Knowledge-Enhanced Social Content Analysis in Generative Modeling

Haoyang Wen

CMU-LTI-25-005

April 2025

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15123

Thesis Committee:

Alexander Hauptmann (Chair)	Carnegie Mellon University
Eduard Hovy	Carnegie Mellon University
Alexander Rudnicky	Carnegie Mellon University
Maarten Sap	Carnegie Mellon University
Heng Ji	University of Illinois Urbana-Champaign

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Language and Information Technology.

Keywords: natural language processing, generative modeling, social content analysis, large language models, information retrieval, retrieval-augmented generation

Abstract

With the rapid advancement in language modeling, we have witnessed great success in building natural language processing or multimedia analysis models capable of performing more complex reasoning and inference. Large models also possess parameterized knowledge and can perform knowledge-centric tasks without using externally stored knowledge. However, this paradigm is accompanied by issues such as outdated, inaccurate knowledge, or hallucinations. Automated social content analysis, on the other hand, is an area often closely intertwined with relevant, and in many cases, up-to-date background knowledge. Since the goal of automatic social content analysis includes inferring or discovering opinions, interests, trends, and insights from text or multimedia content, access to background information or knowledge is usually essential for a comprehensive understanding of the content. Therefore, with the advent of recent generative modeling methods, it is crucial to investigate whether it is still necessary to explicitly use and model the knowledge for social content analysis tasks, as well as to identify effective methods to incorporate the background knowledge.

In this thesis, we demonstrate that generative modeling can be effectively used to perform various social content analysis tasks. We also show that external knowledge is a powerful resource and can benefit the generative social content analysis model during the training and inference stages. For the training stage, we discuss methods to leverage knowledge to enhance generative model training, including transforming the external knowledge base into distant supervision for Twitter profile inference, and using abstract knowledge as training constraints to enhance entity-to-entity stance detection. For the inference stage, we primarily discuss methods to incorporate knowledge into analysis during inference. We first explore methods to find appropriate knowledge for inference with generative modeling, including multimodal reranking and generative retrieval on a domain-specific corpus. Then we discuss specific cases involving knowledge-seeking and knowledge-enhanced inference with generative modeling on social content analysis tasks, including zero-shot and few-shot stance detection, and extensions to multimodal content analysis.

Acknowledgments

I would not have imagined such an incredible journey when I started to learn basic computer science as a young kid. Looking back on this journey, I would not have made it without the support of many people around me.

First and foremost, I would like to thank my advisor, Professor Alexander Hauptmann. I always enjoy the chat with you about the philosophical discussion on what good research should be, and I have learned a lot from your smart ideas about research and project management. I am grateful to have joined the friendly Informedia family and met many brilliant people. I often think back to those times when someone said in the group chat that you were ready to meet students, and we just lined up outside your office door to talk to you and hear your advice.

I am also incredibly thankful to Professor Eduard Hovy. We have been collaborating on some research work, and I am always amazed by your wonderful thoughts. Thank you for being helpful all the time, regardless of my research, thesis, or suggestions for my personal development.

I want to express my special thanks to Professor Heng Ji, who is also my advisor for my master's degree. You opened the door to my research life in the United States and shaped my research skills. I am also grateful to have you as my external committee member and to continue to hear your advice on my research work and career path. I have always admired your dedication to work and your passion for the people around you.

I would like to acknowledge Professor Maarten Sap and Professor Alexander Rudnicky, my thesis committee members, who provided valuable feedback and suggestions on this thesis. Professor Maarten Sap offered to help me out and join my thesis committee at the last minute, and we had very constructive discussions on my thesis and my presentation. It is also always a pleasant experience to talk to Professor Alexander Rudnicky and listen to his advice on my thesis. I would especially like to acknowledge the help and support from Professor Jamie Callan, our PhD program director, who helped me a lot during the coordination of my thesis proposal and defense, especially whenever unusual situations happened. I really like your metaphor about the rain before my defense and the beautiful sunset after my defense.

A portion of this thesis was done when I was doing internships at Google and

Amazon Web Services, where I received tremendous support from my managers, mentors, and collaborators, including Dr. Michael Bendersky, Dr. Honglei Zhuang, Professor Hamed Zamani, Dr. Jiang Guo, Dr. Yi Zhang, Dr. Jiarong Jiang and Dr. Zhiguo Wang. They also provided a lot of help during my job search.

