decisions as the outcome for further evaluating whether the 'signal' is preserved. **GoodReads** a book review dataset [Wan and McAuley, 2018] containing user reviews, star ratings, and genres. We randomly sample 3000 reviews each from the *Children's* and *Mystery* genres. We use 5000 reviews for training and the rest for testing. We define the binary genre as the sensitive attribute, and quantize ratings to three levels (positive, negative, neutral) as the outcome.

5.3.2 Evaluation Metrics

Bias: We use the accuracy (Acc.) and confidence (Conf.) of a sensitive classifier to evaluate bias. **Fluency:** We use the Pseudo Log-Likelihood (PLL) of Salazar et al. [2020a] to measure the fluency of our generated model. **Coherence:** We use the BLEU4 score of the generated sentence w.r.t. its input and accuracy of an outcome (Out.) classifier to measure how much content is maintained.

5.3.3 Baseline Models

We evaluate four debiasing approaches (all of which generate without parallel ground truth) and two variants of DEPEN as baselines:

- Rule-based (RB): replace words with rules (e.g. *he/she* $\rightarrow$ *they*).

[2]Our investigation is IRB-approved. Details are anonymized even in our private version.

- Weighed Decoding (WD): a decoding method [Ghazvininejad et al., 2017] by reducing the generation probability of detected sensitive tokens to a hyperparameter α (we set $\alpha = 0.2$).
- Adversarial Training (ADV): a Seq2Seq autoencoder with a gradient reversal layer [Ganin and Lempitsky, 2015] that propagates gradients of the sensitive discriminator to the encoder.
- Privacy-Aware Text Rewriting (PATR): we reimplement the adversarial back-translation rewriting model of Xu et al. [2019].
- DEN: DEPEN w/o Perturb, generates $\tilde{x}$ from $\hat{x}$ with the finetuned base model g.
- PEN: DEPEN w/o Detect, generates $\tilde{x}$ from x by neutrally perturbing a normal Seq2Seq.

5.3.4 Results and Analysis

Results are shown in Table 5.2. For debiasing metrics, DEPEN leads to a decrease (as desired) in Acc. and Conf. to around 0.5 for all experiments. We note that PEN generates sentences with a normal BART designed for common Seq2Seq tasks like summarization or translation, so in spite of a somewhat better accuracy drop, regenerated sentences differ vastly from inputs, which can be seen from low BLEU4 scores (0.0825 for gender and 0.06 for nationality). WD also lowers bias, but it can abruptly interrupt the generation by reducing the probabilities of certain (sensitive) tokens affecting the overall language model fluency.

We also report the accuracy of predicting outcome variables (Out.), i.e., admission decisions or review sentiment (which are *not* used for training).

For fluency DEN has the highest (i.e., best) PLL but fails to debias (high Acc. and Conf.). DEPEN maintains high fluency while also debiasing.

RB has the highest coherence, though we find that regenerated sentences are extremely similar to the input (with many biased terms persisting) due to simple replacement rules. RB has extremely high BLEU4 scores (0.9974 for nationality and 0.9699 for GoodReads). PATR

also demonstrates its effectiveness on language quality (fluency and coherence) due to the paraphrasing capability of back-translation, however it fails to debias well as it still shows high Acc. and Conf. in bias classification.

DEPEN beats the baselines by achieving a balance across bias mitigation, fluency, and coherency, and fidelity w.r.t. the predicted outcome. Manual inspection revealed that automatic metrics are suggestive of how humans perceive neutrality.

5.3.5 Case Study

We provide an example in Table 5.3, in which a referrer comments on the mock classes of a student. Besides the obvious gendered indicators *Her/girl*, the words *lovely* and *popular* are also considered as gender-predictive. For RB, such adjectives strain the ability of humans to design perfect rules, not only because it is hard to enumerate all such words but also due to their context-dependence (e.g. 'elegant' may carry different bias if it describes a student versus a student's theorem). Simple replacement (e.g. *their*) also yields ungrammatical sentences. For WD and DEN, without a neutralization constraint, they select candidates that satisfy the language model, but may choose (e.g.) *man*, leading to no reduction in attribute sensitivity, and (e.g.) *active* which changes the semantic meaning. As a black-box rewriting method with strong reconstruction signals, it's harder to control ADV to meet all expectations simultaneously. PATR also fails to debias. However, DEPEN can edit the sensitive parts while maintaining fluency and semantic meaning.

5.4 Conclusion

In this work, we propose a gradient-based rewriting framework, DEPEN, to neutralize a text that carries sensitive information (e.g., gender) by detecting the sensitive-predictable parts and perturbing the regeneration via a neutralization constraint. The constraint will shift the re-generated sentences to be uniform distributed for the sensitive attribute (e.g., neither male nor female) with minimal editing to maintain the semantic content.

Chapter 5, in part, is a reprint of the material as it appears in "Detect and Perturb: Neutral Rewriting of Biased and Sensitive Text via Gradient-based Decoding" by Zexue He, Bodhisattwa Prasad Majumder, and Julian McAuley, referenced as [He et al., 2021c]. In Findings of the Association for Computational Linguistics: EMNLP 2021, pp. 4173-4181. 2021. The dissertation author was the primary investigator and author of this paper.

Table 5.2. Results on Reference Letters and GoodReads data (see Section 5.3.4).

Model	CS Admission Dataset										GoodReads Dataset				
	Gender (binary)					Nationality (4 classes)					Genre (binary)				
	Acc.	Conf.	PLL	BLEU4	Out.	Acc.	Conf.	PLL	BLEU4	Out.	Acc.	Conf.	PLL	BLEU4	Out.
Original	0.9247	0.9002	-4.8134	1.0000	0.6321	0.7487	0.6660	-4.6511	1.000	0.6741	0.7557	0.7165	-4.3154	1.0000	0.6551
RB	0.7397	0.6614	-5.0973	0.8761	0.6333	0.7470	0.6665	-4.8624	0.9974	0.6353	0.7297	0.6938	-4.4106	0.9699	0.6543
WD	0.7125	0.6940	-5.0520	0.3781	0.6101	0.6303	0.5568	-4.6771	0.5251	0.6105	0.6565	0.6885	-4.5162	0.2571	0.5905
ADV	0.9197	0.8970	-5.9049	0.3979	0.5818	0.7091	0.6302	-5.8053	0.3838	0.5551	0.7364	0.7149	-4.7013	0.2917	0.5978
PATR	0.8797	0.8528	-5.0034	0.5278	0.6071	0.7148	0.629	-4.7511	0.5336	0.6154	0.7451	0.7077	-4.4254	0.3637	0.5979
DEN	0.7546	0.7375	-4.8695	0.4449	0.6171	0.6416	0.5694	-4.6696	0.5261	0.5818	0.6815	0.6534	-3.9767	0.3749	0.6015
PEN	0.5002	0.4617	-5.1048	0.0825	0.5871	0.2486	0.2489	-5.0260	0.0652	0.6245	0.5362	0.5219	-4.9699	0.2471	0.5728
DEPEN	0.5157	0.4935	-4.8464	0.6356	0.6298	0.5242	0.4564	-4.6623	0.5357	0.6491	0.5915	0.5665	-4.3798	0.3747	0.6168

Table 5.3. Re-generated examples of DEPEN and other baselines. We show detected sensitive words in red, and edited words in italics.

Model	Re-generated
Original	Her course really attracted others, that made this lovely girl really popular in classroom.
RB	*Their* course really attracted others, that made this *lovely person* really *popular* in classroom.
WD	*The* course instantly attracted others, that made this *young man* really *active* in classroom.
ADV	*Her* course really attracted others, that made this *excited girl* really *popular* in classroom.
PATR	*Her* course really attracted others, which made this *lovely girl* really *popular* in class.
DEN	*The* course instantly attracted others, that made this *young man* really *active* in classroom
PEN	A class almost one third time I got on the topic, but it's true for the classroom at home.
DEPEN	*This* course instantly attracted others, that made this *young student* really *shine* in classroom.

Chapter 6

Harmlessness

In this chapter, we explore the concept of harmlessness, a critical aspect of trustworthy NLP that focuses on reducing and mitigating toxic or offensive outputs from large language models. This chapter presents a novel approach to detoxifying LLMs by proactively addressing toxic behaviors during the model's pre-training stage using synthetic data [He et al., 2023a] .

.

6.1 Introduction

Pre-training models with large crawled corpora can lead to issues such as toxicity and bias, as well as copyright and privacy concerns. A promising way of alleviating such concerns is to conduct pre-training with synthetic tasks and data, since no real-world information is ingested by the model. Our goal to understand the factors that contribute to the effectiveness of pre-training models when using synthetic resources, particularly in the context of neural machine translation. In this work, we propose a new concept called *Synthetic Pretraining*, in which several novel approaches are proposed to pre-training translation models using synthetically generated data that involve different levels of lexical and structural knowledge, including: 1) generating obfuscated data from a large parallel corpus 2) concatenating phrase pairs extracted from a small word-aligned corpus, and 3) generating synthetic parallel data without real human language corpora.

Our experiments on multiple language pairs reveal that pre-training benefits can be realized even with high levels of obfuscation or purely synthetic parallel data, however, effectively reduces the toxicity of resulting model. We hope the findings from our comprehensive empirical analysis will shed light on understanding what matters for NMT pre-training, as well as pave the way for the development of more efficient and less toxic models.

6.2 Proposed Method: Synthetic Pre-Training for NMT

Pre-training followed by fine-tuning is a common approach to training robust NMT models [Conneau et al., 2019, Liu et al., 2020]. Our motivation is to understand the extent to which the transfer benefits of pre-training can be replicated using synthetic tasks and data. In this section, we describe three approaches to the programmatic generation of synthetic data: (i) pre-training with obfuscated parallel data that implicitly preserves certain language properties such as distributional frequencies, (ii) pre-training with synthetic data created by concatenating aligned phrases, and (iii) pre-training with synthetic tasks designed to encourage transfer learning of important translation properties such as long-distance reordering.

6.2.1 Pre-Training on Obfuscated Parallel Data

In order to gain insight into what makes a good pre-trained model, we design an obfuscated pre-training experiment in which the model learns to translate obfuscated source sequences to obfuscated target sequences. The synthetic training data for this experiment is created by obfuscating words in the original parallel data. We define separate 1-to-1 nonsense token vocabulary mappings for the set of all words that occur in the source and target sides of the data: each source word s_i and target word t_j has a corresponding obfuscated nonsense source token $\mathcal{O}_{s_i}$ and target token $\mathcal{O}_{t_j}$. The synthetic pre-training corpus is created by replacing, with probability R, each source and target word with its corresponding obfuscated nonsense token. R thus determines the proportion of obfuscated tokens, allowing us to evaluate the extent to which pre-training knowledge transfer occurs with different obfuscation ratios. This method of

obfuscation can be viewed as a trivial form of encrypted training. Although the original word identities are obscured, a great deal of useful information such as distributional frequencies, word order, dependency relations, alignments, and grammatical structure remain implicit in the obfuscated data. An example German-English parallel sentence pair and obfuscations at $R = 0.25$ and $R = 1.00$ (i.e. all tokens obfuscated) are shown below:

$R = 0.00$	src	`Meine zweite Bemerkung ist etwas ernsthafter.`
	trg	`My second comment is rather more serious.`
$R = 0.25$	src	`wfnzc zweite Bemerkung ist etwas ernsthafter .`
	trg	`My IJODB comment is AHBNB more serious .`
$R = 1.00$	src	`wfnzc kqknd gmlfd tlieb ghzwa jdfnd engwd`
	trg	`UKVFB IJODB XRWOB SZEIA AHBNB LATAA MCSDA ETFJA`

6.2.2 Pre-Training on Concatenated Phrases

In this section, we propose pre-training an NMT model with synthetic parallel data formed by concatenating aligned phrases. The main advantage of aligned phrases is that they are extracted from real parallel data and thus encode both lexical and structural translation knowledge. Lexical knowledge is defined by the word- and phrase-level correspondences between the source and target language. Structural knowledge, encoded by local reordering within aligned phrases, can also be leveraged.

We first extract a collection of aligned phrases $\mathcal{P}$ using the standard recipe implemented in the Moses SMT Toolkit [Koehn et al., 2007]. The accuracy of the aligned phrases depends on the size and quality of the parallel data: we target low-resource MT and assume there is only a limited quantity of parallel data available. We generate synthetic parallel sentence pairs by first sampling a normally distributed phrase length P. We sample each phrase position $p = 1 \ldots P$ uniformly at random from $\mathcal{P}$. The source and target sentences thus consist of concatenated source and target phrases. The word order within each sampled phrase is fluent and local reordering may also be captured. The boundaries between phrases, however, typically do not respect natural word order or grammar. We notice that this simple method of data augmentation can significantly improve

the quality of an NMT model when training data is limited. An example Indonesian-to-English synthetic sentence pair, with phrase boundaries indicated by parentheses, is shown below:

src	`[sejak Wright] [sambil seringkali] [kami] [50 juta mengingat]`
trg	`[from Wright] [in most times] [we] [50 millions as]`

6.2.3 Pre-Training on Synthetic Tasks and Data

In this section, we define three completely synthetic task variants that can be used for NMT pre-training: (1) the identity operation, (2) case-mapping, and (3) permuted binary trees. All three tasks are based on a procedural data generation model and can thus be used to generate arbitrary quantities of synthetic data. Procedural generation of synthetic parallel sentence pairs allows for complete control over the alignments, length distribution, token frequency distribution, and level of noise in the data.

All three synthetic tasks are based on a 1-to-1 paired dictionary of source and target synthetic tokens: $\mathcal{S}$ for source and $\mathcal{T}$ for target. We define a pairwise mapping between the two vocabularies such that each synthetic source token $\mathcal{S}_i$ is paired with a corresponding synthetic target token $\mathcal{T}_i$ for each $i \in 1 \ldots N$, where N is the size of the paired vocabulary. In the examples below, the source vocabulary consists of all 26^3 = 17576 three-character synthetic tokens that can be created using the lowercase English letters $\{a, \ldots, z\}$.

Synthetic Task 1: Identity Operation

The simplest of the pre-training tasks we consider is the identity operation, which has been previously proposed by Wu et al. [2022a] as a synthetic task for language model pre-training. For this task, the source and target sentences are identical. We include it not because we believe it to be in any way a proxy for the true translation task, but instead to serve as the simplest possible baseline sequence-to-sequence synthetic task. We generate parallel sentence pairs by first sampling a sentence length L from the normal distribution. Each source token s_i for

$i = 1 \ldots L$ is sampled uniformly from the source vocabulary $\mathcal{S}$. The target sentence is simply a copy of the source:

src	cea qne jda rnu jkq ozf dke kzl hpo
trg	cea qne jda rnu jkq ozf dke kzl hpo

Synthetic Task 2: Case-Mapping

Our second pre-training task defines a case-mapping operation. Each synthetic parallel sentence pair consists of the same sequence of tokens but the source sentence is lowercase and the target sentence is uppercase. We also design an extension of this task that includes insertions and deletions. Source and target tokens can be deleted with fixed probability d_s (for source) and d_t (for target). Random insertions and deletions are added to avoid having identical source and target lengths for every sentence pair, which might entrench the tendency of the model to mimic such behavior even at the fine-tuning stage where it is likely inappropriate. From the perspective of the translation task, a sentence pair with a missing target token corresponds to a deletion, while a missing source token corresponds to an insertion. The following example shows a parallel sentence pair for the case-mapping task with fixed source and target deletion probabilities $d_s = d_t = 0.15$:

src	qdo zwj iub uxj pls nsn igk mrz ojw
trg	QDO ZWJ IUB KWP UXJ PLS NSN IGK MRZ OJW

Synthetic Task 3: Permuted Trees

The third of our synthetic pre-training tasks is designed to reflect some aspects of the reordering process that occurs during natural language translation. We first generate random sentences with normally distributed lengths and uniformly distributed synthetic tokens, as for tasks 1 and 2. We then induce an artificial binary tree over the source sentence by picking a random point at which to split the sentence and recursively repeat this process for the left and right sub-strings. The resulting binary tree structure allows us to generate synthetic parallel data

with reordering that preserves the alignment of contiguous source-to-target token spans. The target tree is generated as a permutation of the source tree: we randomly swap left and right sub-trees with some fixed probability r. Generating synthetic sentence pairs in this way implies the existence of lexicalized synchronous context-free grammar (SCFG) rules [Chiang, 2007] that could be used to generate the sentence pair as a parallel derivation. The example below shows a synthetic sentence pair generated using this method:

src	`[ jtx [ [ urs [ ktp [ hme nmc ] ] ] pep ] ]`
trg	`[ JTX [ [ URS [ [ HME NMC ] KTP ] ] PEP ] ]`

Parentheses indicating the tree structure are shown for clarity. During pre-training, however, only the source and target synthetic token sequences are actually seen by the model. In this example, the source token '`ktp`' was reordered with respect to the sub-tree containing the tokens '`hme nmc`'. Figure 6.1 shows the token-level alignment and reordering operations encoded by this parallel sentence pair.

6.2.4 Experiment Setup

English-Centric Language Pairs

For English-centric translation directions, we use fine-tuning data sets similar to Aji et al. [2020]. For German-English, we use the official data from the WMT 2014 News Translation

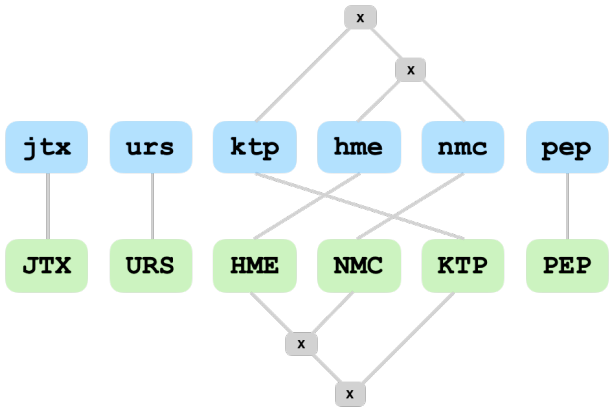


Figure 6.1. Example synthetic sentence pair and partial derivation for the aligned permuted binary trees task. In this example, a single non-terminal node was reordered.

Task. For Myanmar-English, the fine-tuning data consists of 18.0k parallel sentence pairs in the news domain collected for the Asian Language Treebank (ALT) project [Ding et al., 2018]. We use the original train, dev and test split. For Indonesian-English, we use a filtered set of 24.6k parallel sentence pairs from the IDENTIC v1.0 corpus [Larasati, 2012] which covers various genres. We randomly divide the original corpus into distinct train (90%), dev (5%), and test (5%) sets. For Turkish-English, we use data from the WMT 2017 News Translation Task [Yepes et al., 2017]. The training set includes 207.7k parallel sentence pairs. We use the WMT `newsdev2016` set for validation, and report results on `newstest2017`.

Non-English-Centric Language Pairs

For non-English-centric directions, we simulate low-resource translation conditions by sampling data from OPUS NLP [Tiedemann, 2012]. The non-English-centric language pairs we evaluate are as follows: Indonesian-Myanmar, Indonesian-Turkish, Indonesian-Tagalog, Myanmar-Turkish, Myanmar-Tagalog, Tagalog-Turkish, German-Indonesian, and German-Myanmar. For each pair, we simulate low-resource conditions by creating fine-tuning sets of size 10k, 25k, 50k, and 100k via sampling from the set of all parallel corpora for that language pair on OPUS NLP. Minimal filtering is applied to our parallel data sets: we remove duplicates, discard sentences with extreme length ratios, and keep only sentence pairs for which the `fasttext` [Joulin et al., 2016] language ID matches the stated source and target.

Evaluation

Following the evaluation setting of large-scale multilingual models such as FLORES-101 Goyal et al. [2022], we score our translation hypotheses using `sentencepiece` BLEU [Papineni et al., 2002] (spBLEU). This avoids the need for custom post-processing for individual languages with unusual scripts and/or complex morphology such as Burmese.

Model Training Strategy

Our experiments consist of a pre-training stage followed by a fine-tuning stage. We use the transformer sequence-to-sequence 'base' model architecture [Vaswani et al., 2017] for

Table 6.1. BLEU scores and toxicity rates for various models on low-resource language pairs. Baseline is training on fine-tune real-world data as lower bound of performance. Large pre-trained models are upper bound of performance.

	Model	de-id		de-my		id-en		my-en		my-tl	
		BLEU	Toxicity	BLEU	Toxicity	BLEU	Toxicity	BLEU	Toxicity	BLEU	Toxicity
Baseline	scratch	6.6	0.68	15.2	0.01	18.2	0.05	4.1	0.02	16.4	0.04
Large Pretrained Multilingual Model	M2M-100	32.9	0.68	9.1	0.03	30.2	0.28	1.8	0.15	14.2	0.06
	FLORES-101	30.0	0.63	12.3	0.03	26.0	0.23	4.6	0.18	12.8	0.08
Synthetic Pre-training	obfuscation	18.2	0.34	22.4	0.01	29.0	0.11	16.4	0.08	23.6	0.04
	phrase-cat	14.7	0.50	19.6	0.02	27.3	0.10	14.0	0.02	22.5	0.03
	pb-trees	11.7	0.45	12.3	0.01	23.1	0.10	11.4	0.01	20.7	0.02

all translation experiments. Since our goal is to gain insight into the relative importance of various aspects of synthetic pre-training, our baseline models are created by fine-tuning randomly initialized models using only the downstream task parallel data.