My heartfelt thanks go to a long but not comprehensive list of my friends in alphabet order: Krai Chamnivikaipong, Sanyuan Chen, Yingbo Chen, Ta-Chung Chi, Yang Gao, Liangke Gui, Jie He, Xuangui Huang, Manling Li, Wenhe Liu, Chang Liu, Hui Liu, Yuping Luo, Yukun Ma, Yuxiang Peng, Lingyi Peng, Yijun Qian, Huidong Sun, Yan Tang, Zekun Wang, Zilong Wang, Peng Wang, Qingyun Wang, Lei Wu, Yang Xu, Yiheng Xu, Ruihan Yang, Qingyu Yin, Lijun Yu, Pengfei Yu, Qi Zeng, Guanhua Zhang, Denghui Zhang, Xinran Zhao, Xiaoyu Zhu (please forgive me if I lost your name in the list). It is my greatest honor to have met and be friends with each of you at different stages of my life. I am not an extroverted person, and I often keep radio silence on social media platforms. But I cherish every time we reconnect and the help and support from you. I also want to thank my friends in the Pittsburgh Badminton Club. The regular badminton sessions are an essential support to keep me fit and keep my sanity, especially whenever I was close to burnout.

It would not be possible for me to be here without the unconditional love from my parents, Qian Ren and Limin Wen, who gave me a lot of freedom to explore what I am interested in, even when other parents are all anxious about their children's academic study. It is my greatest fortune to have known Anlan He at age 12, who later became my wife after 13 years. I often think back to the night we drove overnight from Champaign to Pittsburgh, which was the beginning of my PhD study life. I am so lucky to have you with me all the time on this journey. Thank you for the understanding, support, and companionship. I look forward to the next, and all the following chapters of our life.

Contents

Introduction **3**

Thesis Statement 4

Thesis Overview 5

Thesis Contributions 7

I Knowledge-Enhanced Training for Social Content Analysis with Generative Modeling **9**

1 Knowledge As Training Data: Towards Open-Domain Twitter Celebrity User Profile Inference **11**

1.1 Overview 11

1.2 Problem Definition and Dataset 14

1.3 Methods 17

1.4 Experiments 21

1.5 Related Work 27

1.6 Conclusion 28

1.7 Attribute Descriptions 28

2 Knowledge As Constraints: Transitive Consistency Constrained Learning for Entity-to-Entity Stance Detection **33**

2.1 Overview 33

2.2 Entity-to-Entity Stance Detection Frameworks 35

2.3 Transitive Consistency Constrained Learning 37

2.4 Experiments 41

2.5 Related Work 51

2.6 Conclusion 52

II Knowledge-Enhanced Inference: Knowledge-Seeking with Generative Modeling 53

3 On Synthetic Data Strategies for Domain-Specific Generative Retrieval 55

3.1 Overview . . . 55
3.2 Generative Retrieval Framework . . . 57
3.3 Supervised Fine-Tuning Data Strategy . . . 58
3.4 Preference Learning Data Strategy . . . 61
3.5 Experiments . . . 63
3.6 Related Work . . . 73
3.7 Conclusion . . . 74
3.8 LLM Prompts . . . 75

4 Multimodal Reranking for Knowledge-Intensive Visual Question Answering 79

4.1 Overview . . . 79
4.2 A Knowledge-Intensive Visual Question Answering Framework . . . 81
4.3 Multi-Modal Reranking . . . 83
4.4 Experiments . . . 85
4.5 Related Work . . . 90
4.6 Conclusion . . . 91

III Knowledge-Enhanced Inference: Applications on Social Content Analysis 93

5 Knowledge-Enhanced Topic Representation: Case Study on Text and Multimodal Social Content Analysis 95

5.1 Overview . . . 96
5.2 Approach . . . 97
5.3 Extension to Multi-Modal Analysis . . . 101
5.4 Experiments . . . 103
5.5 Related Work . . . 107
5.6 Conclusion . . . 108

Conclusion **111**

Summary of Dissertation Findings 111

Future Directions 113

Ethics Considerations **115**

Bibliography **119**

List of Figures

1 Overall outline of this thesis. 5

1.1 An example of paired WikiData and Twitter information. Relevant text spans with corresponding attribute values are highlighted with the same color. 13
1.2 The long tail distribution of different predicates. A few predicates have many examples, while most other predicates only have limited examples. 17
1.3 An example tweet and tag sequence for attribute `employer` and value `United Nations`. 18
1.4 The workflow of the generation-based method, which takes the combination of predicate, Twitter metadata, and a window of tweets as input for a T5-based model, and aggregate the window-level results into user level using majority vote. 19
1.5 Example window-level predictions from generation-based model with their context. 25

2.1 An example of three entity-to-entity stances and their consistency. If we know "Richard Pilger" was against the "attorney general", and the "attorney general" supported the "new policy", we may infer that "Richard Pilger" was also likely against the "new policy". 34
2.2 The overall framework of soft consistency constrained learning objective. We first sample an entity as the shared entity and use this entity to sample two sentences that can be used for stance inference. We then concatenate the two sentences, with combinations of entity pairs from the sampled three entities for entity-to-entity stance detection, and the objective is the penalty for inconsistent predictions. 38
2.3 The Micro F_1 performances of generation and classification models on DSE development set with different balance factor λ. In general, we can observe that with the increase of λ, the performance first improves and then degrades. 45

2.4 Prompt templates for BLOOMZ-176b and Llama2-70b-chat. 49

3.1 The overall workflow of the generative retrieval training and synthetic data utilization at each stage. 59
3.2 Jaccard similarity post-analysis on MultiHop-RAG test set. Synthetic queries from Mixtral 8x7b are generally closer to the test set than those from docT5query. Besides, incorporating granularity and domain-specific attributes further helps with getting queries that are closer to the test set. 69
3.3 Performance comparison between generative retrieval with semantic identifiers and off-the-shelf-retrieval models. We use HIT@4 for MultiHop-RAG and HIT@1 for other datasets as the metric. 73
3.4 Prompts for query generation. 75
3.5 Prompts for constraints-based query generation. 76
3.6 Prompts for query-answer pair generation. 77
3.7 Prompt for keywords-based document identifier generation. 78

4.1 An example from OK-VQA, which requires knowledge to associate deep-dish pizza and Chicago. 80
4.2 A basic KI-VQA framework, which first retrieves relevant top knowledge candidates with using visual question and then combine the question and retrieved knowledge candidates to generate the answer. The dashed box is our reranking module in Section 4.3. 81
4.3 Framework of multimodal reranking. 83

5.1 Overall framework of BART-based generation framework for stance detection. . 98
5.2 Prompts for rationale generation. 102
5.3 The t-SNE visualization of intermediate representations from our model and BERT classification model. Color map: Supportive, Opposite, Neutral. 106

List of Tables

1.1 Statistics of our collected data from WikiData and Twitter. 15

1.2 Comparison between datasets. Our data contains a diverse set of attributes, with more users and values obtained from WikiData. 16

1.3 System performance (%) on our constructed open-domain Twitter user profile inference dataset. 22

1.4 System performance (%) on the subset of the test set that we can find occurrences of attribute values in Twitter context. 23

1.5 Results on Li et al. [96] following the preprocessing as Qian et al. [144]. We re-evaluate the results based on user-level F_1. $p < 0.01$ for both F_1 comparisons. 24

1.6 Effects (%) of result filtering (-threshold), result aggregation (-aggregation) and Twitter metadata on development set. $p < 0.01$ for F_1 comparisons. 24

1.7 Attribute Description 29

1.7 Attribute Description 30

1.7 Attribute Description 31

1.7 Attribute Description 32

2.1 The transitive mapping of a pair of directed stances with a shared entity. "-" denotes no mapping between the pair of stances. We also do not apply transitive mapping for neutral samples. 38

2.2 Comparison of DSE and SEESAW datasets. 41

2.3 Results on DSE dataset. The performances of our methods are averaged performance (%) over 5 runs. 44

2.4 Results on SEESAW dataset. Different from the original setting of SEESAW, we provide entity pair and direction as the input and ask models to predict a non-neutral stance. The performances we reported are averaged performance (%) over 5 runs. 45

2.5 Comparison between vanilla data augmentation and soft consistency constrained learning. Both methods use the same two-step sampling method to obtain the inferred stance of the cross-sentence entity pair. 46
2.6 Effects of two-step sampling method compared to the vanilla uniform random sampling over all valid sentence pairs for stance inference on DSE test set. . . . 46
2.7 Analysis of BLOOMZ-176b performances on different test data on DSE, including test labels that do not require predicting direction, data excluding neutral samples, and test labels without both of them. The results show that the performance suffers from predicting the neutral labels and directional information. 47
2.8 Analysis of Llama2-70b-chat performances on different test data types on DSE. The results show that large language models suffer from predicting the neutral labels and directional information. 48
2.9 Case study to compare the differences between vanilla classification model and classification model with consistency transitive constrained learning. 50

3.1 Examples of different synthetic queries generated from MultiHop-RAG corpus. . 60
3.2 Dataset Statistics 63
3.3 Attributes used in each dataset for constraints-based query generation. 64
3.4 Ablation study on the effect of synthetic queries generated at a sentence-level granularity of context. 66
3.5 Ablation study on generative retrieval performances with or without the constraints-based synthetic queries. 67
3.6 Ablation study on generative retrieval performance trained with or without Context2ID data. The results demonstrate the helpfulness of Context2ID data and learning to memorize the context for generative retrieval. 67
3.7 Analysis on different ways of combining Query2ID and Context2ID data. We compare simple concatenation (Concat) and interleaving (Interleave) that inherently upsamples the Context2ID data. 68
3.8 Generative retrieval performance with synthetic queries from Mixtral 8x7b and docT5query. The results show that queries from Mixtral 8x7b can help train a better generative retrieval model. 68
3.9 Ablation study on atomic identifier-based generative retrieval performance on MultiHop-RAG. 70