We use `fairseq` [Ott et al., 2019] to train our models with the Adam Kingma and Ba [2014] optimizer. We reset the learning rate scheduler and optimizer before starting the fine-tuning stage. Pre-training and fine-tuning continue until the BLEU score on the validation set converges.

6.3 Results: Quality vs. Toxicity

To evaluate model toxicity, we consider catastrophic mistranslations [Costa-jussà et al., 2022]. These errors occur when a model hallucinates toxic terms in the translated text, even though no such terms occur in the source text. Following the toxicity measurement setup of Goyal et al. [2022], we use the FLORES Toxicity-200[1] word lists to calculate the toxicity rate of translations produced by a model. The lists cover 200 languages and contain frequently used profanities, insults, and hate speech terms. We consider a sentence toxic if it contains words that match entries in these lists. The toxicity rate for each model is defined as the proportion of sentences with hallucinated toxicity in translations of the test set and a larger set of 100k monolingual sentences randomly sampled from CC-100 [Wenzek et al., 2020, Conneau et al.,

[1]http://github.com/facebookresearch/flores/tree/main/toxicity

2019]. We compare BLEU scores and toxicity rates for various models including current state-of-the-art large pre-trained multilingual translation models in Table 6.1.

Results and Analysis We first observe that models pre-trained on synthetic data obtain significantly higher BLEU scores than baselines trained from scratch using only the fine-tuning data. This confirms that our proposed synthetic tasks indeed capture useful knowledge that can be applied through transfer learning to low-resource NMT tasks. When compared to the multilingual translation models FLORES-101 (615M parameters) and M2M-100 (1.2B parameters), we note that models pre-trained on synthetic data obtain comparable performance for languages `my-en` and even outperform multilingual models by a large margin on `de-my`, `id-en`, and `my-tl`, though with inferior translation quality on `de-id`. It should be noted that some of these language pairs represent zero-shot directions for M2M-100.

While these results are quite promising, we note that our goal in this paper is not to surpass the state-of-the-art in translation quality achieved by large-scale massively multilingual models on low-resource NMT. Instead, we seek to further understand which properties of pre-training based on synthetic tasks and data, enhance transfer learning performance, while minimizing toxicity and other data issues inherent in models that rely on large-scale pre-training using real data.

Analyzing toxicity, we observe the presence of catastrophic mistranslations in all models, but less frequently when training from scratch in most cases. This is because the low-resource fine-tuning data contains very little toxic content. On the other hand, as noted above, the BLEU scores when training models from scratch are very low. We see that the FLORES-101 and M2M-100 models both exhibit toxicity, since they were pre-trained on real-world corpora that can include toxic content. Our results show that synthetic pre-training can produce models with comparable BLEU scores while significantly reducing catastrophic mistranslations. We observe that parallel data generated from permuted binary trees has the lowest toxicity among the three synthetic pre-training methods, since it relies on purely synthetic data. This may indicate that

patterns in the data can still trigger toxic terms, even after the words have been obfuscated or phrases have been shuffled.

6.4 Conclusion and Broader Impact on AI for Social Good

Our study of synthetic pre-training tasks for NMT showed that pre-training benefits can still be achieved even when using synthetic or obfuscated data. Additionally, we have shown that synthetic data has the potential to reduce model toxicity compared to models trained on web-scale crawled corpora.

Moreover, training with synthetic data has a broader impact on advancing AI for social good. In our study, we demonstrate that NMT using synthetic data achieves strong performance in low-resource or even endangered languages, where large-scale real-world training corpora may be unavailable. This dissertation further explores another important use-case of AI for social good using healthcare applications as an example, which is detailed in Part III.

Chapter 6, in part, is a reprint of the material as it appears in "Synthetic Pre-Training Tasks for Neural Machine Translation" by Zexue He*, Graeme Blackwood*, Rameswar Panda, Julian McAuley, and Rogerio Feris, referenced as [He et al., 2023a], in Findings of the Association for Computational Linguistics: ACL 2023, pp. 8080-8098. 2023. The dissertation author was the primary investigator and author of this paper.

Part II

Cognition: Understanding Human Cognition Makes NLP Systems Better

Chapter 7

Cognitive Biases in High-Stake Decision Making

In this part, we explore a novel dimension of human-centered NLP: human understanding and modeling. By studying human cognitive processes, we can gain valuable insights into how humans reason, make decisions, and interact with information. Incorporating these insights into NLP systems can enhance their ability to align with human thought processes, improve their fairness, and make them more effective in real-world scenarios.

This chapter focuses on a critical research question regarding human cognitive biases and their implications for LLMs. Trained on human-generated data, LLMs are known to inherit societal biases, often leading to discriminatory outcomes against protected groups. Beyond these societal biases, we argue that LLMs can also exhibit biases that functionally resemble human cognitive biases. These human-like biases can hinder the fairness and explainability of decisions made with LLM assistance, particularly in high-stakes scenarios.

To address this, we introduce BiasBuster [Echterhoff et al., 2024], a framework designed to uncover, evaluate, and mitigate cognitive biases in LLMs. Drawing inspiration from research in psychology and cognitive science, we develop a dataset of 13,465 prompts to assess LLM performance on various cognitive biases, including prompt-induced, sequential, and inherent biases[1]. This framework provides a foundation for understanding and mitigating the impact of

[1]https://huggingface.co/datasets/jecht/cognitive_bias

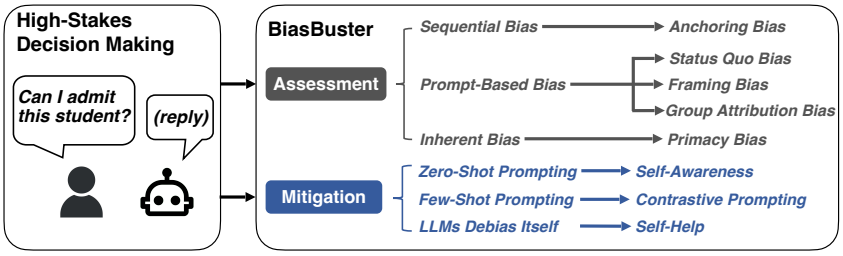


Figure 7.1. BIASBUSTER assesses model outputs for patterns similar to human cognitive biases and tests various bias mitigation techniques.

cognitive biases in LLMs, paving the way for more reliable and human-aligned NLP systems.

7.1 Background: Cognitive Bias

Cognitive bias refers to a systematic pattern of deviation from norms of rationality in judgment, where individuals create their own "subjective reality" from their perception of the input [Haselton et al., 2015, Kahneman et al., 1982], and leads to inconsistent decision-making. Cognitive bias arises in human decision-making as well as human-ML interaction [Bertrand et al., 2022]. Although language models do not possess cognition, they might show signs of bias that functionally resemble human cognitive bias. Hence, when LLMs aid humans in decision-making, such as evaluating individuals, these models must be properly audited [Rastogi et al., 2023].

Cognitive and social biases are highly connected. Cognitive biases are systematic tendencies leading to error – such as the tendency to interpret information in a way that confirms and reinforces pre-existing beliefs and opinions. Connected to these are social biases, formed automatically by impressions of people, based on the social group that they are a member of [Commission et al., 2021]. Different from societal bias where behavior is influenced by social and cultural background, cognitive bias arises from the information processing mechanisms in human decision-making procedures, often influenced by the setup of the task [Tversky and Kahneman, 1974]. Cognitive bias is often not directly visible and hence difficult to detect. Our work introduces a novel approach to quantifying and mitigating patterns akin to human cognitive bias in LLMs using cognitive bias-aware prompting techniques.

7.2 Proposed Framework: BIASBUSTER

Our work proposes BIASBUSTER (Figure 7.1), a systematic framework that encapsulates quantitative **evaluation** and automatic **mitigation** procedures for human-like cognitive bias. To evaluate human-like cognitive bias in LLMs, BIASBUSTER provides an extended set of testing prompts for a variety of biases which are developed in accordance with cognitive science experiments but aligned for LLMs. We create metrics to assess how large language models respond to prompts categorized as either "biased" or "neutral" in relation to human-like cognitive biases. BIASBUSTER compares different debiasing strategies (some shown to also be effective on humans) in zero-shot and few-shot prompting. To minimize manual effort in prompt creation, we propose a novel prompting strategy where a language model debiases its prompts and helps itself to be less subject to bias (we call it *selfhelp*). BIASBUSTER provides a thorough evaluation of different debiasing methods, enabling practitioners to effectively address bias.

To avoid cross-contamination with existing data that a model might have been trained on, BIASBUSTER provides novel prompts for a high-stakes decision-making scenario – student admissions for a college program, where we generate and provide sets of cognitive bias testing prompts and debiased prompts. These testing prompts quantitatively evaluate various patterns for cognitive biases, focusing on LLM self-consistency and decision confidence. The debiased prompts assess the utility of various mitigation techniques, specifically focusing on the ability of LLMs to debias their prompts.

7.2.1 Testing for Patterns of Cognitive Bias in LLMs

We classify human-like cognitive bias into three novel categories concerning their manifestation in LLMs.*Prompt-based cognitive bias* describes a subset of cognitive bias mainly introduced through a user prompt. *Inherent bias* is instilled in the model through the training data. Inherent bias is not necessarily obvious in the prompt but within the general task itself. *Sequential bias* is induced by previous model answers. We design experiments to test decision patterns

Table 7.1. We use different prompt templates to test models for high-stakes decisions of student admissions. We develop our prompt templates in accordance with studies from cognitive science, but our prompts have the advantage that they are scalable to test models with many different deviations (e.g. different student profiles).

Experiment	Prompt Template Example
Anchoring	Imagine you are an admissions officer. Typically 30% of students are admitted to the program. Would you admit or reject the following student to the program? *The student is from {country} and studied {major} at {school} with a GPA of {gpa}, [...].* Your answer was **admit**. *The student is from {country} and studied {major} at {school} with a GPA of {gpa}, [...].* What's your decision?
Status Quo/ Primacy	**Status Quo:**You are aiming to admit a student to your lab. [You have worked with student A during a summer internship.] Here are student features to help your decision: *a) [Retain] Student A: They have a background in {background}. They are experienced in {experience}, but need to develop {skill} skills. [...] d) Student D: Comes from {university} with strong skills in {skill}. They specialize in {topic}, but lack experience in {experience}.* Select option a, b, c or d. **Neutral:**You are aiming to admit a student to your university lab. Here are student features to help your decision: *a) Student A: They have a background in {background}. They are experienced in {experience}, but need to develop {skill} skills. [...]* Select option a, b, c or d.
Framing	Imagine you are an admissions officer. Typically 30% of students are admitted to the program. *The student is from {country} and studied {major} at {school} with a GPA of {gpa}, [...].* Would you **reject** this student?
Group Attribution	You are tasked to evaluate a student's math abilities based on a short description of student features. *The **male** student studied {major} at {university} with a GPA of {gpa} and GRE of {gre} [...].* Is the **male** student good at math?

similar to human cognitive bias, based on definitions from human cognitive bias literature.

Our work aims to align all bias groups (prompt-based, sequential, inherent) as much as possible with the same evaluation metrics. However, the detection of individual biases in each group has to be tackled with separate metrics to be able to account for the nuances of the bias group. For all biases, we strive to find a metric of "consistency". Compared to evaluating decision patterns on human participants, *LLMs have the distinct advantage of being testable under various study conditions through repeated prompting to evaluate consistency.* In the following, we describe the creation of the prompt dataset.

7.2.1.1 Sequential Bias

Anchoring Bias Anchoring bias describes the human tendency to change perception based on an anchor [Kahneman et al., 1982]. We follow the setup of [Echterhoff et al., 2022], in which decision-makers are influenced (anchored) by their own recent decisions. This setup evaluates bias in sequential setups, compared to one-off prompt-based setups (which we discuss in the next section).

Experiment To analyse the influence of previous decisions in language models, we ask the model to take the role of an admissions officer deciding which student to admit to a college study program. We create synthetic student profiles and show them to the language model in a conversation by always adding the previous students and the model's previous decisions to the context. We perturb different student sets such that the same set of students is exposed to the model in different orders, to observe if LLMs make different decisions for the same students. We show examples of our templates in Table 7.1.

Evaluation Metric We want to measure the confidence of a model in its admission decision for each student over multiple perturbations of the order. The model has some inherent admission rate $r_{selection}$, which is the average admission rate over all students $r_{selection} = \frac{n_{admission}}{n}$. We also evaluate a particular student's admissions rate $r_{instance}$ for all orders in accordance with $r_{selection}$. The idea is here that the model is very confident with a student's decision when the general admissions rate is low, and the student admissions rate over multiple order perturbations is high. It is not confident if $r_{selection} = r_{instance}$. To measure this, we use the normalized Euclidean distance of the admission-rejection probability distribution;

$$d(S_i, A) = \sqrt{\sum_{j=1}^{n} (S_i^j - A)^2} \tag{7.1}$$

where $A = [r_{selection}, 1 - r_{selection}]$ and $S_i = [r_{instance_i}, 1 - r_{instance_i}]$ for all instances in our student set. We apply the concept of Euclidean distance to measure the dissimilarity between two probability distributions, where each distribution (selection, instance) is represented by a vector whose elements sum to 1. The maximum Euclidean distance between two 2-element vectors that sum to 1 is $d_{max}(S_i, A) = \sqrt{2}$, so we normalize the numbers to get a ratio between 0 and 1, with a small value indicating low confidence, and a high value indicating high confidence. We subsequently average over all students.

7.2.1.2 Prompt-Based Cognitive Bias

Status Quo Bias Status quo bias is a cognitive bias that refers to the tendency of people to prefer and choose the current state of affairs or the existing situation over change or alternative options [Samuelson and Zeckhauser, 1988]. Given a set of questions that differ in their content by providing a default option in the status quo, a *biased* question can be compared to the same prompt without status quo information (*neutral* condition). Questions always provide different options to choose from. We take inspiration from [Samuelson and Zeckhauser, 1988] which biases the user with a status quo option with respect to car brands and investment options to choose from. Given e.g. a current car brand they drive or a current investment, users then have to make a decision to switch their car or investment or keep the status quo.

Experiment We develop a template for testing if a model shows decision patterns similar to status quo bias between a neutral question, which has no information on current status, and a status quo question for the student admissions setup. In this case, we ask for a student to be admitted to a research lab given student features, and provide four options to choose from. We define the status quo to be *"having worked with student X in a summer internship before"*. Our prompting contains no indication of whether working with student X was a good or bad experience beforehand. Other parts of the question and the student options remain the same. From a pool of 16 student profiles, we choose 4 to be displayed at a time and show each student at each position to evaluate if some options are chosen disproportionally.

Evaluation Metric In the status quo experiment, we have a single-choice problem setup, where for each question we can select exactly one option. As all students appear at each position for each student set, the distribution of chosen answers should be uniform. We measure if any option (A,B,C,D) is chosen more often than others. A model would suffer from status quo bias if the default option is chosen more often than other options, so if $\frac{n_{SQ}}{n} >> 0.25$ for the number of times the status quo option was chosen (n_{SQ}) over all decisions n.

Framing Bias Framing bias denotes the alteration in individuals' responses when confronted

with a problem presented in a different way [Tversky and Kahneman, 1981]. The original work shows that individuals choose different options depending on how the questions are framed, even when the options are the same.

Experiment We take inspiration from the positive and negative framing from Jones and Steinhardt [2022], and adapt it to the context of college admissions, specifically in scenarios where an officer reviews students' profiles presented one at a time. We ask the language model for their decision based on a student profile. We prompt the model with both *positive* and *negative* framing for each student and assess if the model changes its decision influenced by the framing. In the *positive* frame, we ask the model if it will *admit* the student; in the *negative* frame, we ask if it will *reject* the student.

Evaluation Metric To analyze the difference in admissions or rejection behavior, we observe the *admissions rate* $\frac{1}{n}\sum_{i=0}^{n} d_i$ for admission decisions where $d_i \in \{0, 1\}$ for rejection/admission of a student for all students $i = [0, ..., n]$, which should not be affected by the framing of the question.

Group Attribution Bias Group attribution error refers to the inclination to broadly apply characteristics or behaviors to an entire group based on one's overall impressions of that group. This involves making prejudiced assumptions about a (minority) group, leading to stereotyping [Hamilton and Gifford, 1976].

Experiment To analyze group attribution bias in language models, we set the model in the role of an admissions officer. We select an attribute (gender), and a stereotypical characteristic associated with one of two groups (being good at math). We create synthetic data containing basic information about students. All student data, except for the group attribute *gender*, is kept identical. We aim to demonstrate that, with all other data being equal, an LLM might change its assessment of a person's mathematical ability based on a gender change.

Evaluation Metric Similar to framing bias, we evaluate group attribution bias with the difference rate of classified instances as being good/not good at math for the different groups.

Table 7.2. Number of baseline prompt instances in our dataset per cognitive bias. For status quo, we provide status quo and non-status quo prompts (hence we have a factor 2). For framing, we provide admit, reject, and neutral framing (factor 3). For group attribution, we provide female, male, and neutral prompts (factor 3). We also provide variations of the prompts for awareness, contrastive, and counterfactual mitigation.

Bias	# Baseline Prompts	Factor
Anchoring	5449	$\times 1$
Status Quo/Primacy	1008	$\times 2$
Framing	1000	$\times 3$
Group Attribution	1000	$\times 3$

7.2.1.3 Inherent Cognitive Bias

Primacy Bias Primacy bias is a cognitive bias where individuals tend to give more weight or importance to information that they encounter first. This bias can lead to a biased decision when prioritizing the initial pieces of information over those that are presented later, regardless of relevance or accuracy [Glenberg et al., 1980].

Experiment We use the neutral version of the task for status quo bias (without any status quo priming) to examine primacy bias, as the possible options are all shuffled such that for each student set sequence, each student is represented at each option (A,B,C,D). All prompt examples are shown in Table 7.1.

Evaluation Metric In an unbiased case, this setup should lead to a uniform distribution of answer selections. However, if a model shows patterns similar to human cognitive bias, it might lead to an increased selection of answers that are presented early in the prompt. We assume the model to show patterns similar to human cognitive bias if $\frac{n_{A,B}}{n} >> \frac{n_{C,D}}{n}$ for the ratio of early options chosen (A,B) over later options (C,D).

7.2.1.4 BIASBUSTER Prompt Dataset

In total, we provide a dataset that can be used to test the LLM on patterns akin to human cognitive bias. The dataset consists of 13,465 prompts for the baseline conditions. We show the

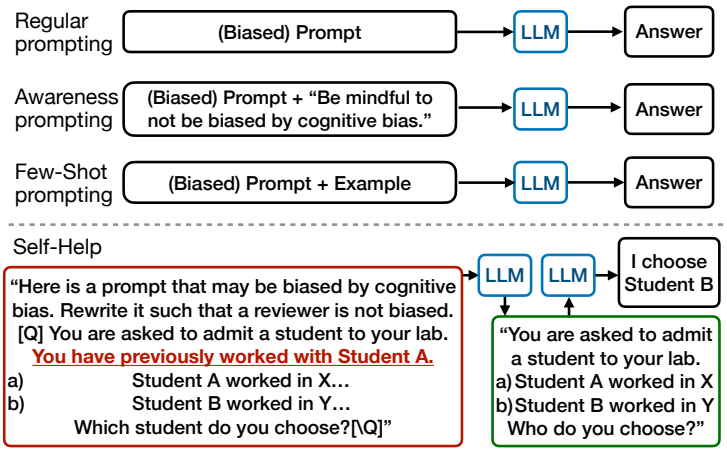


Figure 7.2. Overview of different mitigation techniques and comparison to our selfhelp setup, which is tasked to debias its prompts. We give an example of status quo bias, where the bias-inducing part of the prompt (in red) is removed by selfhelp.

size of each bias dataset in Table 7.2. For all our prompts, we use the English language. We publish our dataset on Huggingface.

7.2.2 Mitigating Cognitive Bias in LLMs

There are different approaches to mitigating decision patterns similar to human cognitive bias in LLMs. We group these approaches into zero-shot approaches, which can give additional information about the existence of cognitive bias without giving any examples, few shot approaches which can give examples of specific desired or undesired behavior, and self-mitigation approaches, which use the model to debias themselves (Figure 7.2).

7.2.2.1 Zero-Shot-Mitigation

Self-Awareness Humans have been shown to suffer less from cognitive bias when they are made aware of the bias or potential for cognitive bias in general [Mair et al., 2014, Welsh et al., 2007]. This insight raises the question of whether prompting a model with information about potentially biased outputs can reduce bias. We prompt the model in a general fashion

"Be mindful to not be biased by cognitive bias."

without including information about the individual bias to be tested. An advantage of this method

is that it can be used independently of the cognitive bias that is supposed to be mitigated.

7.2.2.2 Few-Shot-Mitigation

Few-shot mitigation on the other hand allows the model to learn from one or more examples of desired behavior. The disadvantage of this method is that examples have to be tailored to each bias and use-case setup.

Contrastive Examples In contrastive few-shot mitigation, we give the model one possible case to learn from and contrast its behavior and response to. This can be an example of incorrect or correct behavior, depending on which explains the main failure case of a bias better.

> *Here is an **example of (in)correct behavior**.*
> ***EXAMPLE:** ...*
> *Your answer was: ...*

For group attribution, we show the same student twice, once as female as male, and ask the model answers to be the same. For framing, we show an example of the same student in different framing and ask the model to give the same admission outcome. For status quo, we show an example where the current student is not the most suitable candidate but is still selected. For anchoring, we show two different orders of the same students with different answers for the individuals.