3.10 Preference learning with different numbers of negative candidates. The results show that it is an effective strategy to select negative candidates with ranks higher than the positive candidate, while different numbers of negative candidates may optimize the retrieval performance in different ways. 70
3.11 Examples of synthetic queries generated from DocT5Query and Mixtral 8x7b. . 70
3.12 Comparisons to Off-The-Shelf Retrieval Models Across Datasets 72

4.1 Results comparison on OK-VQA dataset. 86
4.2 Results comparision on A-OKVQA dataset. 87
4.3 Effects of multimodal reranking, compared to model without retrieval and model without reranking. 87
4.4 Effects of multimodal ranking. We can find that learning reranker using distillation from the answer generator can instead hurt the performance. Our multimodal reranker trained with small data provides competitive performance compared to RankT5, which is pretrained on a large amount of data. 88
4.5 Comparison between multi-modal large models on OK-VQA datasets. We can find that our model provides promising performance compared to the zero-shot performance of those large multimodal models. 89
4.6 Effects of discrepancy between knowledge candidates for training and testing. $\rightarrow$ means the qualities of knowledge candidates in training and test are similar. $\searrow$ means the quality in training is better than test. $\nearrow$ means the quality in test is better than training. 89

5.1 A stance detection example from VAST. 96
5.2 Examples input and output templates for stance detection, target prediction, and unlikelihood training. 100
5.3 Performance of different model variants on the overall precision, recall, and F_1 on the development set (%). Each of our model variants is on top of the variant from its previous row. 105
5.4 Stance detection performance (%) on VAST. Our model significantly outperforms previous work on all metrics. Our results are obtained from averaging performances over 5 random seeds. $p < 0.001$ on overall F_1 using Z-test with variance as the standard deviation over multiple runs. 105
5.5 Performance of different model variants on the overall F_1 and Accuracy on the test set (%). 107

Introduction

Introduction

We have witnessed a great advancement in language modeling with the increase in model parameters and data for pre-training, fine-tuning, and preference learning [24, 74, 128, 135, 198]. These models [132, 162, 186, 220] show remarkable performance in many human language understanding and generation tasks, especially with impressive capabilities for zero-shot or few-shot analysis, as well as those tasks that require complex reasoning.

One of the concerns of current language modeling is the memorization and utilization of knowledge [51, 227]. As language models are usually trained with data collected by a cut-off date, they may not have the most up-to-date information with the rapid change in information in the current digital world [129, 224]. The nature of unconstrained autoregressive generation can also lead to hallucinated output, because there is no guarantee that token-by-token generation will always yield the correct information [43, 127, 154]. These phenomena, mostly due to the lack of accurate knowledge, can lead to incorrect predictions when applying language models to specific tasks.

However, automated social content analysis, which aims to discover opinions, interests, trends, and insights from text or multimedia content, is usually closely intertwined with relevant background knowledge [56, 185]. For example, to analyze the cultural background of a speaker, the model should have enough knowledge on the categorization of different backgrounds, the social norms for different communities, and how people with different backgrounds behave differently. The opinion analysis is also highly dependent on the context, for example, the polarity of the analyzed target or source with regard to politics, economics, or ideology. Without enough knowledge, the model may only rely on superficial lexical patterns in a context to make a prediction, which is an oversimplification, and can easily make mistakes when there are semantic or pragmatical nuances. Therefore, it is essential to investigate whether different kinds of knowledge are still important to building today's social content analysis models, especially with generative modeling. The choices of knowledge bases can be structured or semi-structured knowledge bases such as WikiData [1], prior or abstract knowledge,

and unstructured knowledge corpus such as a set of plain text or multimedia documents.

Thesis Statement

In this thesis, we systematically investigate different approaches to incorporating knowledge into content analysis models, with a focus on the generative modeling-based methods as with the trends of adapting large language models. We will discuss several typical methods for utilizing knowledge. The first type of method is to transform knowledge into training signals and to learn better models from it. We discuss transforming an external knowledge base into a large-scale annotation-free distantly supervised dataset for model training and inference and using abstract knowledge as soft logical constraints to conduct constrained learning. The second and most common type of method is to augment the model input with retrieved relevant knowledge information, which can be considered as additional evidence for model prediction and reasoning. We first study the development of the retrieval pipeline in a domain-specific setting with generative modeling and then study the applications of the retrieval-augmented generation framework to social content analysis.