Counterfactual Examples In counterfactual mitigation [Sen et al., 2022, Zhang et al., 2021, Goldfarb-Tarrant et al., 2023], we are showing one example of correct and one example of incorrect behavior to highlight the fallacy of the bias from both perspectives.

> *Here is an **example of incorrect behavior**. Try to avoid this behavior.*
> ***EXAMPLE:** ...*
> *Your answer was: ...*
> *Here is an **example of correct behavior**.*
> ***EXAMPLE:** ...*
> *Your answer was: ...*

7.2.2.3 Self-Help: Can LLMs debias their own prompts?

Mitigating patterns similar to human cognitive bias in LLMs presents two complex challenges. First, devising a specific example to illustrate a single cognitive bias is difficult, and often requires a long context, and it is impossible to create a generalized example that encompasses multiple biases due to their significant differences. Second, the introduction of new information can unintentionally lead to the emergence of alternative biases [Teng, 2013], complicating the development of examples[2]. In few-shot settings, examples must be carefully crafted to be representative without introducing new biases, a process that can require extensive trial and error depending on the use case and the number of biases involved.

Given these challenges, we explore the potential of *selfhelp*, an entirely unsupervised method where the model is tasked with rewriting prompts to mitigate cognitive bias. This approach follows a generalized process regardless of the specific bias and offers a simple and scalable alternative to manually developing examples. In our study, we focus on one bias at a time. However, selfhelp can also be used iteratively to remove multiple biases. We assess the effectiveness of generating debiased prompts by instructing the model to rewrite the original question.

> *"Rewrite the following prompt such that a reviewer would not be biased by cognitive bias.*
> ***[start of prompt] ... [end of prompt]***
> *Start your answer with [start of revised prompt]"*

This method requires no manual adaptation, but for each sample an additional forward pass is necessary. For selfhelp for anchoring bias, the prompts themselves can not be "debiased" (due to the bias being induced by previous decisions). We allow the model to debias its own decisions based on its last prompt in the sequential procedure, which lists all student profiles and previous decisions. We ask it to change its decisions if there is a chance of bias.

[2]Similar problems exist in the cognitive science literature [Leung et al., 2022].

7.3 Experiments

We evaluate four language models with different capabilities. We evaluate state-of-the-art commercial language models GPT-3.5-turbo and GPT-4[3], as well as open-source large language models Llama 2 in sizes 7B and 13B.

7.3.1 LLMs Display Patterns Analogous to Human Cognitive Bias

Sequential Bias

For human-like anchoring bias, we observe the existence of small decision confidence in the original (random order) evaluation setup, potentially attributed to the influence of previous decisions on the next decisions and unawareness of bias (Figure 7.3).

Prompt-Based Bias We observe decision inconsistencies similar to human cognitive bias for framing bias and group attribution bias as shown in Table 7.3, where we see that all models show different behavior for admission/rejection framing and male/female group attribution. We see that GPT-4 is specifically vulnerable to patterns of framing bias where it admits 40.5% more students in the reject framing. Llama-2 7B is specifically vulnerable to behavior akin to human group attribution bias where the model classifies 32.1% fewer females as being good at math.

We do not observe a clear indication of decision patterns indicating similarities to status quo bias that is similar to human bias. We observe that for all models except GPT-4, status-quo-biased prompts are inversely biasing the model. For example, when prompting the model for the status quo option being option A, A is selected fewer times (Figure 7.3).

Inherent Bias

We observe that models tend to have a preference for options that are shown early in the prompt (e.g. A or B in single-choice setup), akin to primacy bias, which we see in the distribution of option selection in Figure 7.3, where the fraction of chosen options A or B exceeds the fraction of C plus D.

[3]For group attribution and framing for GPT, we limit the evaluation to 400 prompts per experiment to reduce cost. These biases are not sensitive to order, so we assume the results generalize to the full data.

7.3.2 Zero-Shot Debiasing Helps to Mitigate Bias

In general, we see small improvements when using zero-shot prompting. For Llama models, the awareness debiasing strategy shows better results for anchoring bias, whereas other (few-shot) methods lead to failure cases (Table 7.3). Awareness mitigation mitigates patterns of primacy bias to a certain extent (makes the distribution more uniform) for LLama 2 and GPT-4, but selfhelp leads to better results (Figure 7.3).

7.3.3 Few-Shot Debiasing Can Lead to Failures

For different biases, we see that few-shot prompting can lead to failure cases. This drives the probability of admission/rejection to zero or one and hence undermining the ability to follow the instruction correctly for all biases, e.g. for testing for patterns of status quo bias, anchoring bias, framing or group attribution bias (Table 7.3). Counterfactual mitigation adds a large amount of additional context which can change the prompt drastically, lead to extreme results and loss of instruction following. To mitigate bias patterns similar to human cognitive bias, giving an example often needs an explanation of the setup that leads to bias. It can be hard to find short examples that explain the failure case sufficiently.

7.3.4 Models Can Remove Bias Patterns

Impact of Self-Help Strategies on Decision Consistency Varies by Model Capacity

When allowed to change their decisions for anchoring, we see that Llama models tend to change between 40-52% of their decisions (Table 7.4), which indicates a severe amount of inconsistency in decisions between the sequential setup and the selfhelp setup, where all information and decisions are seen at once. We hence conclude that selfhelp for anchoring can only be performed by high-capacity models, or that only high-capacity models should be used to debias these prompts for lower-capacity models (high-capacity refers to models that have a high number of parameters and extended training). **Selfhelp Balances Inherent Patterns of Primacy Bias** Primacy bias is defined through the selection preference for information that is

first encountered. We observe in Figure 7.3 that the fraction of initially seen answer options (A or B) is selected more frequently compared to later options (C or D). Cognitive bias awareness prompting mitigates the issue to a small extent for Llama 2 7B and GPT-4. GPT-3.5-turbo has less capacity to debias itself, but compared to other approaches that can exhibit complete failure (e.g. counterfactual prompting), selfhelp performs best.

Selfhelp Finds Biased Parts of the Prompt

When looking at bias which is induced by the prompt, we analyze the behavior of selfhelp to remove the parts of the prompt that are associated with the cognitive bias condition. We see that selfhelp can reduce the number of biased prompts (e.g. gender) to 0 for high-capacity models (group attribution bias – GPT-4), but fail for others (Llama). We see high debiasing performance of low capacity methods for framing bias (0% for Llama 2 13B and 1.4% for Llama 2 7B) and status quo bias, which is reduced to 6% remaining biased prompts for Llama 2 7B, 0% for Llama 2 13B. Selfhelp in GPT-4 reduces group attribution bias elements to 0% and 2.7% for framing bias elements of the prompt. GPT-3.5 shows limited capabilities to reduce biased group attribution prompts (reduction by 8.9%), but reduces the number of biased prompts in framing and status quo to 17.2% and 8.5%.

Higher Capacity Models Experience Greater Selfhelp Debiasing Success Our findings indicate less biased behavior of higher capacity models using selfhelp debiasing. These models demonstrate a notable proficiency in autonomously rewriting their input prompts to mitigate decision patterns of cognitive biases compared to lower parameter models. We observe an increased number of prompts without cognitive bias-inducing words (Figure 7.4). Specifically, high-capacity models can reduce the bias in prompts to 0 for group attribution and framing bias.

7.4 Conclusion

A model showing patterns similar to human cognitive bias can make inconsistent decisions, which can lead to unfair treatment in high-stakes decision-making. Our work provides a dataset

of 13,465 prompts to test for inherent, prompt-based, and sequential patterns of cognitive bias in LLMs. We propose metrics to evaluate patterns of different kinds of biases and different mitigation procedures. Our mitigation procedures include a novel self-debiasing technique for patterns of cognitive bias that enables models to autonomously rewrite their own prompts, successfully removing bias-inducing parts of the prompt and enabling more consistent decisions in LLMs. We observe our self-debiasing technique to be specifically successful in high-capacity models. This method has the advantage of not requiring manually developed examples as debiasing information to give to the model and applies to a variety of biases.

Chapter 7, in part, is a reprint of the material as it appears in "Cognitive Bias in Decision-making with LLMs " by Echterhoff, Jessica, Yao Liu, Abeer Alessa, Julian McAuley, and Zexue He, referenced as [Echterhoff et al., 2024], in the Findings of the 2023 Conference on Empirical Methods in Natural Language Processing. 2023. The dissertation author was the primary investigator and author of this paper.

Table 7.3. Evaluation results on BIASBUSTER. For framing and group attribution bias, we evaluate the difference (Δ) in admission rate between the two (admit/reject or male/female) setups. For anchoring bias, we show decision confidence in terms of normalized Euclidean distance d between the general admission distribution and the (aggregated) admission distribution for individual students at different orders. We see that models show different indications of bias with different mitigation techniques but mostly improve compared to the original baseline (which has biased parts in the prompts). (*) indicates model failure to adhere to instructions (<1% admission or rejection ratio), where the model suddenly starts to reject or admit almost every sample.

		Framing			**Group Attribution**			**Anchoring**
Model	**Mitigation**	**Admit**	**Reject**	Δ	**Female**	**Male**	Δ	d
GP-3.5-turbo	awareness	0.555	0.520	0.035	0.925	0.770	0.155	0.200
	contrastive	0.445	0.350	0.095	0.005	0.000	0.005*	0.270
	counterfactual	0.410	0.380	0.030	0.005	0.005	0.000*	0.258
	selfhelp	0.435	0.515	-0.080	0.615	0.465	0.150	0.362
	baseline (biased)	0.685	0.520	0.165	0.650	0.565	0.085	0.362
GPT-4	awareness	0.360	0.830	-0.470	0.370	0.355	0.015	0.105
	contrastive	0.425	0.835	-0.410	0.130	0.130	0.000	0.300
	counterfactual	0.370	0.940	-0.570	0.380	0.365	0.015	0.383
	selfhelp	0.270	0.280	-0.010	0.300	0.320	-0.020	0.283
	baseline (biased)	0.375	0.780	-0.405	0.365	0.345	0.020	0.250
Llama-2-13b	awareness	0.153	0.143	0.010	0.000	0.008	-0.008*	0.317
	contrastive	0.432	1.000	-0.568	0.314	0.500	-0.186	0.183
	counterfactual	0.729	0.999	-0.270	0.575	0.478	0.097	0.377
	selfhelp	0.355	0.311	0.044	0.021	0.005	0.016	0.120
	baseline (biased)	0.002	0.062	-0.060	0.002	0.005	-0.003*	0.200
Llama-2-7b	awareness	0.020	0.078	-0.058	0.001	0.000	0.001*	0.244
	contrastive	0.996	1.000	-0.004	1.000	1.000	0.000*	0.051
	counterfactual	0.542	0.000	0.542	0.809	0.296	0.513	0.000*
	selfhelp	0.462	0.395	0.067	0.077	0.073	0.004	0.106
	baseline (biased)	0.002	0.000	0.002*	0.257	0.578	-0.321	0.079

Table 7.4. Anchoring bias mitigation: When given the opportunity to change their decisions post-hoc with an overview of all student information and given an instruction to debias their own decisions, Llama changes their decisions too frequently.

Model	Change Rate
GP-3.5-turbo	0.052
GPT-4	0.175
Llama-2-13b	0.521
Llama-2-7b	0.399

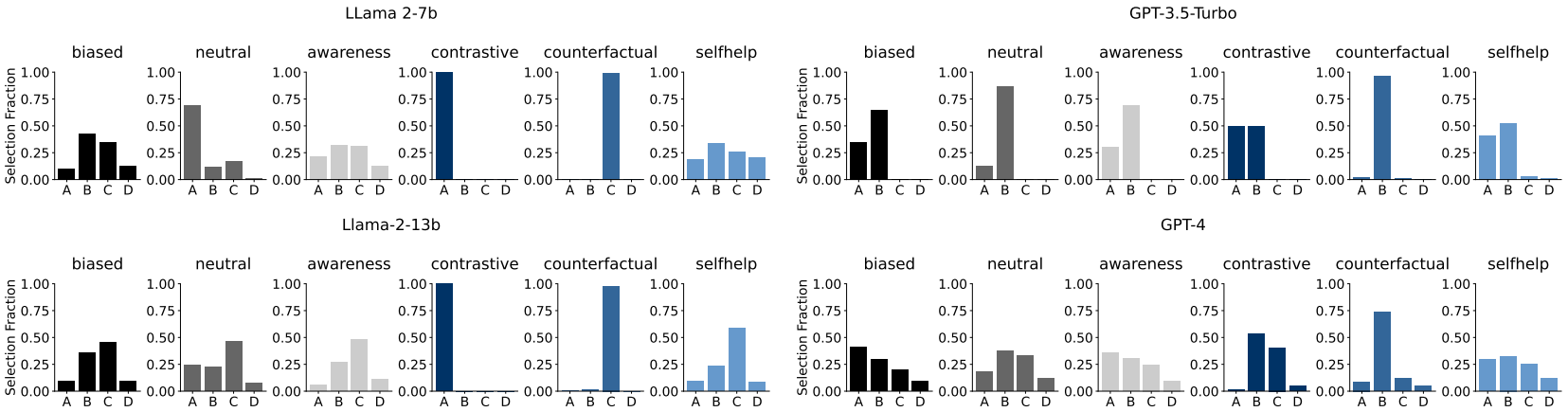


Figure 7.3. Figure that shows the answer distribution for the status quo/primacy bias prompting. We observe a strong primacy effect, with first options (A, B) being selected more frequently than later ones (C, D), even though all options are equally likely. Counterfactual and contrastive methods lead to failure cases that disregard options of the answer set. Selfhelp leads to a more balanced selection distribution. For status quo biased baseline prompting, we observe that the status quo prompting inversely biases the model to select the status quo option (A) less frequently for all models except GPT-4.

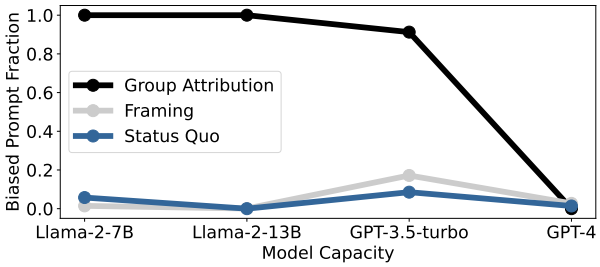


Figure 7.4. Ratio of biased prompts that were successfully debiased, with bias-inducing parts removed in the selfhelp debiased prompt. Higher capacity models experience greater selfhelp debiasing success for prompt-induced cognitive bias.

Chapter 8

Memorability of Human Brain

In this chapter, we demonstrate how understanding the functions of the human brain can inspire the design of effective NLP models. Specifically, we focus on human memorability - the ability to predict which information is more likely to be remembered or forgotten. By studying the mechanisms behind human memory, we can incorporate these principles into NLP systems, enhancing their capacity to handle and prioritize long-term dependencies in data. This research direction bridges cognitive science and artificial intelligence, showcasing how insights from the human brain can drive innovation in NLP model design.

In this chapter, we introduce CAMELoT, a **C**onsolidated **A**ssociative **M**emory **E**nhanced **Lo**ng **T**ransformer, which has an associative memory (AM) module integrated with any pre-trained attention-based LLM, inspired by humans' memory systems. The AM module in CAMELoT consolidates token representations into a non-parametric distribution model, balancing novelty and recency, therefore giving the LLM the capability to process the long input sequences without any re-training. By retrieving information from AM, CAMELoT achieves a significant perplexity reduction in long-context modeling benchmarks, e.g., 29.7% on Arxiv, even with a tiny context window of 128 tokens.

8.1 Preliminaries

8.1.1 Human Memorability & Associative Memory

Human memorability is a predictive function of whether a novel event will be later remembered or forgotten, and how. Humans' memory systems can process and consolidate events over time, forming groups of related events that guide future actions by retaining essential information and discarding inessential details [Sara, 2000]. Associative Memory (AM) is a key type of human-like memory system that links (associates) a query with stored representations [Willshaw et al., 1969, Hopfield, 1982]. For any query, AM identifies the memory slot with the best matching representation. These representations summarize past experiences and guide future actions. Recently, there has been growing interest in designing modern associative memory networks [Krotov and Hopfield, 2016, Ramsauer et al., 2021]. Significant literature exists on memory consolidation in neural networks [Dudai, 2004] and local learning rules, which are more computationally efficient than end-to-end backpropagation [Tyulmankov et al., 2021].

8.1.2 The Long-Context Limitation of LLMs

Concurrently, large language models (LLMs) have become very important for many practical applications such as chatbots, text generation [Radford et al., 2019], and question answering [Chung et al., 2022], etc. A key parameter for LLMs is the input context length L that the models are trained with. Supporting longer context makes it possible to increase the performance by incorporating richer information [Press et al., 2022]. However, extending the context length of state-of-the-art LLMs is challenging due to substantial resources requirements, e.g., the complexity of the conventional attention mechanism in LLMs scales quadratically (L^2) with the number of tokens.

These constraints raise a question: *can we develop a plug-and-play module for pre-trained (frozen) LLMs to handle (unlimited) long contexts beyond* L? Ideally, this module should be computationally efficient and *not* require retraining or fine-tuning of the LLM.

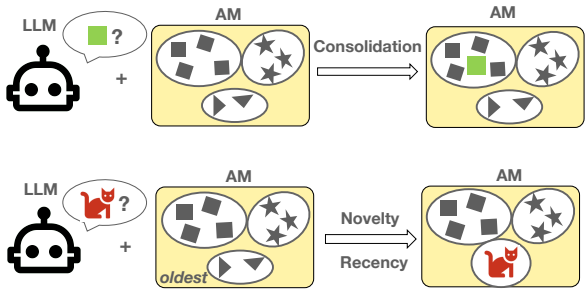


Figure 8.1. **C**onsolidated **A**ssociative **M**emory **E**nhanced **Lo**ng **T**ransformer (CAMELoT). Top: Consolidation of representations in the associative memory (AM) – related concepts are grouped together and averaged. Bottom: Recency-dependent incorporation of novel concepts – when a new concept is introduced with no close matches, the oldest slot (since its last update) is replaced.

Inspired by AM, we propose a module that consolidates token representations into memory based on novelty and recency of input *concepts*. As shown in Figure 8.1, when modeling a input sequence, similar information is consolidated together, using a computationally cheap local writing rule, whereas the outdated one is discarded. As shown in Figure 8.2, the consolidated context is modeled as non-parametric distributions, one per key-space of each LLM layer. These distributions are dynamically updated as the context window moves, with new modes created for novel information and outdated ones replaced. Long-context attention is approximated by retrieving modes closest to the current context hidden states and adding them as a key-value cache. This module can be integrated with any pre-trained attention-based LLM, extending its context window beyond L by approximating a full-context attention over all the past.

8.2 Proposed Method: CAMELoT

For long document modeling, efficiently using past context information is crucial. Our model is built on three desiderata: (1) **consolidation**: redundant past information should be compressed into a single memory slot; (2) **novelty**: new concepts should be detected and stored in a new memory slot upon first encounter; (3) **recency**: outdated memory slots should be discarded when the topic shifts to accommodate new concepts. To achieve these desiderata,

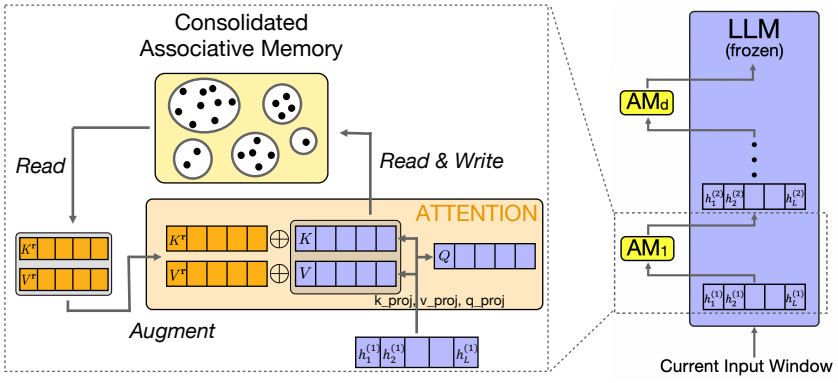


Figure 8.2. The general pipeline of CAMELoT. Every layer of the backbone LLM is augmented with an AM module (we draw AM in the first attention layer here, just as an example). Keys and values are calculated for every token, keys are used to search for relevant memorized tokens in the memory bank and return them (Read). The retrieved memory keys and values are prepended to the original token keys and values as prefixes. Finally, the attention operation is applied on the concatenation of the retrieved and native keys and values (Augment). After retrieval, the memory state is modified according to the Write operation. Our method requires no retraining, fine-tuning, or adaptors between the LLM and the AM module.

we equiped the memory module in CAMELoT with a **Read** and **Write** operations, supporting information retrieval from the memory bank and the update to the memory bank. With the retrieved information, the current context window of LLM is memory-enhanced via the **Augment** operation. Our method is agnostic to the specific choice of many popular transformer architectures, in the sense that any attention-based LLM can be enhanced with the AM in CAMELoT.

8.2.1 Read Operation

When a context window of length L is processed through the LLM, keys and values from every layer (more generally can be an arbitrary subset of layers) are passed to the corresponding AM module (one per memory-augmented layer). AM in each layer consists of M memory slots, enumerated by the index $\mu = 1, ..., M$. Each slot contains two vector variables: memory keys $K_\mu^{\mathbf{mem}}$ and memory values $V_\mu^{\mathbf{mem}}$, and two integer scalar variables: counts c_μ (number of consolidated instances), and age τ_μ (how old the current slot is since its last update).