Another important theme of this thesis is the methodology for content analysis modeling, where we have an emphasis on exploring generative modeling-based analysis, compared to the traditional classification- or extraction-based analysis. The generative modeling takes a text or multimedia sequence as input and predicts the output token autoregressively. We study methods to incorporate generative modeling into different content analysis tasks, which mainly transform the analysis prediction into generating a sequence of text tokens. In this thesis, we would like to have a rigorous comparison of the performance of generative modeling and traditional methods and where and how knowledge enhancement can further help the generative modeling-based methods.

In this thesis, among a wide variety of social content analysis tasks, we investigate tasks for knowledge enhancement with a theme to identify sensitive topics or issues concerning different backgrounds. Except for direct extraction from the context, these tasks typically require models to conduct inference to obtain information that may not explicitly occur in the context. These tasks can be divided into two different parts. The first part is to investigate the possibility of identifying the personal or social background from a social context. The second part focuses on the analysis of the subjective or objective stances expressed in the context. With the combination of these two parts, we can potentially obtain statistics of the opinions toward a certain topic with regard to different backgrounds, which can be especially helpful in drafting

more inclusive texts or statements and reducing unintentional offenses.

This thesis attempts to bridge the gap between the analysis of social content from a short snippet and the rich background information behind the text with advanced generative modeling methods. By involving knowledge in the modeling at different stages, our goal is to create analysis models that make more accurate, controllable, and faithful analyses.

Thesis Overview

This thesis aims to explore different methods of using knowledge to enhance generative models of social content analysis. This thesis is divided into three parts. The focus of the first part is to use knowledge to enable or enhance the training of social content analysis models. We will discuss two different methods: transforming the knowledge base into distantly supervised training data and adopting abstract knowledge as constraints for model training. The focus of the second part is on the foundation of knowledge-augmented generation, where we explore the settings of multimodal knowledge-augmented generation and the option to use LLMs as the retriever for domain-specific corpus. The focus of the third part is on the application of the knowledge-augmented generation paradigm for specific social content analysis tasks. The overall outline of the thesis is illustrated in Figure 1.

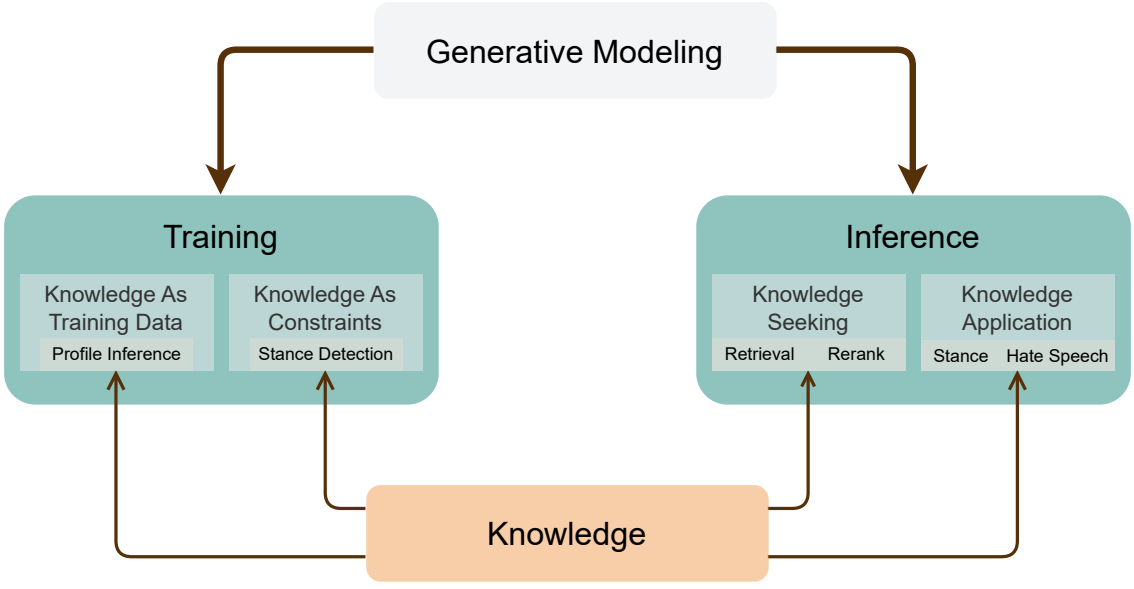


Figure 1: Overall outline of this thesis.

Part I. Knowledge-Augmented Training for Social Content Analysis with Generative Modeling. In this part, we aim to use knowledge to enable or enhance the training of generative social content analysis models. In Chapter 1, we explore open-domain Twitter user profile inference. We conduct a case study where we collect publicly available WikiData public figure profiles and use diverse WikiData predicates for profile inference. We further propose a prompt-based generation method, which can infer values that are implicitly mentioned in the Twitter information. In Chapter 2, we work on the stance detection between two entities in the context. We consider soft transitive consistency as the abstract knowledge to involve the modeling of stance correlation among inter-connected entity pairs. We propose transitive consistency constrained learning, which first finds connected entity pairs and their stances and adds an additional objective to enforce transitive consistency.

Part II. Knowledge-Augmented Inference: Knowledge-Seeking with Generative Modeling. In this part, we aim to explore the foundational paradigm for knowledge-augmented generation. In Chapter 4, we explore multimodal retrieval-augmented generation. We introduce an additional module, a multimodal reranker, to improve the ranking quality of knowledge candidates for answer generation. Our reranking module takes multimodal information from both candidates and questions and performs cross-item interaction for better relevance score modeling, and we train the reranker with distant supervision. In Chapter 3, we explore using LLMs to perform retrieval by generating representative document identifiers for domain-specific corpus. We aim to build the LLM-based generative retriever with fully synthetic data and explore factors that may affect the performance of generative retrieval, including the coverage of synthetic data, different types of identifiers, and different training stages.

Part III. Knowledge-Augmented Inference: Applications on Social Content Analysis. In this part, we aim to apply the knowledge-augmented generation paradigm to specific social content analysis tasks. In Chapter 5, we explore zero-shot and few-shot stance detection with a conditional generation framework and formulate the problem as denoising from partially filled templates. We argue that this paradigm can better utilize the semantics among input, label, and target texts. We propose several auxiliary objectives, including jointly training target prediction and incorporating manually constructed incorrect samples with unlikelihood training. More importantly, we verify the effectiveness of target-related Wikipedia knowledge with the generation framework. We also extend the detection to multimodal content, such as detecting hateful speech in memes. We argue that many multimodal contents require additional

background knowledge to resolve, such as the species on the image and their unique features. We propose retrieving relevant knowledge based on the multimodal contents and adopting the knowledge into the final prediction reasoning.

Thesis Contributions

This thesis revolves around knowledge and generative modeling for social content analysis. The contributions can be summarized as follows:

Knowledge is essential in learning social content analysis models. We verify that knowledge in different forms is essential to learn robust social content analysis models. In this thesis, we discover three types of knowledge forms: semi-structured knowledge bases (Chapter 1), abstract knowledge that can be converted into soft constraints (Chapter 2), and textual or multimodal background knowledge (Chapter 4 3 5). We demonstrate the use of semi-structured knowledge bases or abstract knowledge to enhance model training and the use of textual or multimodal background knowledge to enhance model inference.

Generative modeling is a powerful paradigm in social content analysis. In this thesis, we also demonstrate the effectiveness of using generative modeling for different social content analysis tasks, especially in few-shot or zero-shot settings, such as the open-domain Twitter profile inference(Chapter 1) and zero-shot or few-shot stance detection (Chapter 5). In addition, we show that although frozen language models are pretty robust in simple classification setups, those models still underperform when we require more structured analysis, even with in-context examples (Chapter 2 4).

Knowledge can be involved at different stages of the social content analysis model learning or inference. In this thesis, we explore multiple stages to involve knowledge and demonstrate effectiveness. In the training stage, we show that semi-structured knowledge can be transformed into distantly supervised training data (Chapter 1), and abstract knowledge can be converted into training constraints with auxiliary objectives (Chapter 2). We also show that retrieved background knowledge, such as the topic information, can be considered as an additional input for prediction generation (Chapter 5) in the inference stage.

Generative modeling helps establish a strong domain-specific retrieval pipeline. In this thesis, we also explore using generative modeling to build a domain-specific retrieval pipeline. We discuss synthetic data strategies for building domain-specific generative retrieval (Chapter 3) and building distantly supervised multimodal reranker from the visual question answering data (Chapter 4). Our results show that these methods can help build a strong domain-specific retrieval pipeline, especially with limited resources for domain-specific model training.

Part I

Knowledge-Enhanced Training for Social Content Analysis with Generative Modeling

Chapter 1

Knowledge As Training Data: Towards Open-Domain Twitter Celebrity User Profile Inference

In this chapter, we introduce the method to transform the combination of a structured knowledge base, WikiData, and Twitter data into large-scale training data for our proposed open-domain Twitter user profile inference problem. This chapter represents the first type of methods that utilize external knowledge to enhance generative modeling training for social content analysis. We also discuss the tradeoff between the extraction-based method and the generative modeling-based method for this open-domain Twitter user profile inference problem, the limitations of the generative modeling-based method with a detailed ethical statement for our data, potential benefits and risks from this work, and our efforts to mitigate the risks.