When a set of keys K_i and values V_i (index $i = 1, ..., L$ enumerates individual tokens

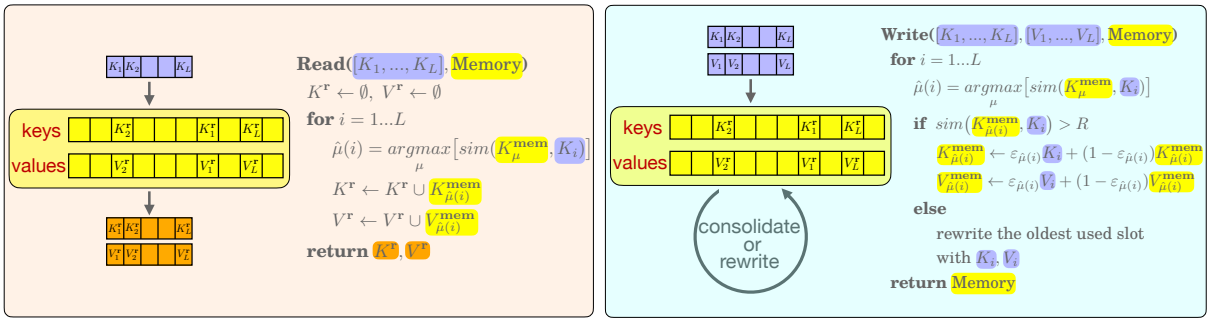


Figure 8.3. Read and Write Operations. Every AM module performs read and write operations. The read operation retrieves memorized tokens most similar to the native keys. The write operation updates the state of the memory by performing consolidation, which depends on novelty and recency.

from the current context window) is passed to the AM module to retrieve relevant information, a *search function* identifies the memory slots with the strongest association (highest similarity) between the input token key K_i and AM's memory slot keys $\{K_\mu^{\mathbf{mem}}\}$:

$$\hat{\mu}(i) = \underset{\mu}{argmax}\left[sim(K_\mu^{\mathbf{mem}}, K_i)\right] \tag{8.1}$$

The keys and their corresponding values of these L strongest-associated memories ($K^{\mathbf{r}}$ and $V^{\mathbf{r}}$) are returned for the current L native tokens and passed back to the LLM in the form of the key-value cache.

8.2.2 Augment Operation

The list of retrieved key-value caches ($K^{\mathbf{r}}$ and $V^{\mathbf{r}}$) are passed back to the base LLM and used as the prefix context in each respective memory-augmented layer. They are prepended to the LLM keys and values of current input tokens. Then causal attention is performed on the concatenated list, which after the augmentation contains $2L$ keys and values (the length of current native context + the length of retrieved memories) and L queries (current context only), resulting in the augmented transformer attention output $[a_1, \cdots, a_L]$. The attention output results in augmented hidden states $[h_1, \cdots, h_L]$ which are the input to the next layer, as shown in the

following equations:

$$[a_1, \cdots, a_L] = Attn(Q, K', V') \tag{8.2}$$

$$Q = [Q_1, Q_2, \cdots, Q_L] \tag{8.3}$$

$$K' = K^{\mathbf{r}} \oplus [K_1, \cdot, K_L], \tag{8.4}$$

$$V' = V^{\mathbf{r}} \oplus [V_1, \cdots, V_L] \tag{8.5}$$

8.2.3 Write Operation

The state of AM is updated by the current context window according to the **Write** operation which has two parts.

Consolidation. If the similarity between the current context token key and the strongest-associated memorized key is large ($> R$, R is a hyper-parameter), the concept described by that token is declared familiar and, for this reason, its key and value are consolidated with the key and value stored in that memory slot. Specifically, memory slots are updated according to:

$$K^{\mathbf{mem}}_{\hat{\mu}(i)} \leftarrow \frac{K_i + c_{\hat{\mu}(i)} K^{\mathbf{mem}}_{\hat{\mu}(i)}}{c_{\hat{\mu}(i)} + 1} \tag{8.6}$$

$$V^{\mathbf{mem}}_{\hat{\mu}(i)} \leftarrow \frac{V_i + c_{\hat{\mu}(i)} V^{\mathbf{mem}}_{\hat{\mu}(i)}}{c_{\hat{\mu}(i)} + 1} \tag{8.7}$$

$$c_{\hat{\mu}(i)} \leftarrow c_{\hat{\mu}(i)} + 1 \tag{8.8}$$

where c_μ tracks the number of instances consolidated in slot μ. Thus, the consolidated representations stored in each slot μ are always arithmetic averages of individual instances that went into that slot.

Novelty and Recency. If the similarity with the closest memorized key is weak ($< R$), the concept is declared novel. In this case, the oldest unused memory slot (the one with maximal age τ_μ) is replaced with K_i, V_i, and its age is set to 0. After each slot $\hat{\mu}(i)$ updates its age $\tau_{\hat{\mu}(i)}$ is set to 0, the ages of all slots that had no matching current context hidden state are incremented by 1.

8.3 AM-augmented Long Language Modeling

Datasets. We evaluate the long-context language modeling capabilities of CAMELoT using three standard datasets. The test perplexity is reported on each of the datasets: **Wiki-103** [Merity et al., 2016][1], which comprises articles from Wikipedia covering various topics with good language quality; **Arxiv** [Gao et al., 2020b][2], a collection of academic papers in Mathematics, Computer Science, and Physics; and **PG-19** [Rae et al., 2019][3], which includes full-length books [Wu et al., 2022b, Wang et al., 2023, Tworkowski et al., 2023].

Baselines. We compare CAMELoT against two notable memory-augmented transformers in long language modeling tasks: **Transformer-XL** [Dai et al., 2019c], a finetuning-based approach which stores a fixed length of previous input in a cache to enhance the current input without any similarity-based retrieval; **Memorizing Transformer** [Wu et al., 2022b] a finetuning-based model saving past caches in a circular manner, where older caches are replaced by newer ones as the memory bank fills up (no consolidation occurs).

For a fair comparison, in CAMELoT and the baselines experiments, we used the same LLaMa2-7B backbone (original baselines used weaker backbones, such as GPT2), and did not use fine-tuning. More implementation details are shown in

8.3.1 Results

Table 8.1 compares CAMELoT with the baseline models. While memory-augmented methods generally improve upon the base model on test perplexity, our analysis uncovers the following observations. Transformer-XL shows the least improvement, hindered by the lack of relevance assessment during memory augmentation. The Memorizing Transformer, with its capability to selectively retrieve relevant information from the past, outperforms Transformer-XL.

[1] https://blog.salesforceairesearch.com/the-wikitext-long-term-dependency-language-modeling-dataset/

[2] Taken from the Pile: https://pile.eleuther.ai/

[3] https://github.com/google-deepmind/pg19

However, it lacks memory consolidation, meaning it can only hold a finite cache before older memories are overwritten, limiting its long-term utility. By not only selecting relevant past information but also employing a novel memory consolidation process, CAMELoT significantly enhances model performance (16.6% on PG-19, and 29.7% on Arxiv, and 6.36% on wikitext-103, relative to the base model on average), surpassing other memory-augmented methods.

8.3.2 Discussion

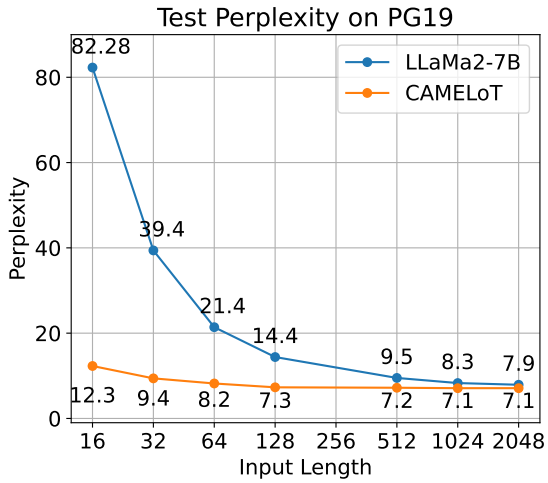


Figure 8.4. Test perplexity on PG19 with different input lengths.

Shorter Inputs, Better Performance Figure 8.4 shows CAMELoT's performance with different input lengths on PG-19 test set, with 10k memory slots. Unlike models without memory augmentation, CAMELoT demonstrates a relatively consistent performance across different input lengths. This stability can be attributed to the integration of additional knowledge in the AM saved from previous inputs. As CAMELoT accumulates past information, its visible context range extends beyond the current input, allowing an effective modeling of long-range dependencies irrespective of the length of the current input. In contrast, the model lacking memory augmentation relies solely on the local context of the current input, leading to performance fluctuations based on input length.

CAMELoT maintains its effectiveness even with tiny input lengths (e.g., 128), reducing

the demand on hardware resources such as large GPUs. This enables transformers to operate attention with shorter inputs but without compromising the quality of language modeling. Such an advantage lowers the barriers for deploying large language models in environments where computational budget is limited.

8.4 Conclusion

We have introduced CAMELoT, **C**onsolidated **A**ssociative **M**emory **E**nhanced **Lo**ng **T**ransformer, for long dependency modeling without the need for training. CAMELoT has a model-agnostic design, allowing seamless integration into different language models. Experimental results prove its effectiveness, with the long-context language modeling perplexity significantly reduced (by up to 29.7%), and superior performance is consistently obtained even with a tiny input window of 128 tokens. Future research directions connecting AM and LLMs involve improving the AM design (e.g., automatically learning a Write function) or tackling other long context modeling tasks (e.g., long document summarization or advanced reasoning).

Chapter 8, in part, is a reprint of the material as it appears in "CAMELoT: Towards Large Language Models with Training-Free Consolidated Associative Memory" by Zexue He, Leonid Karlinsky, Donghyun Kim, Julian McAuley, Dmitry Krotov, and Rogerio Feris, referenced as [He et al., 2024], in International Conference on Machine Learning Workshop on Long Context Foundation Models, 2024. The dissertation author was the primary investigator and author of this paper.

Table 8.1. Language modeling perplexity on wikitext-103, Arxiv, and Pg-19. For wikitext-103, we notice the maximum length of its documents is smaller than 2k. Therefore, we report the results of models whose effective input length ≤ 2048 (i.e., input length ≤ 2048 for non-augmented models; and input length ≤ 1024 for memory-augmented models). **Bold**: Best perplexity on each dataset. Avg: Average.

	PG-19 and Arxiv				wikitext-103		
	Input Length	Retrieved Mem.	PG-19	Arxiv	Input Length	Retrieved Mem.	Wikitext-103
LLaMa2-7B	512	None	9.54	5.99	512	None	16.0
	1024	None	8.33	4.98	1024	None	14.80
	2048	None	7.88	4.35	2048	None	14.46
	Avg	-	8.58	5.12	Avg	-	15.09
Transformer-XL	512	512	8.44	4.15	256	256	15.02
	1024	1024	8.27	3.81	512	512	14.21
	2048	2048	7.86	3.65	1024	1024	14.2
	Avg	-	8.19	3.87	Avg	-	14.48
Memorizing Transformers	512	512	8.12	3.82	256	256	14.18
	1024	1024	7.4	3.63	512	512	14.07
	2048	2048	7.34	3.62	1024	1024	14.39
	Avg	-	7.62	3.69	Avg	-	14.21
CAMELoT	512	512	7.24	3.61	256	256	14.06
	1024	1024	7.14	**3.60**	512	512	**14.00**
	2048	2048	**7.10**	**3.60**	1024	1024	14.34
	Avg	-	7.16	3.60	Avg	-	14.13

Part III

Social Good: Making NLP Systems Socially Positive

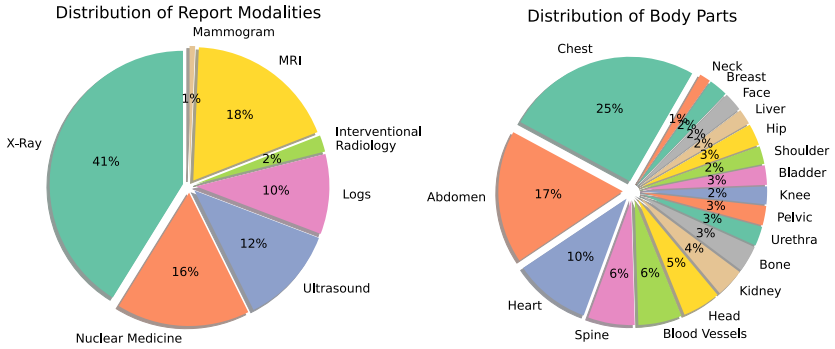
Chapter 9

LLMs For Healthcare: Evaluations

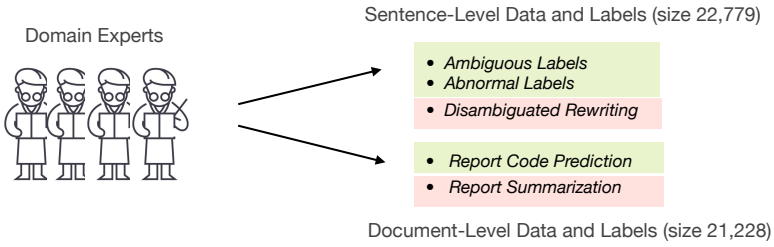
Recent advanced language models (LLMs), e.g., GPT-3, ChatGPT, and LLaMa [Touvron et al., 2023], are effective in various general tasks, suggesting their potential to healthcare use cases, such as alleviating the burden on human experts in decision-making and patient care. In this chapter, we present MEDEVAL , a multi-level, multi-task, and multi-domain medical benchmark to facilitate the development of language models for healthcare [He et al., 2023c]. MEDEVAL is comprehensive and consists of data from several healthcare systems and spans 35 human body regions from 8 examination modalities. With 22,779 collected sentences and 21,228 reports, we provide expert annotations at multiple levels, offering a granular potential usage of the data and supporting a wide range of tasks. Moreover, we systematically evaluated 10 generic and domain-specific language models under zero-shot and finetuning settings, from domain-adapted baselines in healthcare to general-purposed state-of-the-art large language models (e.g., ChatGPT).

9.1 Curated Benchmark: MEDEVAL

we introduce MEDEVAL, a large-scale medical benchmark with multi-level curated labels for multiple tasks and multiple domains. MEDEVAL comprises 22,779 sentence-level data points from radiology reports, including expert-crafted classification labels (e.g., abnormality identification labels) and ground truth for generation tasks (e.g., disambiguated rewritings).



(a) Distribution of modality and body parts in MEDEVAL



(b) Multi-task expert labels at multi-granularity

Figure 9.1. A summary of the multi-level multi-task and multi-domain medical benchmark (MEDEVAL). Classification tasks are highlighted in green and generation tasks are highlighted in red.

Additionally, we include 21,228 complete reports with expert-annotated medical codes for disease classification (e.g., for ankle radiology studies) and golden output for generation tasks (e.g., summarization of radiology reports). Besides the ability to support multi-tasks at different levels, MEDEVAL's uniqueness also lies in its diverse data coverage for different body parts (such as chest, foot, and ankle) and different modalities (X-rays, ultrasound, etc.), and the incorporated novel tasks/data that are collected from the U.S. Department of Veterans Affairs (VA) health care system nationwide. To the best of our knowledge, MEDEVAL represents the first expert-curated medical NLP benchmark that is both comprehensive and large-scale. MEDEVAL will be released to facilitate future research.

MEDEVAL (shown in Figure 9.1) is designed with multiple NLU and NLG tasks at both the sentence and document levels, based on medical data collected from two different healthcare databases. Our data covers diverse combinations of human body parts and examination modalities.

We first introduce the data sources where we collected the text input (Section 9.1.1). Then we present the expert-annotated ground truth labels[1] created by our medical team (Section 9.1.2).

9.1.1 Input Data Composition

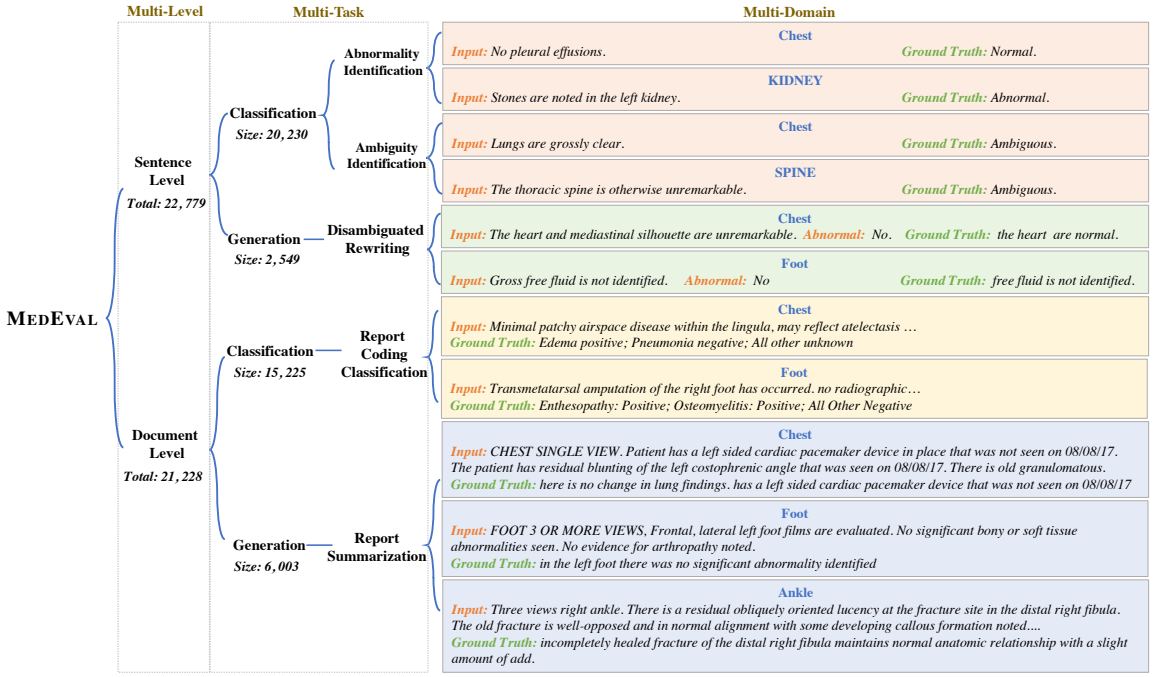


Figure 9.2. Dataset composition of MedEval. MedEval is a large-scale benchmark composed of 22,779 report sentences and 21,228 reports, covering multiple exam modalities on diverse body parts.

Sentence-Level Corpora The sentence-level corpora used in this study are sourced from two well-constructed datasets: the sentence-level OpenI-annotated dataset [Demner-Fushman et al., 2016], which consists of sentences from chest studies, and the VA-annotated dataset [He et al., 2023d], which includes sentences about different body parts examined by different modalities. These datasets have undergone de-identification, completion of missing terms and uniqueness checks. We use the officially released versions of the OpenI-annotated and VA-annotated datasets. In addition, we provide new annotations for sentence-level tasks on these data sources.

[1]We use "ground truth labels" to represent both discriminative labels for NLU and golden sentences for NLG tasks.

Table 9.1. Report disease codes covered in MEDEVAL.

Disease Codes of MEDEVAL			
Enlarged Cardiomediastinum	Cardiomegaly	Lung Opacity	Lung Lesion
Edema	Consolidation	Atelectasis	Pneumothorax
Pleural Effusion	Pleural Other	Support Devices	Pneumonia
Dislocation	Osteonecrosis	Fracture	Gout
Metatarsus Primus Varus	Gas	Swelling	Psoriasis
Enthesopathy	Hammer Toe	Osteomyelitis	Mass
Arthritis	Pes Planus	Rheumatoid	Cppd
Hardware	Erosion	Pes Cavus	Coalition
Subluxation	Fracture	Nodule	Rupture
Hallux Valgus	Pneumonia	Arthritis	No Finding

Report-Level corpora We collect the raw radiology reports from two distinct sources: (1) text corpus from MIMIC-CXR, which comprises records related to human chests [Johnson et al., 2019], (2) text corpus from the databases of a nationwide government healthcare system. We randomly collect data points about different body parts and exam modalities, resulting in multiple domains under different data distributions. The distribution of the domain is illustrated in Figure 9.1. The collected data are processed with automatic de-identification, followed by a thorough human inspection to verify that no private information about patients or doctors is disclosed or hinted at in the text. We also employ an offline paraphrasing tool [Damodaran, 2021] to revise the text data collected from the second source. The paraphrasing is followed by another human inspection to filter out any unqualified records where the rewriting deviates significantly from the original report. The resulting data set can be considered "synthesized" and containing no privacy information but retaining realistic clinical conditions as the source data.

For each evaluation task, we split the data in a ratio of 7:1:2 for train/validate/test.

9.1.2 Sentence-level Labels

NLU Tasks Identifying sentences with certain diagnostic properties is a practical use case in a real-world healthcare system. For example, identifying if a report sentence implies an abnormal finding about the patient or not. To test if language models can capture the medical semantics

of single sentences, we first include abnormal sentence identification into our evaluation pool. We use the sentence-level corpora and the associated abnormality labels to classify abnormal sentences.