1.1 Overview

Users' profile information provides invaluable user features. Accurate automatic user profile inference is helpful for downstream applications such as personalized search [166, 184, 213, 228] and recommendations [16, 52, 120], and computational social media analysis [11, 14, 18, 178]. However, there are increasing privacy concerns that conducting profiling without appropriate regulations may reveal people's private information. Therefore, it is essential to investigate the extent of profiling to promote proper use and make the potential risks clear to the public and policy makers.

Previous work on user profile inference has focused on a very limited set of attributes, and models for different attributes employ different strategies. One line of research has formulated it as a classification problem for attributes such as gender [114, 115, 151, 158], age [32, 42, 78, 157, 161], and political polarity [4, 36, 150]. In such classification settings, each attribute has it own ontology or label set, which is difficult to generalize to other attributes, especially for attributes that have many possible candidate values (*e.g.* geo-location, occupation). In addition, some work involves human annotation, which is expensive to be acquired and may raise fairness questions for labeled individuals [86].

Another line of research uses an extraction-based method, such as graph-based [143] and unsupervised inference [59] for geolocation, distant supervision-based extraction [96, 144]. However, they still only cover limited attributes that cannot produce comprehensive profiles. Besides, many attribute values are only implicitly mentioned in Twitter context, which cannot be directly extracted.

In this chapter, instead of limited attributes, we explore whether open-domain celebrity profiles can be effectively inferred. Taking WikiData [190] as the source of profile information, which provides a much more diverse predicate set, we find WikiData profiles that have Twitter accounts. We further collect Twitter information for each account, including their recent tweets and Twitter metadata, and build models to infer profiles from collected Twitter information, which is solely based on publicly available information and does not involve any additional human annotation efforts.

We first follow Li et al. [96] to use profile information to generate distantly supervised instances and build a sequence labeling-based profile extraction model, similar to Qian et al. [144]. In order to allow open-domain inference, we propose using attribute names as prompts [92] for input sequences to capture the semantics for attribute predicates instead of involving attribute names into the tag set. However, the extraction approach requires that answers must appear in the Twitter context, which ignores some implicit text clues. Therefore, we further propose a prompt-based generation method [148] to infer user profiles, which can additionally produce values that are not straightforwardly mentioned in the Twitter information.

Our statistics show that only a limited number of WikiData attribute values can be directly extracted from Twitter information. Our experiments demonstrate a significant improvement when using the generation-based approach compared to the extraction-based approach, indicating that performing inference instead of pure extraction will be able to obtain more information from tweets. Further analysis shows that the improvement comes mainly from the power of combining extraction and inference on information not explicitly mentioned. However, we still

Predicate	Value
Entity ID	Q76
Name	Barack Obama
Country of citizenship	United States of America
Occupation	Politician
Position held	President of the United States
Work location	Washington, D.C.
Spouse	Michelle Obama
...	...

(a) WikiData information.

Barack Obama

Dad, husband, President, citizen. Washington, DC

Across the country, Americans are standing up for abortion rights—and I'm proud of everyone making their voices heard. Join a march near you:

...

Happy Mother's Day! I hope you all let the moms and mother-figures in your life know how much they mean to you. @MichelleObama, thank you for being a wonderful mother and role model to our daughters and to so many others around the world.

...

(b) Twitter information.

Figure 1.1: An example of paired WikiData and Twitter information. Relevant text spans with corresponding attribute values are highlighted with the same color.

find several challenges and limitations for the model to be applied for real-world use, including performances of low-resource attributes, distributional variances between celebrities and normal people, and spurious generation.

Our contributions are summarized as follows:

- To the best of our knowledge, this is the first work to explore open-domain Twitter user profiles.
- We create a new dataset for user profile inference from WikiData, providing with rich and accurate off-the-shelf profile information that can facilitate future social analysis research.
- We propose a prompt-based generation-based method for user profile inference that provides a unified view to infer different attributes.

1.2 Problem Definition and Dataset

In this section, we first define the open-domain user profile inference and then describe the dataset collection in detail.