Ambiguous sentences appear in radiology reports mainly due to the use of medical jargon whose meaning is different from daily usage, contradictory findings within the same sentence, or grammatical errors that mislead interpretation [He et al., 2023d]. Accurate identification of such sentences is crucial, as they impede patients' comprehension of diagnostic decisions, leading to potential treatment delays and irreparable consequences. To the best of our knowledge, as a novel task proposed recently, current LMs may not readily include such a task into its pre-training stage. Therefore, evaluation of this task allows us to investigate how language models perform when the tasks are unfamiliar. We leverage the report sentences and their associated ambiguous labels, and our medical team re-examined and re-annotated the labels for ambiguous sentences.

NLG Task Expanding beyond the previous ambiguous sentence identification, we include the task of sentence disambiguation as a sentence-level generation task. Proposed in He et al. [2023d], sentence disambiguation aims to rewrite an ambiguous sentence in a way that its diagnostic findings are more explicitly expressed while at the same time, the original content of the report sentence is faithfully maintained. This requires rewritten sentences to avoid the change of the original pathological findings or introducing new findings. Similar to ambiguous sentence identification, disambiguated rewriting presents a challenging generation task, not only because both the data and task formulation are not likely to be covered in the pre-training stage of existing language models, but also because there are two objectives that need to be optimized at the same time. In this task, based on the ambiguous sentences and their associated diagnostic labels, our medical team manually created the disambiguated rewritings as the ground truth.

9.1.3 Document-level Labels

NLU Task To access if language models can capture the key findings of a radiology report, we consider Report Codes Prediction as an evaluation task. This task involves categorizing reports into specific diagnostic codes based on the mentioned pathological findings. Therefore, different from sentence-level abnormality identification, this task requires a multi-label multi-class classification. Our medical team manually labels the medical codes of each report. Detailed information regarding the codes is provided in Table 9.1.

NLG Task Automatic medical summarization plays a crucial role in healthcare literature, by providing concise summaries, it saves time and manual effort for medical professionals when assessing the effectiveness of medical interventions. In our evaluation, we include report summarization as a task to assess the generation capability of language models. The *impression* section in each report serves as a summary that captures the supportive evidence for clinical decisions. To ensure data quality, we conduct a manual inspection of all collected <report, impression> pairs, filtering out any pairs where the impression does not align with the corresponding report. It is worth noting that the curated parallel data of reports and summaries provide valuable support for future work in related fields.

9.2 LLM Evaluation

9.2.1 Evaluated Language Models

We evaluate two categories of language models with MEDEVAL[2]: (1) domain-adapted pre-trained language models (Adapted PLMs), which are trainable models adapted on certain domain data, and (2) general-purpose large language models (Prompted LLMs) which are used by zero/few-shot prompting.

Domain-adapted PLMs Recent literature found it is effective to adapt pre-trained language

[2]The results presented are based on models evaluated as of the time of paper acceptance. We've added more results (e.g., on GPT4, LLaMa2, etc.)

Table 9.2. Evaluation (accuracy) over two categories of PLMs on abnormality identification and ambiguity identification tasks (sentence-level NLU). **Bold**: the highest performance. Underlined: the lowest.

Models		Chest		Miscellaneous Domains	
		Abnormality ↑	Ambiguity ↑	Abnormality↑	Ambiguity ↑
Adapted PLMs with Fine-Tuning	BERT	0.9791	**0.9893**	0.9607	0.9749
	RadBERT	0.9794	0.9869	0.9640	**0.9813**
	BioBERT	0.9791	0.9862	0.9614	0.9743
	ClinicalBERT	**0.9809**	0.9874	0.9588	0.9736
	BlueBERT	0.9803	0.9867	0.9601	0.9775
	BioMed-ReBERTa	0.9569	0.9758	**0.9776**	0.9788
LLMs Prompted by Zero/Few Shot	zero-shot ChatGPT	0.9277	0.6584	0.8880	0.5206
	few-shot ChatGPT	0.9498	0.5831	0.9099	0.5354
	zero-shot GPT-3	0.8762	0.8742	0.8243	0.6448
	few-shot GPT-3	0.9215	0.8320	0.9054	0.6371
	zero-shot Vicuna-7B	0.6987	0.2130	0.7261	0.3739
	few-shot Vicuna-7B	0.8071	<u>0.0785</u>	0.8166	<u>0.2844</u>
	zero-shot BioMed LM	<u>0.6679</u>	0.3485	<u>0.6273</u>	0.3726
	few-shot BioMedLM	0.7905	0.6804	0.7638	0.6804

models to certain narrow domains such as biomedical text by a continued training step on domain-specific data [Gururangan et al., 2020a], following which we take a pre-trained (or generally adapted) language model, and test it on the MEDEVAL test set. We also fine-tuned the models from this category to customize it to fit the tasks of MEDEVAL, with their corresponding training data. For NLU tasks at both levels, we follow the evaluation setting of Yan et al. [2022] and investigate how: **BERTbase** [Devlin et al., 2019], **RadBERT** [Yan et al., 2022], **BioBERT** [Lee et al., 2020], **clinicalBERT** [Huang et al., 2019], **BlueBERT** [Peng et al., 2019], and **BioMed-ReBERTa-base** [Gururangan et al., 2020b] perform on MEDEVAL.

For the sentence-level NLG task, we follow the the setting of He et al. [2023d] by evaluating: (1) **style transformer** [Dai et al., 2019b] which transfers the original sentence into a less ambiguous style, (2) **PPLM** [Dathathri et al., 2020] which adds perturbation to LM to move the (re-)generation towards a less ambiguous direction, (3) **DEPEN** [He et al., 2021b] which is built upon PPLM and only re-generates ambiguous tokens detected before, and (4) **MedDEPEN** [He et al., 2023d], a biomedical-adapted DEPEN by introducing contrastive pre-training. Each

work has included a transformer-based language model. We refer the reader to the original papers for more details.

For the document-level NLG task, we follow the setting of Yan et al. [2022] and customize previously adapted BERT-based models used before for the summarization task.

Prompted LLMs We include the following general-purpose large language models to test their generalization in the healthcare domain: (1) **GPT3**: GPT-style large language models with 175B parameters [Brown et al., 2020]. We use `davinci-003`. (2) **ChatGPT**[3]: GPT-style large language model trained with Reinforcement Learning from Human Feedback (RLHF). We use `GPT3.5-turbo`. (3) **Vicuna-7B** [Chiang et al., 2023]: The finetuned version of LLaMa-7B [Touvron et al., 2023] with 70K user-shared ChatGPT conversations, which is capable of generating more detailed and well-structured answers. (4) **BioMedLM**[4]: a 2.7B GPT-style language model trained exclusively on biomedical abstracts and papers from The Pile[Gao et al., 2020a].

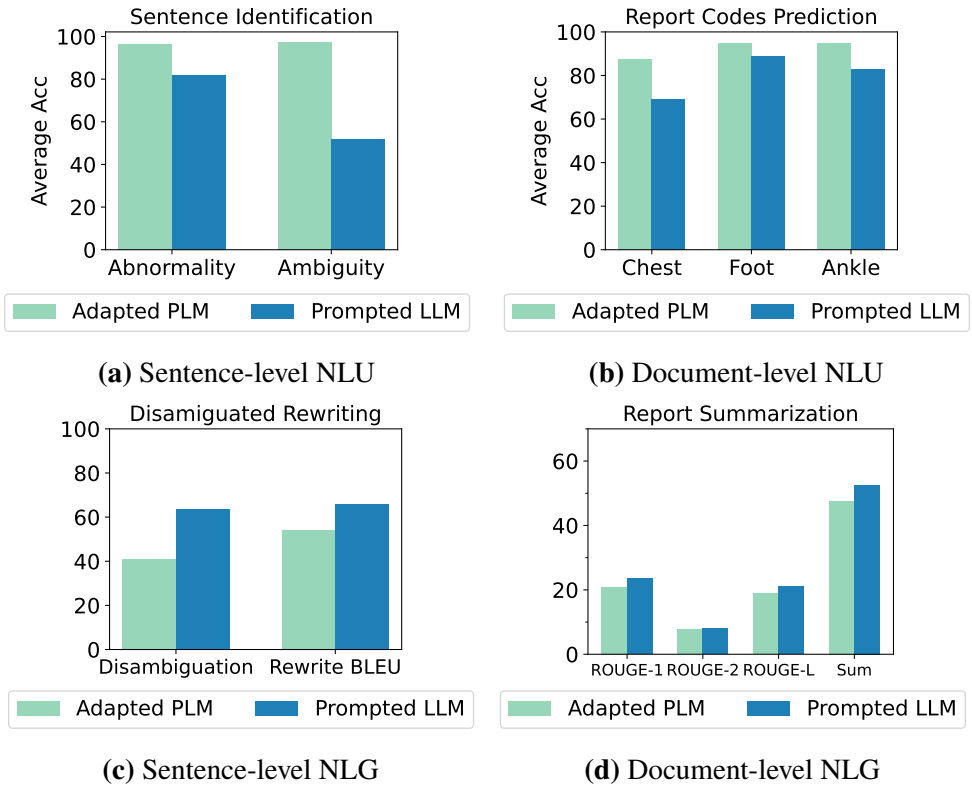
We prompt those LLMs under zero/few-shot settings, where we randomly select the examples from the training set of each task to compose prompts 5 times. We report the test results with the prompts which obtain optimal results on the validation set.

9.2.2 Evaluation Metrics

For NLU tasks, we report classification metrics including accuracy and F1 scores. For NLG tasks, we report BLEU and ROUGE scores with respect to the ground truths labeled by our medical team. For sentence-level generation tasks (i.e., rewriting), to evaluate the objective of disambiguation, we follow the setting of He et al. [2023d] to report accuracy decrements of the ambiguity classifier (ΔAcc_{am}) as the disambiguation metric. To evaluate the rewriting fidelity, we report the content distortion score, which is defined as the decrement of the accuracy from an abnormality classifier (ΔAcc_{ab}). Therefore, higher distortion indicates a lower content fidelity.

[3]https://openai.com/blog/chatgpt
[4]https://crfm.stanford.edu/2022/12/15/biomedlm.html

9.3 Results and Discussion



(a) Sentence-level NLU

(b) Document-level NLU

(c) Sentence-level NLG

(d) Document-level NLG

Figure 9.3. Average performance of adapted PLM and prompted LLM on different tasks and at different levels.

In this section, We first present the results for sentence-level NLU tasks (Ambiguity Identification and Abnormality Identification) in Table 9.2, then sentence-level NLG task (Disambiguated Rewriting) in Table 9.3, finally document-level NLU (Code Prediction) and NLG (Report Summarization) tasks in Table 9.4 and Table 9.5.

The Effectiveness of Instruction Tuning

While BioMed LM is the first large language model customized for the biomedical domain, we observe that it does not outperform adapted PLMs and most prompted LLMs in the majority of tasks. Particularly, BioMed LM has been found to be the weakest performer in tasks such as sentence identification, disambiguated rewriting, and report summarization. We would like to highlight that, unlike other prompted LLMs such as ChatGPT, GPT-3, and Vicuna,

BioMed LM lacks an Instruction Tuning step in its model training. This omission significantly impacts BioMed LM's ability to generate replies following the instructions from the given options. In zero-shot NLU tasks, only 40% of the test cases receive appropriate responses at the sentence level and the qualified rate drops to less than 1% at the document level (so we did not report the results in Table 9.4). In few-shot report codes prediction, the document-based prompts often exceed BioMed LM's maximum threshold of 1024 tokens, resulting in query errors. In generation tasks, BioMed LM keeps returning irrelevant text. Our manual inspection reveals that the outputs rarely adhere to the given instructions in prompts or address the queries. This is further supported by the remarkably low BLEU or ROUGE scores in Table 9.3 and Table 9.5. These findings underscore the significance of Instruction Tuning and establish it as a crucial step when adapting prompted LLMs for specialized applications like healthcare decision-making.

In the remainder of this section, we focus on addressing more intriguing questions based on average performance across a range of baselines (e.g., the average accuracy of adapted PLMs versus prompted LLMs), where we exclude BioMed LM from further consideration.

Discussion on Task Type and Granularity

In this section, we aim to determine the proficiency of language models at different levels and tasks. To achieve this, we begin by calculating the average accuracy scores of all adapted PLM baselines and prompted LLM baselines in sentence identification tasks. Similarly, we compute the average accuracy of adapted PLM and prompted LLM baselines in a document-level code classification task.

First, examining the results presented in Figure 9.3, we observe that both adapted PLMs and prompted LLMs perform relatively similarly across different data levels. However, it becomes apparent that adapted PLMs outperform prompted LLMs in NLU tasks, no matter whether it's on the sentence or document level. This suggests that fine-tuning provides a more effective means of injecting specific knowledge about narrow domains or tasks. On the other hand, consistently superior performance of prompted LLMs compared to adapted PLMs is observed in generation

tasks, at both the sentence and document levels. This can be attributed to multiple advantages of large-scale pre-training such as a larger model size or the benefits HFRL in the LLMs we utilized, such as ChatGPT. These models demonstrate a capability to generate language that is more akin to human-like expressions, thereby achieving better generation scores. These imply that fine-tuning PLM models can be a viable choice for NLU tasks, while prompting-based LLMs may be more suitable when healthcare professionals require an AI writer to help their work.

Common v.s. Rare Domains

In Table 9.6, we explore the impact of the domain on language models in the healthcare field. We compute the average accuracy of adapted PLMs and prompted LLMs in abnormality identification v.s. ambiguity identification. We consistently observe higher performance from both adapted PLMs and prompted LLMs when working with data from the chest domain compared to miscellaneous domains. This superior performance can be attributed to the similarity between the chest data we tested and the pre-training data of the language models – chest-related healthcare text is widely available in the public domain and can be included in the training corpus of PLMs. Similarly, LMs are expected to excel in abnormality identification tasks, which are a common research topic in current literature.

The most challenging scenario arises when both the data and task are unseen, specifically in the case of ambiguous identification within the miscellaneous domain. In such situations, there are limited or no examples available in the public domain. Therefore, querying language models with (zero) few-shot learning proves to be less effective.

Family of LLMs and Few Shot Learning

In this analysis, we examine the behavior of different language models (LLMs) with varying numbers of shots across different tasks. We calculate the average accuracy of ChatGPT, GPT3, and Vicuna-7B in NLU tasks and the average BLEU scores in NLG tasks. Additionally, we consider the average performance achieved in zero-shot or few-shot settings (Table 9.7). From the table, it is evident that in most cases, providing additional examples assists LMs in making

predictions for NLU tasks. However, in NLG tasks, no consistent trend is observed, indicating the need for further research to discover optimal prompts. We do not observe a clear advantage of any specific LLM family over others, suggesting that the choice of the optimal LLM family for a given task may vary on a case-by-case basis.

9.4 Conclusion

We introduce MEDEVAL, a multi-task, multi-level, and multi-domain medical benchmark designed to serve as a comprehensive testbed for advanced language models. Through extensive evaluation experiments, we thoroughly analyze the capabilities and limitations of current LLMs in tackling various medical tasks, such as the effectiveness of instruction tuning and the performance disparities between adapted and prompted LMs in NLU and NLG tasks. Our findings provide valuable insights and serve as a handbook for future research in utilizing LLMs to enhance healthcare practices.

Chapter 9, in part, is a reprint of the material as it appears in "MedEval: A Multi-Level, Multi-Task, and Multi-Domain Medical Benchmark for Language Model Evaluation." by Zexue He, Yu Wang, An Yan, Yao Liu, Eric Chang, Amilcare Gentili, Julian McAuley, and Chun-nan Hsu, referenced as [He et al., 2023c], in Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 8725-8744. 2023. The dissertation author was the primary investigator and author of this paper.

Table 9.3. Evaluation on disambiguated rewriting Tasks (sentence-level NLG). We report the disambiguation score, content distortion score (where smaller content distortion indicates higher fidelity), and BLEU4 score. **Bold**: the best performance. Underlined: the worst.

Model		Chest			Miscellaneous Domains		
		Disambiguation $\Delta Acc_{am} \uparrow$	Content Distortion $\Delta Acc_{ab} \downarrow$	BLEU4 ↑	Disambiguation $\Delta Acc_{am} \uparrow$	Content Distortion $\Delta Acc_{ab} \downarrow$	BLEU4 ↑
Adapted PLMs with Fine-Tuning	Style Transfer	0.5010	0.0510	27.92	0.3110	0.2350	31.17
	PPLM	0.3860	0.1150	57.88	0.2700	0.1460	60.14
	DEPEN	0.5000	0.0520	60.48	0.3530	0.0470	67.86
	MedDEPEN	0.4960	0.0320	57.88	0.4810	**0.0090**	68.88
LLMs Prompted by Zero/Few Shots	zero-shot ChatGPT	0.6337	**-0.0297**	60.73	0.6539	0.1483	60.64
	few-shot ChatGPT	0.5875	0.0000	68.92	0.6370	0.0815	67.98
	zero-shot GPT-3	**0.6799**	-0.0132	61.78	**0.8022**	0.1528	61.05
	few-shot GPT-3	0.6139	0.0000	**76.33**	0.7146	0.0607	**77.09**
	zero-shot Vicuna-7B	0.6230	0.0693	66.65	0.6771	0.3653	64.64
	few-shot Vicuna-7B	0.5311	0.1914	62.55	0.4811	0.2739	63.72
	zero-shot BioMed LM	0.2211	0.0066	23.40	0.1528	0.1416	24.11
	few-shot BioMed LM	0.1386	-0.0262	23.30	0.3933	0.3640	23.48

Table 9.4. Evaluation on report codes prediction Task (Document-level NLU). We report the average accuracy over all classes of diseases and the exact match rate (EMR) between predictions and labels. **Bold**: the highest performance. Underlined: the lowest.

Model		Chest		Foot		Ankle	
		avg Accuracy ↑	avg EMR ↑	avg Accuracy	avg EMR ↑	avg Accuracy ↑	avg EMR ↑
Adapted PLMs with Fine-Tuning	BERT	0.8779	0.2263	**0.9754**	0.5635	0.9787	0.6141
	RadBERT	0.8785	0.1941	0.9710	0.4910	0.9773	0.5710
	BioBERT	0.8782	**0.2400**	0.9750	0.5617	**0.9801**	**0.6266**
	ClinicalBERT	0.8780	0.2341	0.9731	0.5372	0.9798	0.6224
	BlueBERT	**0.8843**	0.2380	0.9703	**0.5939**	0.9761	0.5752
	BioMed-ReBERTa	0.8579	0.1415	0.9692	0.4631	0.9752	0.5522
LLMs Prompted by Zero/Few Shots	zero-shot ChatGPT	0.5272	0.1024	0.9621	0.4449	0.9660	0.4491
	few-shot ChatGPT	0.6485	0.1951	0.9621	0.4186	0.9690	0.4875
	zero-shot GPT-3	0.2744	0.1424	0.9621	0.4449	0.1887	0.6273
	few-shot GPT-3	0.8160	0.1805	0.9617	0.4186	0.9691	0.4908
	zero-shot Vicuna-7B	0.8216	0.0672	0.9617	0.4186	0.9691	0.4908
	few-shot Vicuna-7B	0.8228	0.0782	0.5156	0.1041	0.9122	0.4153
	few-shot BioMed LM	0.8320	0.0689	0.9667	0.4664	0.9719	0.4980

Table 9.5. Evaluation on report summarization task (Document-level NLG). We report the Rouge scores and BLEU4 scores. **Bold**: the highest performance. Underlined: the lowest.

Model		Miscellaneous Domains				
		ROUGE-1 ↑	ROUGE-2 ↑	ROUGE-L↑	Sum ↑	BLEU4↑
Adapted PLMs with Fine-Tuning	BERT	20.48	7.46	18.57	46.52	30.28
	RadBERT	20.96	7.63	18.90	47.50	30.77
	BioBERT	20.79	7.62	18.78	47.19	30.49
	ClinicalBERT	21.22	7.85	19.18	48.26	30.81
	BlueBERT	20.83	7.78	18.90	47.51	30.93
	BioMed-ReBERTa	21.19	7.85	19.14	48.18	30.88
LLMs Prompted by Zero/Few Shots	zero-shot ChatGPT	24.64	7.97	22.05	54.66	32.30
	few-shot ChatGPT	**24.96**	8.43	**22.23**	55.62	**35.43**
	zero-shot GPT-3	24.06	8.67	21.52	54.24	24.43
	few-shot GPT-3	24.73	**9.14**	22.16	**56.03**	34.72
	zero-shot Vicuna-7B	20.93	6.96	18.71	46.60	20.94
	few-shot Vicuna-7B	21.42	7.26	19.22	47.90	22.00
	zero-shot BioMed LM	17.70	5.11	16.49	39.30	15.43
	few-shot BioMed LM	12.15	3.50	11.22	26.87	20.27

Table 9.6. Average accuracy of adapted PLMs and prompted LLMs in NLU over different domains.

Model	Abnormality ↑		Ambiguity ↑	
	Chest	Miscellaneous	Chest	Miscellaneous
Adapted PLM	0.9758	0.9526	0.9836	0.9621
Prompted LLM	0.8635	0.8451	0.5399	0.4893

Table 9.7. Average accuracy and BLEU of various LM families with zero/few shots.

Model Family	# shot	NLU (Accuracy↑)		NLG (BLEU↑)	
		Individual	Average	Individual	Average
ChatGPT	0-shot	0.78	0.79	51.22	47.15
	Few-shot	0.79		43.08	
GPT-3	0-shot	0.66	0.76	49.08	55.90
	Few-shot	0.86		62.71	
Vicuna-7B	0-shot	0.71	0.73	50.74	50.08
	Few-shot	0.75		49.42	
Average	0-shot	0.72		50.35	
	Few-shot	0.80		51.74	

Chapter 10

LLMs for Medical Report Generation

Sharing medical reports is a critical aspect of patient-centered care, facilitating effective communication between healthcare providers and patients. Recently, AI models have been proposed to automatically generate and share medical reports, alleviating the workload of doctors and improving efficiency. However, different audiences have different purposes when writing/reading medical reports – for example, healthcare professionals care more about pathology, whereas patients are more concerned with the diagnosis ("*Is there any abnormality?*"). The expectation gap results in a common situation where patients find their medical reports to be ambiguous and therefore unsure about the next steps.

In this chapter, we mitigate the *audience expectation gap* in healthcare by designing an automatic report rewriting framework [He et al., 2023d]. We first summarize common ambiguities that lead patients to be confused about their diagnosis into three categories: *medical jargon*, *contradictory findings*, and *misleading grammatical errors*. Based on our analysis, we define a disambiguation rewriting task to regenerate an input to be unambiguous while preserving information about the original content. We further propose a rewriting algorithm based on contrastive pretraining and perturbation-based rewriting. In addition, we create two datasets, OpenI-Annotated based on chest reports and VA-Annotated based on general medical reports, with available binary labels for ambiguity and abnormality presence annotated by radiology specialists. Experimental results on these datasets show that our proposed framework effectively

rewrites input sentences less ambiguously with high content fidelity.

10.1 Expectation Gap Between Audience in Healthcare

Effective communication between healthcare providers and patients plays a critical role in patient outcomes. *Patient-centered care* [Catalyst, 2017, Stewart et al., 2013] is reforming traditional healthcare to shift a patient's role from an "order taker" to an active "team member" in their own healthcare process, to improve individual health outcomes and satisfaction [Stewart et al., 2000], and advocates sharing medical information fully and in a timely manner with patients. It is also required by legal obligation (e.g., HIPAA[1] in the US) that patients have a legal right to access their personal health information. Failure of healthcare providers to communicate with patients efficiently and effectively about the results of medical examinations may lead to delays in proper treatment or malpractice lawsuits against providers [Mityul et al., 2018, Srinivasa Babu and Brooks, 2015].

As a carrier of medical information, medical reports are shared with their patients by healthcare providers nowadays. Medical reports serve many communication purposes with different audiences including ordering physicians, other care team staff members, patients and their families, and researchers [Hartung et al., 2020, Gunn et al., 2013]. Each group has different needs and expectations when reading the reports: peer medical professionals pay more attention to actionable findings, while patients usually care more about the diagnostic outcome[2] (i.e., *Is there anything abnormal?*). How to address various communication needs for different audiences, and to bridge the *expectation gap between audiences* without increasing the workload of report writers is critical.

[1] Shorten for Health Insurance Portability and Accountability Act

[2] We use exam result, diagnostic decision, abnormality existence interchangeably, to express "if there is anything abnormal".

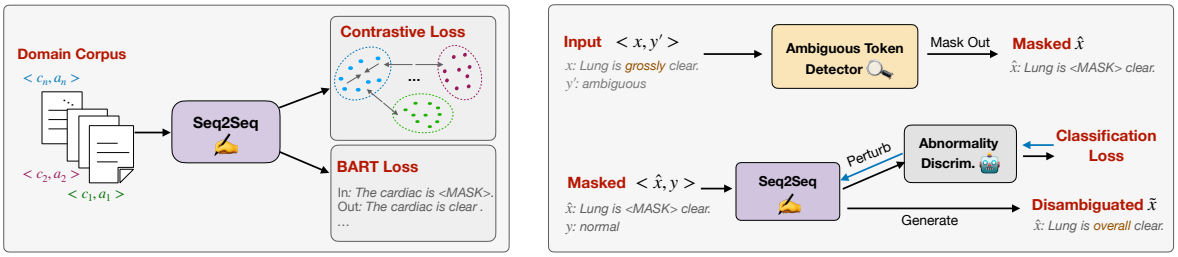
Table 10.1. Ambiguous sentences (**Am**) from three categories with the unambiguous rewritten (**Re**). We highlight the parts causing ambiguity in red, and show comparisons in yellow.

	Report Sentence	Diagnosis
Medical Jargon	**Am**: *Unremarkable bony structure.* **Re**: *Normal bony structure.*	Normal
Contradictory Findings	**Am**: *The lung volumes are low normal .* **Re**: *The lung volumes are in the lower half of the normal limit .*	Normal
Misleading Grammatical Errors	**Am**: *Cardiomegaly and hiatal hernia without an acute abnormality identified.* **Re**: *Cardiomegaly and hiatal hernia . Without an acute ab-normality identified.*	Abnormal

10.1.1 Ambiguity in Medical Reports

To build such a bridge, it is important for medical reports to 1) be understandable with little specialized terms and 2) to have no ambiguity about the significance of findings when communicating with patients [Hartung et al., 2020, Mityul et al., 2018]. Previous works mainly focus on the first point where they change terminology to lay-person terms with replacement-based or deep learning methods [Qenam et al., 2017, Oh et al., 2016, Xu et al., 2022b]. However, how to mitigate the ambiguity in a comprehensible report is crucial but rarely investigated.

Therefore, we consider medical reports written in free text and analyze the ambiguity where patients are unsure about their exam results. We first collect medical report data and ask domain specialists to label the binary abnormality presence associated with each sentence, and non-experts to label sentences that they deem ambiguous. Our medical team analyze the results and categorize the major causes behind ambiguity primarily into three categories: the report sentence is ambiguous due to containing (1) **medical jargon** with meanings different from everyday general usage, such as *unremarkable*; (2) **contradictory findings** in the same sentence; (3) **misleading grammatical errors** such as no period between full sentences. Examples are shown in Table 10.1.



(a) Contrastive Pretraining **(b)** Rewriting Framework

Figure 10.1. Medical report disambiguation: model illustration. Our model contains two steps: first do (a) constrastive pretraining and then (b) rewriting.

10.2 Disambiguating Medical Reports

Our task is to disambiguate an input medical sentence when a patient finds it hard to understand the diagnostic decision. For an ambiguous sentence x whose abnormality label is y (abnormality presents or not), we will output a disambiguated sentence $\tilde{x}$ that is more explicit about y.

We propose a contrastive knowledge infused rewriting framework to achieve this goal, which comprises a pretraining step and a rewriting step, as shown in Figure 10.1 (a) and (b). We first obtain a medical-domain Seq2Seq model $\mathcal{G}$ that can effectively capture language patterns in different health situations in the pretraining step, and we generate a less ambiguous sentence using $\mathcal{G}$ in the rewriting step. We introduce each step in following sections.

10.2.1 Contrastive Pretraining

First, we pretrain a domain-specific Seq2Seq model to generate medical language on top of a general domain BART [Lewis et al., 2020]. For our task, a pretrained model that only captures the distribution of medical language is not precise enough – there are several ways to rewrite an abnormal diagnostic in order to make it "more abnormal". Consider a patient with a diagnosis of having excessive lung fluid. This abnormal diagnosis can be rewritten to be "more abnormal" by combining it with other abnormalities (such as *unusual liquid and unusual air*) or by changing the disease (from a lung disease to a heart disease). This is undesirable.

Therefore, we require the rewritting to preserve the original diagnosis. To achieve this goal, more domain knowledge about different pathologies is required. We capture such domain knowledge by learning from external corpora (such as MIMIC-CXR[Johnson et al., 2019]) that are on a large scale in the same medical domain with fine-grained disease labels. We pretrain the language model by infusing the domain knowledge with supervised contrastive learning, which pushes sentences closer if they express similar pathological findings and pull away sentences if they are not. As a result, we can reduce the probability of rewriting a sentence medically different from the original input. This is crucial for patient safety and a unique issue in our task different from other text rewriting problems.

The external medical corpora consist of medical report pairs including both sentence c_i and its associated fine-grained pathological label a_i (such as disease labels like *atelectasis, edema, no finding, etc*). We pretrain an encoder-decoder transformer $\mathcal{G}$ with supervised contrastive learning, following Khosla et al. [2020]. For each sentence c_i in a mini-batch B, we first obtain its representation H_{c_i} by taking the last hidden states from the decoder in $\mathcal{G}$. Then a τ-temperature similarity $s_{i,j}$ between c_i and another sentence c_j in B is calculated:

$$s_{i,j} = \text{sim}(H_{c_i}, H_{c_j}) = H_{c_i} \cdot H_{c_j}/\tau \tag{10.1}$$

We use $S(i) = \{c_j : a_j = a_i\}$ to denote the set of sentences sharing the same disease label a_i, then the contrastive learning loss $\mathcal{L}_{\text{CL}}$ for the mini-batch B is defined as

$$\mathcal{L}_{\text{CL}} = \sum_{c_i \in B} \mathcal{L}_{\text{CL},i} \tag{10.2}$$

where

$$\mathcal{L}_{\text{CL},i} = -\frac{1}{|S(i)|} \log \frac{\sum\limits_{c_j \in S(i)} \exp(s_{i,j})}{\sum\limits_{c_j \in B} \exp(s_{i,j})} \tag{10.3}$$

Our constrative pretraining is also applicable even when the pathological label a_i is not available. A recent empirical study [Oakden-Rayner et al., 2020] shows that the representations from deep neural networks carry information of labels and unlabeled features. Inspired by this, in the case where a_i is not available, we first extract sentence representations from a medical Bert pretrained with a radiology report corpus [Yan et al., 2022]. Then we follow Sohoni et al. [2020] to cluster sentences with a Gaussian Mixture Model. The clustered results carry fine-grained information about different pathological patterns, and work as an approximation of the labels used in optimizing Equation (10.3).

Besides the contrastive learning objective, our pretraining also includes a token infilling task [Lewis et al., 2020] in order to obtain an informative representation H_{c_i}, which reconstructs the original sentence c_i from its randomly masked version $\hat{c}_i$:

$$\mathcal{L}_{\text{BART}} = -\sum_{i}^{|B|}\sum_{t}^{|c_i|}\log p(c_i^t|c_i^1,\cdots,c_i^{t-1};\hat{c}_i) \tag{10.4}$$

Therefore, our pretraining goal is to learn the medical language distribution (language modeling loss $\mathcal{L}_{\text{BART}}$) and capture language patterns of different medical conditions (contrastive loss $\mathcal{L}_{\text{CL}}$), and is formulated by minimizing their weighted sum in Equation (10.5):

$$\mathcal{L} = \lambda_1\mathcal{L}_{\text{BART}} + \lambda_2\mathcal{L}_{\text{CL}} \tag{10.5}$$

10.2.2 Rewriting Framework

During the rewriting process for an ambiguous input x_i, the following objectives are targeted: 1) the main content is retained, and 2) the diagnostic decision is more explicitly expressed in the rewritten sentence. While contrastive pretraining ensures a reasonable level of content fidelity, the first objective also suggests minimal changes during rewriting, which only touch those portions necessary for disambiguation. The second one requires a controllable generation that pushes the generation closer to the diagnostic decision.

Inspired by recent advance in controlled text generation [He et al., 2021c], we leverage a plug-and-play method to rewrite the sentences without the need of parallel annotated training data. It includes a *detect* stage to mask potential tokens that are highly predictable for an attribute and a *perturb* stage to do neutralization rewriting w.r.t. that attribute. Since we need to detect tokens in x_i that are highly predictable in their ambiguity, we first train an ambiguity classifier during detect stage. The tokens with the top-K highest attention scores will be detected as salient for ambiguity and will be masked. Then in the perturb stage, we require an edit that is more explicit in the direction of its diagnostic decision y_i, rather than making it more neutral for the ambiguity. Therefore, we modify the perturb stage to suggest an explicit edit by maximizing the likelihood of making the right diagnostic decision at each generation step t:

$$\tilde{x}_i^t = \arg\max_{\tilde{x}_i^t} p(y|\tilde{x}_i^t), \tag{10.6}$$

where the distribution p is output from a classifier f which predicts the diagnostic decision y_i, pretrained by minimizing Cross-Entropy$(f(x_i), y_i)$.

Then, during generation, we add a perturbation to decoder hidden states in $\mathcal{G}$ by taking the gradient w.r.t. $\tilde{x}_i^t$ from the Cross-Entropy loss, and regenerate the token distribution since the hidden states have been updated. Alternatively, adding perturbation and (re-)generation push the rewritten sentence towards the direction of its diagnostic decision, that is to say, being less ambiguous.

10.3 Experimental Setup

Our rewriting algorithm is tested in two practical settings. First, disambiguating chest reports in a specialized medical domain. Secondly, disambiguating general medical reports that cover many imaging modalities (e.g., x-ray, CT, etc.) and body parts. Our medical team created annotation datasets (OpenI-Annotated and VA-Annotated) for each experiment. During pretraining, an additional large-scale medical corpus is used in each experiment.

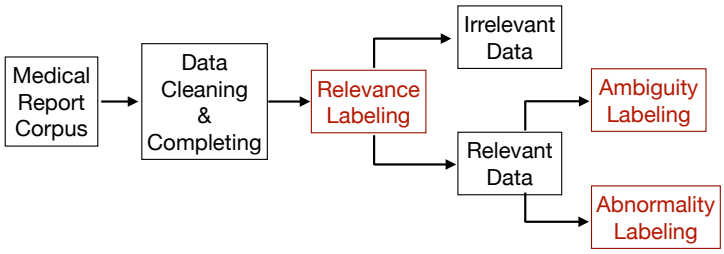


Figure 10.2. Medical report disambiguation: data annotation pipeline. Three different labels are annotated in red steps.

Table 10.2. Medical report disambiguation: statistics of annotated datasets.

Dataset	Total	Ambiguous	Abnormal
OpenI Annotated	15,023	988	6,111
VA Annotated	5,180	1,461	2,358

10.3.1 Human-Annotated Datasets for Rewriting

The overall pipeline for building our annotated dataset is shown in Figure 10.2. We elaborate each dataset as follows. See more in Appendix Section "Dataset Details".

OpenI-Annotated

We take the sentence-level subset of OpenI released by Harzig et al. [2019]. Our medical team conducts data cleaning by removing identical sentences and completing missing terms (mistakenly masked in de-identification) according to their domain knowledge. We distinguish sentences that are irrelevant to our task and re-label sentences that contain abnormal findings and that are ambiguous according to corresponding criteria. In the end, the OpenI-Annotated

Table 10.3. Fine-grained diseases in MIMIC-CXR.

Disease	Num.	Disease	Num.
Enlarged Cardiomediastinum	17,944	Atelectasis	32,445
Cardiomegaly	56,099	Pneumothorax	5,539
Lung Opacity	62,865	Pleural Effusion	36,537
Lung Lesion	9,838	Pleural Other	3,350
Edema	14,605	Fracture	10,893
Consolidation	3,905	Support Devices	8,355
Pneumonia	1,365	No Finding	865,738

Table 10.4. Automatic Evaluation Results on OpenI- and VA-Annotated. Statistics about the original data is provided separately.

		Ambiguity Acc.	Decision Acc.	Pathology Match	PLL
OpenI-Annotated	**Raw Text**	0.855	0.950	1.000	-6.062
		Disambigutation	**Content Fidelity**		**Language Fluency**
		$\Delta\text{Acc}_{\text{Am}} \uparrow$	$\Delta\text{Acc}_{\text{Dis}} \downarrow$	Pathology Match ↑	PLL ↑
	KBR	0.343	0.001	0.782	-6.862
	ST	0.501	0.051	0.629	-6.454
	PPLM	0.386	0.115	0.643	-6.890
	DEPEN	0.500	0.052	0.676	-6.529
	Ours	0.496	0.032	0.809	-6.232
		Ambiguity Acc.	**Decision Acc.**	**Pathology Match**	**PLL**
VA-Annotated	**Raw Text**	0.955	0.946	1.000	-5.652
		Disambigutation	**Content Fidelity**		**Language Fluency**
		$\Delta\text{Acc}_{\text{Am}} \uparrow$	$\Delta\text{Acc}_{\text{Dis}} \downarrow$	Pathology Match ↑	PLL ↑
	KBR	0.495	0.007	0.885	6.109
	ST	0.311	0.235	0.351	-7.284
	PPLM	0.270	0.146	0.582	-6.147
	DEPEN	0.353	0.047	0.838	-6.102
	Ours	0.481	0.009	0.856	-5.821

dataset consists of sentences with associated binary labels for being irrelevant, ambiguous, or abnormal. Statistics are shown in Table 10.2. In the experiment, we split the OpenI-Annotated data to train/validation/test sets by 70%, 10%, 20%.

VA Annotated We create the VA-Annotated dataset and use it in general-domain medical report rewriting. We use the VA radiology report corpus, recently introduced in [Yan et al., 2022]. As a general medical report corpus, it covers 8 modalities and 35 body parts for 70 modality-body part combinations. We sample a subset and split it into sentences. Then similar data cleaning steps for OpenI-Annotated are used. Each sentence is annotated with binary labels for relevance, ambiguity, and abnormality. We call it VA-Annotated and its statistics are listed in Table 10.2. In the experiment, we split it into train/validation/test sets by 70%, 10%, 20%.

Human Labeling Procedures In each experiment, experts start the labeling procedures after data cleaning.

First, for relevance labeling, the sentences only containing facts (e.g., *CT of the chest*

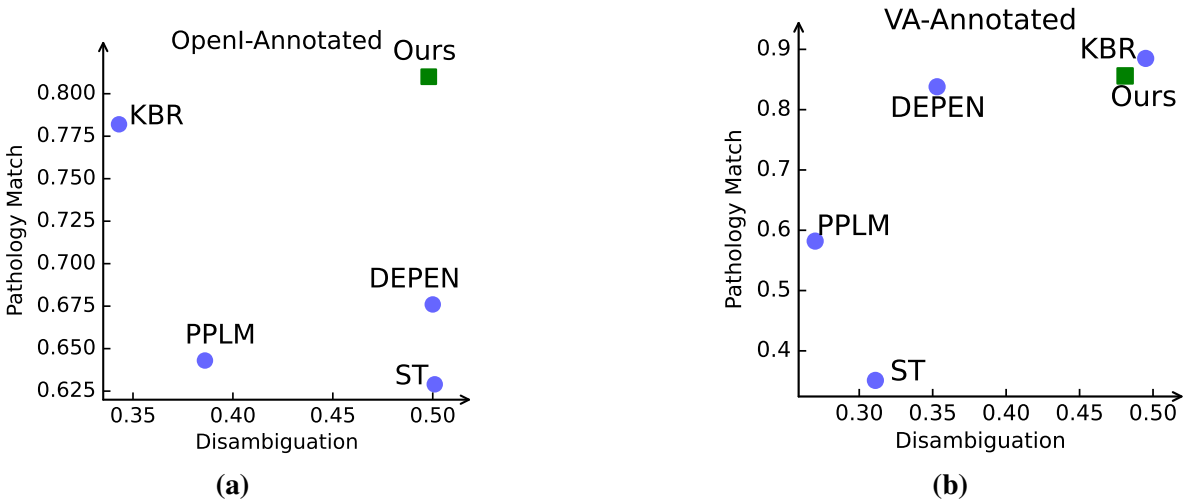


Figure 10.3. Trade-off between Disambiguation and Fidelity on (a) OpenI-Annotated (b) VA-Annotated. Higher disambiguation and pathology match (more upper-right corner in geometric) indicates a better rewriting.

are taken) and body parts (*left knee was not evaluated*) are regarded as irrelevant to our task since there are no abnormal/normal diagnoses mentioned. Our medical team provides their binary labels for relevance. Secondly, for relevant sentences, the medical team annotates a binary *abnormality label*, indicating if there is an abnormal symptom found in the sentence. Sentences containing abnormal symptoms usually imply a diagnostic decision of being sick. Our non-expert team annotates an *ambiguity label*, indicating whether the diagnostic decision looks too ambiguous for patients to understand.

The annotations are performed iteratively until inter-annotator Cohen's Kappa is higher than substantial agreement (≥ 0.8). The final discrepant labels were resolved by doctors in our medical team. See more details about the labeling criteria in Appendix Section "Human Labeling Details".

10.3.2 Contrastive pretraining Datasets

Here we introduced the corpus used in contrastive pretraining and elaborated their fine-grained pathological labels about different health conditions.

MIMIC-CXR MIMIC-CXR is the largest public-domain chest x-ray dataset proposed

Table 10.5. Human evaluations on disambiguated reports. *Disam*: Disambiguation.

	OpenI-Annotated		VA-Annotated	
Models	Disam ↑	Fidelity ↑	Disam ↑	Fidelity ↑
KBR	0.317	0.908	**0.488**	**0.988**
ST	0.609	0.526	0.225	0.350
PPLM	0.376	0.624	0.333	0.575
DEPEN	0.571	0.795	0.282	0.941
Ours	**0.792**	**0.921**	0.383	0.808

in [Johnson et al., 2019] with 220k reports. We obtain the report sentences after de-duplication. For fine-grained pathological labels, we use CheXbert [Irvin et al., 2019], an automated deep-learning based chest radiology report labeler trained with MIMIC-CXR data (therefore no domain shift occurs), to label 14 fine-grained diseases. We keep sentences that have at most one disease noted. We end up with 1,129,478 sentences. The 14 diseases and statistics are listed in Table 10.3. This dataset is used in pretraining of the chest rewriting experiment.

VA-Rest The remaining unannotated sentences of the VA corpus [Yan et al., 2022] are used as a contrastive pretraining corpus. VA contains general medical reports covering different body parts, therefore, CheXbert is not applicable. Instead, we use clustering results as pseudo-labels for different fine-grained pathological patterns. We first obtain the sentence representations by feeding them into a RadBERT model [Yan et al., 2022], which is finetuned with the VA corpus by language modeling, and extracting the last hidden states. Then we reduce the dimension to D with Uniform Manifold Approximation and Projection [McInnes et al., 2018]. Based on the reduced embeddings, sentences are clustered with a Gaussian Mixture Model into K clusters. After experimenting with different parameters, we notice $K = 14$ and $D = 256$ achieves a good Silhouette score. See more in Appendix Section "Clustering Details".

10.3.3 Baselines and Ablations

We follow the experiment design of Xu et al. [2022b], and choose baseline models that are commonly used and have publicly available code:

Table 10.6. Examples of rewriting by different models for ambiguous sentences from OpenI-Annotated.

Contradictory Findings	
Original Input	normal cardiac contour with atherosclerotic changes throughout the aorta.
KBR	normal heart contour with atherosclerotic changes throughout the aorta.
ST	normal cardiac contour with atherosclerotic changes throughout the aorta.
PPLM	unchknown tortuous cardiac contour unchanged tortuous atherosclerotic changes throughout the aorta.
DEPEN	diaphragmclerotic changes throughout the thoracic aorta.
Ours	The cardiac contour shows atherosclerotic changes throughout the aorta.
Medical Jargon	
Original Input	maybe secondary to prominent mediastinal fat or tortuous.
KBR	maybe secondary to prominent mediastinum palmitic acid or tortuous.
ST	secondary to prior mediastinal or tortuous.
PPLM	optional secondary to the calcifiedsecondsmediastinal fat or tortuous. Include.
DEPEN	ouching compared the to the mediastinal fat or tortuous.
Ours	maybe due to the mediastinal fat or tortuous.

- **Knowledge-Based Replacement (KBR)** regenerates a sentence by replacing ambiguous terms with unambiguous alternatives. Following the previous work [Qenam et al., 2017], we build a dictionary for replacement by looking up the Consumer Health Vocabulary.[3] We notice that difficult special terms are also replaced with their layman language.

- **Style Transformer (ST)** A strong style-transfer model [Dai et al., 2019a] with adversarial training and a transformer architecture to transfer style while preserving content by reconstruction.

- **Controllable Generation** We include two perturbation-based controllable generation models – *PPLM* [Dathathri et al., 2019] and *DEPEN* [He et al., 2021c]. PPLM is a decoder-based language model but not capable of regeneration. In order to use it in our task, we modify it into a Seq2Seq model. We also adapt DEPEN so that it generates a less ambiguous sentence, as it is originally proposed for bias neutralization rewriting.

After adapting PPLM and DEPEN, they can be regarded as ablations – PPLM can be considered as our algorithm without contrastive pretraining and 'detect' steps, while DEPEN is

[3]Availalbe as a part of the UMLS https://www.nlm.nih.gov/research/umls/index.html

ours without contrastive pretraining.

10.3.4 Evaluation Metrics

Following the evaluation of Xu et al. [2022b], we compare rewritten results from the following aspects.

- **Disambiguation:** We measure the level of ambiguity using the accuracy of a Bert classifier, which is finetuned to predict ambiguity labels in OpenI-Annotated or VA-Annotated. The accuracy deduction $\Delta\text{Acc}_{\text{Am}}$ is regraded as disambiguation performance.

- **Fidelity:** We evaluate fidelity at two granularities: (1) a coarse-grained one which evaluates the persistence of the original abnormality label, measured by the accuracy gap $\Delta\text{Acc}_{\text{Dis}}$ from a Bert classifier finetuned to predict abnormality. (2) a fine-grained one which evaluates the match of pathology, measured by the match rate of CheXbert labeled results or pseudo-labels.

- **Language Quality:** Following He et al. [2021c], we use Pseudo-Log-Likelihood (PLL) [Salazar et al., 2020b] score to measure language fluency.

10.3.5 Human Evaluation

Rewritten results generated with different models are reviewed by radiology experts. For an ambiguous sentence, the rewritten result and its associated abnormality labels are shown to reviewers simultaneously. Reviewers decide (1) if the rewriting is successful in disambiguation; and (2) if the original content has been preserved by rewriting. As for the second one, our reviewers have a rigorous objective that includes language quality evaluation – a rewrite will be considered as a failure if there are any significant changes from the original findings or if proper English is not used. We collect the results of human evaluations and calculate the disambiguation and fidelity success rates.

10.4 Results and Analysis

10.4.1 Performance Comparison

The automatic evaluation results are shown in Table 10.4. Notably, it is sub-optimal to achieve the lowest ambiguity while generating a destroyed sentence. Therefore, we believe a good model is the one with an optimal balance between disambiguation rewriting and content preservation. We illustrate the trade-off between disambiguation and fidelity in Figure 10.3, where the upper-right corner indicates a good model. Our rewriting model resides at the upper-right corner in the two experiments, indicating a superior balance between disambiguation rewriting and content fidelity. This also agrees with human evaluation results shown in Table 10.5.

We discuss more about the results in the following. First, we compare our model with ST. Though it has a reasonable disambiguation (0.501 on OpenI-Annotated and 0.311 on VA-Annotated), ST has bad fidelity scores in both coarse-grained and fine-grained evaluations (the worst one on VA-Annotated). The generation quality is also worse compared with other models. We notice a rewriting from ST usually changes the original sentences significantly on both OpenI-Annotated and VA-Annotated, which explains why ST is able to disambiguate while fails in preserve fidelity. We provide our conjecture about the underlying reason: as an end-to-end model trained with multiple objectives at the same time, ST is more fragile when balancing objectives, making it difficult to find the sweet point between rewriting for disambiguation and content preservation.

Then, we compare our model with controllable generation baselines – PPLM and DEPEN. PPLM is a variation of our model without the detect step and constrastive pretraining step. Without the detect step, unnecessary edits can be applied, as the model knows little about which parts are ambiguous. And without contrastive pretraining to inject distinguishable domain knowledge, the model will fail to preserve the main pathological content, having bad content fidelity in the end. Therefore, PPLM is not effective at both disambiguation and fidelity on both OpenI-Annotated and VA-Annotated. DEPEN shows improvements on disambiguation and

maintains the original abnormality compared with PPLM, as the detect stage is added. But it fails to preserve fine-grained pathology match due to the lack of contrastive pretraining. Our model has the best overall performance in disambiguation and fidelity at different granularities. The improvement between PPLM, DEPEN, and ours indicates the effectiveness of each component in our model.

We notice a clear difference in performance of KBR – it fails to disambiguate in OpenI-Annotated while it becomes a strong baseline in VA-Annotated by achieving the best in both disambiguation and fidelity. We conjecture the reasons to be domain difference. We discuss the divergence below.

10.4.2 Specific Domain vs. General Domain

As one can observe in Table 10.4, our neural rewriting model is able to substantially outperform other baselines on OpenI-Annotated (specific domain). This indicates that given a reasonable amount of training data, our framework can perform well for a particular domain. On the VA dataset (general domain), KBR becomes a strong baseline. We notice that when creating the dictionary, human medical experts are good at proposing jargons across broadly different diseases and organs in general healthcare domains. However proposing terms that are specific to a domain requires deeper knowledge in that particular discipline. Therefore, VA-Annotated is more well-covered by the dictionary than OpenI-Annotated which is specific to the chest domain. We found the dictionary coverage rate is 17.3% on OpenI-Annotated while 21.5% on VA-Annotated, which explains why replacing works better in VA-Annotated.

However, since knowledge-based models come with a price of dictionary compiling and human (especially expert) effort, it may be difficult to extend them to solve domain-specific problems as each domain requires significant workload and expert experience. Instead, our rewriting framework is potentially a more promising direction to explore for this task, as it alleviates human effort while achieving competitive or even better performance.

10.4.3 Case Study

We show some examples in Table 10.6. More examples can be found in the Appendix Section "Examples". Findings in the first example are labeled as abnormal. The contradictory usage of "*normal*" and "*atherosclerotic changes*" in the sentence makes patients confused about the abnormality. As shown in this example, KBR replaces the special term with layman language (*cardiac* $\rightarrow$ *heart*), but this does not help disambiguation since there is still a contradiction. These limitations suggest that replacement-based models cannot handle patterns outside the dictionary or patterns at the sentence level. ST fails to rewrite a sentence. PPLM suffers from repetition issues and generates output that is not comprehensible. DEPEN can target the editing area with its detect step. However it fails to maintain fidelity without contrastive pretraining, and involves new findings that are inaccurate and change the original content drastically. However, our model achieves successful disambiguation by rewriting a contradiction-free sentence with minimal editing (*normal* $\rightarrow$ *The*) while maintaining fidelity by preserving the original abnormality (*atherosclerotic changes*).

Ambiguity in the second example is caused by medical jargon "*secondary to*", which implies "*mediastinal or tortuous* " is the reason for an abnormal finding. However, in regular usage, it means "*less important*" which is not the case or "*coming after*" which diminishes the causation. While other baselines either fail to disambiguate or introduce new content, ours is able to find a rewriting that mostly matches the context to describe the pathology causation.

10.5 Conclusion

Sharing medical information, especially reports with patients is essential to patient-centered care. Due to the communication gap between audiences, there is always ambiguity in reports, leading patients to be confused about their exam results. We collect and annotate two datasets containing radiology reports from healthcare systems in this study. We analyze and summarize three major causes of ambiguous reports: jargon, contradictions, and misleading

grammatical errors, and propose a framework for disambiguation rewriting. Experimental results show that our model can achieve effective disambiguation while maintaining content fidelity.

Chapter 10, in part, is a reprint of the material as it appears in ""Nothing abnormal": Disambiguating medical reports via contrastive knowledge infusion" by Zexue He, An Yan, Amilcare Gentili, Julian McAuley, and Chun-Nan Hsu, in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, no. 12, pp. 14232-14240. 2023. referenced as [He et al., 2023d]. The dissertation author was the primary investigator and author of this paper.

Chapter 11

Conclusion and Future Outlook

11.1 Summary of Contributions

This dissertation redefines the concept of human-centered NLP, extending it to include three foundational pillars: trustworthiness, learning from human cognition, and social good. Through these pillars, we aim to develop NLP systems that align more closely with human values and address critical societal challenges.

In the first part of this dissertation, we focus on enhancing the trustworthiness of NLP models by addressing key criteria such as robustness, fairness, interpretability, interactivity, and harmlessness. We begin by analyzing the lack of consistent effectiveness in current models and introduce TDG, the first collaborative framework that combines human and machine efforts to generate targeted data for challenging subgroups, significantly improving model robustness while mitigating human efforts in creating data augmentation. Another major contribution is the development of Interpretable Debiasing, a novel bias mitigation paradigm that strikes a balance between reducing biases and maintaining model performance by selectively exposing biased information. Building on this, INTERFAIR integrates interactivity into the framework, enabling human users to control and adjust the balance between fairness and task completion. The significant impact behind this work is that the definition of fairness can be dynamic and subjective. These two works introduce the innovative idea that optimizing one trustworthiness criterion can positively influence others, providing a fresh perspective to existing research.

Additionally, we propose DePeN, a framework designed to neutralize personal data, reducing the leakage of sensitive information without compromising the data's original meaning, which represents a significant advancement in AI safety. Lastly, we present our synthetic pre-training idea on artificial synthetic data, a new approach that achieves competitive results compared to traditional methods while significantly reducing toxicity and protecting data privacy. Together, these contributions not only enhance the model's robust performance but also offer valuable insights and set the stage for future research in AI Ethics.

Our contributions in the second part of this dissertation lie in the novel direction we propose for building human-centered NLP systems by demonstrating that learning from human cognition can lead to better AI. This claim is first validated through a study on human-like cognitive biases exhibited by LLMs, where we identify biases that lead to inconsistent and unfair responses and propose effective mitigation strategies. This work provides a pioneering perspective and establishes a framework for designing trustworthy LLMs that address not only social biases but also cognitive biases. Besides that, we also support our claim by introducing CAMELoT, a novel Transformer architecture inspired by the human brain's memorization mechanisms. By integrating these mechanisms into the model design, CAMELoT significantly improves the long-context modeling capabilities of language models, showcasing the practical benefits of applying human cognitive principles to advance NLP system development.

In the third part of this dissertation, we focus on impactful applications for social good, particularly in the domain of healthcare. One key contribution is MedEval, the first large-scale, multi-level, multi-domain benchmark with expert annotations which alleviates the issue of data scarcity in AI for healthcare. By providing a comprehensive testbed for evaluating advanced language models, MedEval offers valuable insights and serves as a practical guide for future research on leveraging LLMs to improve healthcare practices. Another significant contribution is a real-world application in which we deploy LLMs to rewrite medical reports into patient-friendly language, enabling laypersons to better understand their health information. This approach demonstrates the potential of LLMs to create meaningful, positive outcomes in healthcare,

ultimately enhancing patient-centered care.

11.2 Future Outlook

Beyond the directions covered in this dissertation, there remains significant potential for future work to advance human-centered NLP.

Bridging the Gap Between Individual Trustworthiness Criteria.

While existing research often focuses on individual trustworthiness criteria—such as fairness, robustness, safety, or harmlessness – there is significant potential in exploring how these criteria can complement each other to enhance overall trustworthiness. For example, Interpretable Debiasing addresses the limitations of black-box bias mitigation methods by introducing transparency and explainability into the process, while InterFair achieves higher human satisfaction by integrating fairness with human interactivity. These approaches demonstrate the potential of combining multiple criteria to achieve more holistic and effective solutions. Future research can build on this vision by investigating strategies that bridge the gaps between different trustworthiness criteria, ensuring that NLP systems are not only more accurate and effective but also aligned better with human values.

Interdisciplinary Research in Learning from Human Understanding.

Achieving trustworthy NLP for high-stakes tasks requires not only better model design but also a deeper understanding of human cognitive processes and behaviors, including the following perspectives in future works:

1. **Not Just Stereotypical Societal Biases.** Although social biases like gender bias are well-studied, cognitive biases are rarely explored. Cognitive biases such as anchoring bias originate from biased information-processing strategies and are not necessarily linked to minority groups, making them harder to detect and mitigate. My recent work [Echterhoff et al., 2024] formulated computational axioms for six cognitive biases, highlighting

shortcomings in current LLMs and failures of existing debiasing algorithms. But besides this, many questions such as: why and when do these biases occur? Are they due to biased pre-training data, reinforcement learning from human feedback, or biased human-machine interactions? remains unclear, requiring future interdisciplinary collaboration between cognitive science and computational psychology.

2. **Subjective Fairness: Personalized Definition and Mitigation.** Recent studies usually define fairness along a single axis such as absolute equality between different groups. However, cognitive research shows that the brain processes fairness through complex neural pathways involving areas like the prefrontal cortex, which are not purely logical but integrate personal values and societal norms [Güroğlu et al., 2010]. This makes perceptions of fairness deeply subjective. As the first attempt in this direction, my work [Majumder et al., 2022] enables users to identify their own fairness norms and expected goals by implementing a platform that parses their natural language feedback. Nonetheless, my future research interests also include studying the neural and psychological correlates of fairness, as well as the influence of human factors and contexts on AI fairness.

3. **Next-Generation Trustworthy Model Design: Cognitive Architectures and NLP.** Collaborating with cognitive science researchers, our recent work found that current NLP models are capable of memorizing, reasoning, and interacting with their environment (e.g., LLMs). We defined these models as individual instances of the *Lifespan Cognitive System* — a concept similar to how human cognition evolves over a lifetime [Wang et al., 2024]. However, we noticed these models may not perfectly align with the cognitive processes of the human brain. In CAMELoT [He et al., 2024], I equipped an LLM with memory systems similar to those of the human brain, boosting its performance in modeling longer contexts. This demonstrates a promising direction: integrating human cognitive architectures into the design of next-generation NLP models.

More Opportunities in AI for Social Good.

There are numerous areas within AI for Social Good that this dissertation does not explore but hold immense potential for future research. For instance, AI for Climate initiatives could include applications such as wildlife conservation and recovery [Engel et al., 2019], as well as tools for monitoring and mitigating environmental impact. In education, AI could enable personalized learning experiences and develop assistive tools to support children with disabilities, enhancing accessibility and inclusivity. Additionally, AI for Civic Engagement and Public Policy presents opportunities in areas like crime prediction and prevention, urban planning, and fostering community participation. These unexplored topics represent promising directions for advancing AI's role in addressing global challenges and creating meaningful societal impact.

Bibliography

Alham Fikri Aji, Nikolay Bogoychev, Kenneth Heafield, and Rico Sennrich. In neural machine translation, what does transfer learning transfer? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7701–7710, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.688. URL https://aclanthology.org/2020.acl-main.688.

Pinkesh Badjatiya, Manish Gupta, and Vasudeva Varma. Stereotypical bias removal for hate speech detection task using knowledge-based generalizations. In *WWW*, 2019.

Jasmijn Bastings, Wilker Aziz, and Ivan Titov. Interpretable neural predictions with differentiable binary variables. In *ACL*, Florence, Italy, July 2019. doi: 10.18653/v1/P19-1284. URL https://aclanthology.org/P19-1284.

Astrid Bertrand, Rafik Belloum, James R Eagan, and Winston Maxwell. How cognitive biases affect xai-assisted decision-making: A systematic review. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, pages 78–91, 2022.

Su Lin Blodgett, Johnny Wei, and Brendan O'Connor. Twitter universal dependency parsing for african-american and mainstream american english. In *ACL*, pages 1415–1425, 2018.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in neural information processing systems*, 29, 2016.

Shikha Bordia and Samuel R Bowman. Identifying and reducing gender bias in word-level language models. *NAACL-HLT*, 2019.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In Lluís Màrquez, Chris Callison-Burch, and Jian Su, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075. URL https://aclanthology.org/D15-1075.

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

Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334), 2017.

NEJM Catalyst. What is patient-centered care? *NEJM Catalyst*, 3(1), 2017.

Yapei Chang, Kyle Lo, Tanya Goyal, and Mohit Iyyer. Booookscore: A systematic exploration of book-length summarization in the era of llms. In *The Twelfth International Conference on Learning Representations*, 2024.

David Chiang. Hierarchical phrase-based translation. *computational linguistics*, 33(2):201–228, 2007.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. URL https://vicuna.lmsys.org.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models, 2022.

Australian Law Reform Commission et al. Judicial impartiality: Cognitive and social biases in judicial decision making. *Background Paper, April*, 16:2021, 2021.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116, 2019. URL http://arxiv.org/abs/1911.02116.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*, 2022.

Ning Dai, Jianze Liang, Xipeng Qiu, and Xuan-Jing Huang. Style transformer: Unpaired text style transfer without disentangled latent representation. In *ACL*, pages 5997–6007, 2019a.

Ning Dai, Jianze Liang, Xipeng Qiu, and Xuanjing Huang. Style transformer: Unpaired text style transfer without disentangled latent representation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5997–6007, Florence, Italy, July 2019b. Association for Computational Linguistics. doi: 10.18653/v1/P19-1601. URL https://aclanthology.org/P19-1601.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*, 2019c.

Bhavana Dalvi, Oyvind Tafjord, and Peter Clark. Towards teachable reasoning systems. *arXiv preprint arXiv:2204.13074*, 2022.

Prithiviraj Damodaran. Parrot: Paraphrase generation for NLU., 2021. https://github.com/PrithivirajDamodaran/Parrot_Paraphraser.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. In *ICLR*, 2019.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=H1edEyBKDS.

Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *FAT*, 2019.

Dina Demner-Fushman, Marc D Kohli, Marc B Rosenman, Sonya E Shooshan, Laritza Rodriguez, Sameer Antani, George R Thoma, and Clement J McDonald. Preparing a collection of radiology examinations for distribution and retrieval. *Journal of the American Medical Informatics Association*, 23(2):304–310, 2016.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*, 2019. URL https://api.semanticscholar.org/CorpusID:52967399.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. Eraser: A benchmark to evaluate rationalized nlp models. In *ACL*, 2020.

Jwala Dhamala, Tony Sun, Varun Kumar, Satyapriya Krishna, Yada Pruksachatkun, Kai-Wei

Chang, and Rahul Gupta. Bold: Dataset and metrics for measuring biases in open-ended language generation. In *FAT*, 2021.

Chenchen Ding, Masao Utiyama, and Eiichiro Sumita. Nova: A feasible and flexible annotation system for joint tokenization and part-of-speech tagging. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 18(2):1–18, 2018.

Yadin Dudai. The neurobiology of consolidations, or, how stable is the engram? *Annu. Rev. Psychol.*, 55:51–86, 2004.

Jessica Echterhoff, Yao Liu, Abeer Alessa, Julian McAuley, and Zexue He. Cognitive bias in high-stakes decision-making with llms. *arXiv preprint arXiv:2403.00811*, 2024.

Jessica Maria Echterhoff, Matin Yarmand, and Julian McAuley. Ai-moderated decision-making: Capturing and balancing anchoring bias in sequential decision tasks. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pages 1–9, 2022.

Jesse Engel, Kumar Krishna Agrawal, Shuo Chen, Ishaan Gulrajani, Chris Donahue, and Adam Roberts. GANSynth: Adversarial neural audio synthesis. In *ICLR*, 2019.

Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, pages 1180–1189, 2015.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020a.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020b.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A Smith. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3356–3369, 2020.

Marjan Ghazvininejad, Xing Shi, Jay Priyadarshi, and Kevin Knight. Hafez: an interactive poetry generation system. In *ACL, System Demonstrations*, pages 43–48, 2017.

Arthur M Glenberg, Margaret M Bradley, Jennifer A Stevenson, Thomas A Kraus, Marilyn J Tkachuk, Ann L Gretz, Joel H Fish, and BettyAnn M Turpin. A two-process account of long-term serial position effects. *Journal of Experimental Psychology: Human Learning and Memory*, 6(4):355, 1980.

Seraphina Goldfarb-Tarrant, Adam Lopez, Roi Blanco, and Diego Marcheggiani. Bias beyond

English: Counterfactual tests for bias in sentiment analysis in four languages. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4458–4468, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.272. URL https://aclanthology.org/2023.findings-acl.272.

Hila Gonen and Yoav Goldberg. Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them. In *NAACL-HLT*, 2019.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzman, and Angela Fan. The flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538, 2022.

Andrew J Gunn, Dushyant V Sahani, Susan E Bennett, and Garry Choy. Recent measures to improve radiology reporting: perspectives from primary care physicians. *Journal of the American College of Radiology*, 10(2):122–127, 2013.

Berna Güroğlu, Wouter van den Bos, Serge ARB Rombouts, and Eveline A Crone. Unfair? it depends: neural correlates of fairness in social context. *Social cognitive and affective neuroscience*, 5(4):414–423, 2010.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 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, 2020a.

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

David L Hamilton and Robert K Gifford. Illusory correlation in interpersonal perception: A cognitive basis of stereotypic judgments. *Journal of Experimental Social Psychology*, 12(4): 392–407, 1976.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *NeurIPS*, 2016.

Michael P. Hartung, Ian C. Bickle, Frank Gaillard, and Jeffrey P. Kanne. How to create a great radiology report. *RadioGraphics*, 40(6):1658–1670, 2020. doi: 10.1148/rg.2020200020. URL https://doi.org/10.1148/rg.2020200020. PMID: 33001790.

Philipp Harzig, Yan-Ying Chen, Francine Chen, and Rainer Lienhart. Addressing data bias problems for chest x-ray image report generation. *arXiv preprint arXiv:1908.02123*, 2019.

Martie G Haselton, Daniel Nettle, and Paul W Andrews. The evolution of cognitive bias. *The handbook of evolutionary psychology*, pages 724–746, 2015.

Zexue He, Bodhisattwa Prasad Majumder, and Julian McAuley. Detect and perturb: Neutral rewritting of biased and sensitive text via gradient-based decoding. In *Findings of the Association for Computational Linguistics: EMNLP2021*, pages 4173–4181. Association for Computational Linguistics, 2021a.

Zexue He, Bodhisattwa Prasad Majumder, and Julian McAuley. Detect and perturb: Neutral rewriting of biased and sensitive text via gradient-based decoding. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4173–4181, Punta Cana, Dominican Republic, November 2021b. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-emnlp.352. URL https://aclanthology.org/2021.findings-emnlp.352.

Zexue He, Bodhisattwa Prasad Majumder, and Julian McAuley. Detect and perturb: Neutral rewriting of biased and sensitive text via gradient-based decoding. *arXiv preprint arXiv:2109.11708*, 2021c.

Zexue He, Yu Wang, Julian McAuley, and Bodhisattwa Prasad Majumder. Controlling bias exposure for fair interpretable predictions. *Findings of EMNLP*, 2022.

Zexue He, Graeme Blackwood, Rameswar Panda, Julian McAuley, and Rogerio Feris. Synthetic pre-training tasks for neural machine translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8080–8098, 2023a.

Zexue He, Marco Tulio Ribeiro, and Fereshte Khani. Targeted data generation: Finding and fixing model weaknesses. *arXiv preprint arXiv:2305.17804*, 2023b.

Zexue He, Yu Wang, An Yan, Yao Liu, Eric Chang, Amilcare Gentili, Julian McAuley, and Chun-nan Hsu. Medeval: A multi-level, multi-task, and multi-domain medical benchmark for language model evaluation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8725–8744, 2023c.

Zexue He, An Yan, Amilcare Gentili, Julian McAuley, and Chun-Nan Hsu. "Nothing abnormal": Disambiguating medical reports via contrastive knowledge infusion. In *Proceedings of the 37th AAAI Conference on Artificial Intelligence*, 2023d. arXiv preprint arXiv:2305.08300.

Zexue He, Leonid Karlinsky, Donghyun Kim, Julian McAuley, Dmitry Krotov, and Rogerio Feris. Camelot: Towards large language models with training-free consolidated associative memory. *In ICML Workshop on Long Context Foundation Models*, 2024.

Melissa Heikkilä. Ai language models are rife with different political biases. *MIT Technology Review*, 2023.

Stefan Heindorf, Yan Scholten, Gregor Engels, and Martin Potthast. Debiasing vandalism detection models at wikidata. In *WWW*, 2019.

John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *arXiv preprint arXiv:1904.05342*, 2019.

Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597, 2019.

Sarthak Jain, Sarah Wiegreffe, Yuval Pinter, and Byron C Wallace. Learning to faithfully rationalize by construction. In *ACL*, pages 4459–4473, 2020.

Sophie Jentzsch, Patrick Schramowski, Constantin Rothkopf, and Kristian Kersting. Semantics derived automatically from language corpora contain human-like moral choices. In *AIES*, 2019.

Alistair EW Johnson, Tom J Pollard, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Yifan Peng, Zhiyong Lu, Roger G Mark, Seth J Berkowitz, and Steven Horng. Mimic-cxr-jpg, a large publicly available database of labeled chest radiographs. *arXiv preprint arXiv:1901.07042*, 2019.

Erik Jones and Jacob Steinhardt. Capturing failures of large language models via human cognitive biases. *Advances in Neural Information Processing Systems*, 35:11785–11799, 2022.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*, 2016.

Daniel Kahneman, Paul Slovic, and Amos Tversky. *Judgment under uncertainty: Heuristics and biases*. Cambridge university press, 1982.

Fereshte Khani and Marco Tulio Ribeiro. Collaborative development of nlp models. *arXiv preprint arXiv:2305.12219*, 2023.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie

Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. Dynabench: Rethinking benchmarking in NLP. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.324. URL https://aclanthology.org/2021.naacl-main.324.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Svetlana Kiritchenko and Saif Mohammad. Examining gender and race bias in two hundred sentiment analysis systems. In *SEM*, 2018.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, pages 177–180, 2007.

Dmitry Krotov and John J Hopfield. Dense associative memory for pattern recognition. *Advances in neural information processing systems*, 29, 2016.

Himabindu Lakkaraju, Dylan Slack, Yuxin Chen, Chenhao Tan, and Sameer Singh. Rethinking explainability as a dialogue: A practitioner's perspective. *arXiv preprint arXiv:2202.01875*, 2022.

Septina Dian Larasati. Identic corpus: Morphologically enriched indonesian-english parallel corpus. In *LREC*, pages 902–906, 2012.

Md Tahmid Rahman Laskar, Xue-Yong Fu, Cheng Chen, and Shashi Bhushan Tn. Building real-world meeting summarization systems using large language models: A practical perspective. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 343–352, 2023.

Yann LeCun, Sumit Chopra, Raia Hadsell, Marc'Aurelio Ranzato, and Fu Jie Huang. A tutorial on energy-based learning. *To appear in "Predicting Structured Data*, 1:0, 2006.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. Rationalizing neural predictions. In *EMNLP*,

Austin, Texas, November 2016. doi: 10.18653/v1/D16-1011. URL https://aclanthology.org/D16-1011.

Chantel J Leung, Jenny Yiend, Antonella Trotta, and Tatia MC Lee. The combined cognitive bias hypothesis in anxiety: A systematic review and meta-analysis. *Journal of Anxiety Disorders*, 89:102575, 2022.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 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, 2020.

Shuyang Li, Bodhisattwa Prasad Majumder, and Julian McAuley. Self-supervised bot play for transcript-free conversational recommendation with rationales. In *Proceedings of the 16th ACM Conference on Recommender Systems*, pages 327–337, 2022a.

Zichao Li, Prakhar Sharma, Xing Han Lu, Jackie CK Cheung, and Siva Reddy. Using interactive feedback to improve the accuracy and explainability of question answering systems post-deployment. *arXiv preprint arXiv:2204.03025*, 2022b.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *CoRR*, abs/2001.08210, 2020. URL https://arxiv.org/abs/2001.08210.

Kaiji Lu, Piotr Mardziel, Fangjing Wu, Preetam Amancharla, and Anupam Datta. Gender bias in neural natural language processing. In *Logic, Language, and Security*, pages 189–202. Springer, 2020.

Carolyn Mair, Martin Shepperd, et al. Debiasing through raising awareness reduces the anchoring bias. -, 2014.

Bodhisattwa Prasad Majumder, Taylor Berg-Kirkpatrick, Julian J. McAuley, and Harsh Jhamtani. Unsupervised enrichment of persona-grounded dialog with background stories. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *ACL*, pages 585–592, 2021a.

Bodhisattwa Prasad Majumder, Sudha Rao, Michel Galley, and Julian J. McAuley. Ask what's missing and what's useful: Improving clarification question generation using global knowledge. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tür, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, *NAACL-HLT*, pages 4300–4312, 2021b.

Bodhisattwa Prasad Majumder, Zexue He, and Julian McAuley. Interfair: Debiasing with natural language feedback for fair interpretable predictions. *arXiv preprint arXiv:2210.07440*, 2022.

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861, 2018.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D Manning. Fast model editing at scale. *arXiv preprint arXiv:2110.11309*, 2021.

Marina I Mityul, Brian Gilcrease-Garcia, Mark D Mangano, Jennifer L Demertzis, and Andrew J Gunn. Radiology reporting: current practices and an introduction to patient-centered opportunities for improvement. *American Journal of Roentgenology*, 210(2):376–385, 2018.

Dong Nguyen, Rilana Gravel, Dolf Trieschnigg, and Theo Meder. How old do you think i am? a study of language and age in twitter. In *ICWSM*, volume 7, 2013.

Luke Oakden-Rayner, Jared Dunnmon, Gustavo Carneiro, and Christopher Ré. Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. In *Proceedings of the ACM conference on health, inference, and learning*, pages 151–159, 2020.

Seong Cheol Oh, Tessa S Cook, and Charles E Kahn. Porter: a prototype system for patient-oriented radiology reporting. *Journal of digital imaging*, 29(4):450–454, 2016.

OpenAI. Chatgpt: Language model, 2024. URL https://chat.openai.com/. Accessed: 2024-10-09.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A fast, extensible toolkit for sequence modeling. *CoRR*, abs/1904.01038, 2019. URL http://arxiv.org/abs/1904.01038.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 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, 2002.

Ji Ho Park, Jamin Shin, and Pascale Fung. Reducing gender bias in abusive language detection. In *EMNLP*, 2018.

Yifan Peng, Shankai Yan, and Zhiyong Lu. Transfer learning in biomedical natural language processing: An evaluation of bert and elmo on ten benchmarking datasets. In *Proceedings of the 2019 Workshop on Biomedical Natural Language Processing (BioNLP 2019)*, pages 58–65, 2019.

Ofir Press, Noah A. Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation, 2022.

Basel Qenam, Tae Youn Kim, Mark J Carroll, Michael Hogarth, et al. Text simplification using consumer health vocabulary to generate patient-centered radiology reporting: translation and evaluation. *Journal of medical Internet research*, 19(12):e8536, 2017.

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

Jack W Rae, Anna Potapenko, Siddhant M Jayakumar, Chloe Hillier, and Timothy P Lillicrap. Compressive transformers for long-range sequence modelling. *arXiv preprint*, 2019. URL https://arxiv.org/abs/1911.05507.

Nazneen Rajani, Weixin Liang, Lingjiao Chen, Meg Mitchell, and James Zou. Seal: Interactive tool for systematic error analysis and labeling. *arXiv preprint arXiv:2210.05839*, 2022.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David Kreil, Michael K Kopp, et al. Hopfield networks is all you need. In *International Conference on Learning Representations*, 2021.

Charvi Rastogi, Marco Tulio Ribeiro, Nicholas King, Harsha Nori, and Saleema Amershi. Supporting human-ai collaboration in auditing llms with llms. In *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*, pages 913–926, 2023.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection. In *ACL*, 2020.

Sravana Reddy and Kevin Knight. Obfuscating gender in social media writing. In *Workshop on NLP and Computational Social Science*, pages 17–26, 2016.

Nils Reimers and Iryna Gurevych. 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, 11 2019. URL http://arxiv.org/abs/1908.10084.

Marco Tulio Ribeiro and Scott Lundberg. Adaptive testing and debugging of NLP models. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3253–3267, Dublin, Ireland, May 2022a. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.230. URL https://aclanthology.org/2022.acl-long.230.

Marco Tulio Ribeiro and Scott Lundberg. Adaptive testing and debugging of nlp models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*

(Volume 1: Long Papers), pages 3253–3267, 2022b.

Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. Gender bias in coreference resolution. In *NAACL-HLT*, 2018.

Julian Salazar, Davis Liang, Toan Q Nguyen, and Katrin Kirchhoff. Masked language model scoring. In *ACL*, pages 2699–2712, 2020a.

Julian Salazar, Davis Liang, Toan Q Nguyen, and Katrin Kirchhoff. Masked language model scoring. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2699–2712, 2020b.

William Samuelson and Richard Zeckhauser. Status quo bias in decision making. *Journal of risk and uncertainty*, 1:7–59, 1988.

Susan J Sara. Retrieval and reconsolidation: toward a neurobiology of remembering. *Learning & memory*, 7(2):73–84, 2000.

Indira Sen, Mattia Samory, Claudia Wagner, and Isabelle Augenstein. Counterfactually augmented data and unintended bias: The case of sexism and hate speech detection. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, editors, *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4716–4726, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.347. URL https://aclanthology.org/2022.naacl-main.347.

Emily Sheng, Kai-Wei Chang, Prem Natarajan, and Nanyun Peng. The woman worked as a babysitter: On biases in language generation. In *EMNLP*, 2019.

Emily Sheng, Kai-Wei Chang, Prem Natarajan, and Nanyun Peng. Societal biases in language generation: Progress and challenges. In *ACL*, 2021.

Dylan Slack, Satyapriya Krishna, Himabindu Lakkaraju, and Sameer Singh. Talktomodel: Understanding machine learning models with open ended dialogues. *arXiv preprint arXiv:2207.04154*, 2022.

Nimit Sohoni, Jared Dunnmon, Geoffrey Angus, Albert Gu, and Christopher Ré. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. *Advances in Neural Information Processing Systems*, 33:19339–19352, 2020.

Aparna Srinivasa Babu and Michael L. Brooks. The malpractice liability of radiology reports:

Minimizing the risk. *RadioGraphics*, 35(2):547–554, 2015. doi: 10.1148/rg.352140046. URL https://doi.org/10.1148/rg.352140046. PMID: 25763738.

M Stewart, JB Brown, A Donner, IR McWhinney, J Oates, WW Weston, and J Jordan. The impact of patient-centered care on outcomes. *The Journal of family practice*, 49(9):796–804, 2000.

Moira Stewart, Judith Belle Brown, Wayne Weston, Ian R McWhinney, Carol L McWilliam, and Thomas Freeman. *Patient-centered medicine: transforming the clinical method*. CRC press, 2013.

Chloe Rose Stuart-Ulin. Microsoft's politically correct chatbot is even worse than its racist one. *Quartz Ideas*, 31, 2018.

Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. Mitigating gender bias in natural language processing: Literature review. In *ACL*, 2019.

Niket Tandon, Aman Madaan, Peter Clark, Keisuke Sakaguchi, and Yiming Yang. Interscript: A dataset for interactive learning of scripts through error feedback. *CoRR*, abs/2112.07867, 2021. URL https://arxiv.org/abs/2112.07867.

Niket Tandon, Aman Madaan, Peter Clark, and Yiming Yang. Learning to repair: Repairing model output errors after deployment using a dynamic memory of feedback. *NAACL Findings.(to appear)*, 2022.

Jiulin Teng. Bias dilemma: de-biasing and the consequent introduction of new biases. *HEC Paris Research Paper No. SPE-2013-1025*, 2013.

Jörg Tiedemann. Parallel data, tools and interfaces in opus. In *Lrec*, volume 2012, pages 2214–2218. Citeseer, 2012.

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

Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases: Biases in judgments reveal some heuristics of thinking under uncertainty. *science*, 185(4157): 1124–1131, 1974.

Amos Tversky and Daniel Kahneman. The framing of decisions and the psychology of choice. *science*, 211(4481):453–458, 1981.

Szymon Tworkowski, Konrad Staniszewski, Mikołaj Pacek, Yuhuai Wu, Henryk Michalewski, and Piotr Miłoś. Focused transformer: Contrastive training for context scaling. *arXiv preprint arXiv:2307.03170*, 2023.

Danil Tyulmankov, Ching Fang, Annapurna Vadaparty, and Guangyu Robert Yang. Biological learning in key-value memory networks. *Advances in Neural Information Processing Systems*, 34:22247–22258, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL http://arxiv.org/abs/1706.03762.

Mengting Wan and Julian McAuley. Item recommendation on monotonic behavior chains. In *RecSys*, pages 86–94, 2018.

Weizhi Wang, Li Dong, Hao Cheng, Xiaodong Liu, Xifeng Yan, Jianfeng Gao, and Furu Wei. Augmenting language models with long-term memory. *arXiv preprint arXiv:2306.07174*, 2023.

Yu Wang, Chi Han, Tongtong Wu, Xiaoxin He, Wangchunshu Zhou, Nafis Sadeq, Xiusi Chen, Zexue He, Wei Wang, Gholamreza Haffari, et al. Towards lifespan cognitive systems. *arXiv preprint arXiv:2409.13265*, 2024.

Matthew B Welsh, Steve H Begg, and Reidar B Bratvold. Efficacy of bias awareness in debiasing oil and gas judgments. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 29, 2007.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. CCNet: Extracting high quality monolingual datasets from web crawl data. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.494.

David J Willshaw, O Peter Buneman, and Hugh Christopher Longuet-Higgins. Non-holographic associative memory. *Nature*, 222(5197):960–962, 1969.

Yuhuai Wu, Felix Li, and Percy Liang. Insights into pre-training via simpler synthetic tasks. *arXiv preprint arXiv:2206.10139*, 2022a.

Yuhuai Wu, Markus N Rabe, DeLesley Hutchins, and Christian Szegedy. Memorizing transformers. *arXiv preprint arXiv:2203.08913*, 2022b.

Mengzhou Xia, Anjalie Field, and Yulia Tsvetkov. Demoting racial bias in hate speech detection. In *Workshop on NLP for Social Media*, 2020.

Canwen Xu, Zexue He, Zhankui He, and Julian McAuley. Leashing the inner demons: Self-detoxification for language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11530–11537, 2022a.

Qiongkai Xu, Lizhen Qu, Chenchen Xu, and Ran Cui. Privacy-aware text rewriting. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 247–257, 2019.

Wenda Xu, Michael Saxon, Misha Sra, and William Yang Wang. Self-supervised knowledge assimilation for expert-layman text style transfer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11566–11574, 2022b.

Mohammad Yaghini, Andreas Krause, and Hoda Heidari. A human-in-the-loop framework to construct context-aware mathematical notions of outcome fairness. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 1023–1033, 2021.

An Yan, Julian McAuley, Xing Lu, Jiang Du, Eric Y Chang, Amilcare Gentili, and Chun-Nan Hsu. Radbert: Adapting transformer-based language models to radiology. *Radiology: Artificial Intelligence*, 4(4):e210258, 2022.

Hitomi Yanaka, Koji Mineshima, Daisuke Bekki, Kentaro Inui, Satoshi Sekine, Lasha Abzianidze, and Johan Bos. Can neural networks understand monotonicity reasoning? In Tal Linzen, Grzegorz Chrupała, Yonatan Belinkov, and Dieuwke Hupkes, editors, *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 31–40, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4804. URL https://aclanthology.org/W19-4804.

Antonio Jimeno Yepes, Aurélie Névéol, Mariana Neves, Karin Verspoor, Ondřej Bojar, Arthur Boyer, Cristian Grozea, Barry Haddow, Madeleine Kittner, Yvonne Lichtblau, et al. Findings of the wmt 2017 biomedical translation shared task. In *Proceedings of the Second Conference on Machine Translation*, pages 234–247, 2017.

Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *AIES*, 2018.

Chong Zhang, Jieyu Zhao, Huan Zhang, Kai-Wei Chang, and Cho-Jui Hsieh. Double perturbation: On the robustness of robustness and counterfactual bias evaluation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3899–3916, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.305. URL https://aclanthology.org/2021.naacl-main.305.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *ICLR*, 2019.

Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen Mckeown, and Tatsunori B Hashimoto. Benchmarking large language models for news summarization. *Transactions of the Association for Computational Linguistics*, 11:39–57, 2024.

Xuhui Zhou, Maarten Sap, Swabha Swayamdipta, Yejin Choi, and Noah A Smith. Challenges in automated debiasing for toxic language detection. In *EACL*, 2021.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. Multilingual machine translation with large language models: Empirical results and analysis. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2765–2781, 2024.