1.2.1 Problem Formulation

The ultimate goal of user profile inference is to infer certain attribute value given the Twitter information of a user. In Twitter, as shown in Figure 1.1b, we mainly use the collection of recent Twitter tweets from a user u to represent Twitter information, which we denote as

$$\boldsymbol{X}_{\text{tweet, u}} = \left[\boldsymbol{x}_{\text{tweet},u,1}, \ldots, \boldsymbol{x}_{\text{tweet},u,n_{\text{tweet},u}}\right],$$

where each $\boldsymbol{x}_{\text{tweet},u,i}$ represents a sequence from a single tweet. In addition, we also concatenate the user's publicly available Twitter metadata (username, display name, bio and location) into a single sequence as complementary user information $\boldsymbol{x}_{\text{user},u}$. The final input from Twitter is the combination of user metadata and recent tweets

$$\boldsymbol{X}_u = \left[\boldsymbol{X}_{\text{tweet},u}; \left[\boldsymbol{x}_{\text{user},u}\right]\right].$$

We then assume that user profiles follow the key-value representation

$$R_{\text{u}} = \left\{(p_{u,1}, v_{u,1}), \ldots, (p_{u,n_{r,u}}, v_{u,n_{r,u}})\right\},$$

where each pair $(p_{u,i}, v_{u,i})$ represents the predicate and value of an attribute. Figure 1.1a shows an example key-value profile obtained from WikiData.

The model for open-domain user profile inference is to infer the value v of an attribute p from an user u given their Twitter information and a specific attribute predicate with parameter θ

$$f(\boldsymbol{X}_u, p; \theta) = v.$$

1.2.2 Dataset Creation

Our dataset consists of WikiData public figure profiles and corresponding Twitter information. An example of paired WikiData profile and Twitter information is shown in Figure 1.1. We first discuss the collection of WikiData profiles and then discuss the collection of Twitter information.

Category	#
# predicates	58
# average examples / predicate	12,238
# average candidates / predicate	1,179
# average tokens / answer	1.99
# tweets	13,570,664
# average words per tweet	15.3
# users (train)	106,699
# users (dev)	15,243
# users (test)	30,486

Table 1.1: Statistics of our collected data from WikiData and Twitter.

WikiData processing. WikiData is a structural knowledge base, which can be easily queried with databases such as MongoDB[1] using its dump[2]. It contains rich encyclopedia information, including information on public figures. Each WikiData entity consists of multiple properties and corresponding claims, which can be considered as the predicate value pairs as shown in Figure 1.1a[3].

First, we use WikiData to filter entities that are persons with Twitter accounts. This can be done by checking whether each entity contains the property-claim pair "instance of" (P31) "human" (Q5) and then checking whether the entity includes the property "Twitter username" (P2002). Then we extract the account of those filtered persons using the claim (value) of property "Twitter username" (P2002). If there are multiple claims, we use the first only.

Next, for each entity, we check all its properties to build the person's profile. In Figure 1.1a, as an example, we can see that the property "occupation" is "politician". For each property and claim, we only consider their text information, and we use English information only. If there are multiple claims for a property, we use the first one. We drop all properties that do not have an English name for either predicate or value, or properties that do not contain any claims.

Since WikiData profiles usually contain many noisy properties that are not suitable (*e.g.*, blood type) for Twitter user profile inference, we clean the data by 1) filtering extremely low-

[1] https://www.mongodb.com/

[2] https://dumps.wikimedia.org/wikidatawiki/entities/

[3] Please refer to https://www.mediawiki.org/wiki/Wikibase/DataModel for further details of Wikibase DataModel.

frequency properties; 2) manually selecting some meaningful and discriminative properties and 3) removing sensitive personal information listed in the Twitter Developer Agreement and Policy, such as political affiliation, ethnic group, religion, and sex or gender[4].

Twitter processing. We collect publicly available Twitter information for users that we gather from WikiData, as shown in Figure 1.1b. The Twitter information consists of the user's at most 100 recent publicly available tweets, as well as their metadata that includes username, display name, bio (a short description that a user can edit in their profile) and location. We remove all web links and hashtags from those tweets.

Category	Our Data	Li et al. (2014)	Fang et al. (2015)
# predicates	**58**	3	6
# users	**152K**	10.6K	2.5K
# values	**709K**	10.6K	15K
# tweets	13M	**39M**	846K

Table 1.2: Comparison between datasets. Our data contains a diverse set of attributes, with more users and values obtained from WikiData.

Statistics. We collect more than 168k public figures from Wikidata and filter out users whose Twitter accounts are no longer accessible. We obtain about 152K users with 13 million tweets in total. We randomly split the users into train, development, and test sets by 7:1:2. Detailed statistics are shown in Table 1.1. We compare it with previous work such as Li et al. [96] and Fang et al. [42], demonstrated in Table 1.2. We find that our dataset contains many more diverse predicates compared to Li et al. [96] and Fang et al. [42]. We also have a much larger number of users and attribute values compared to the previous work. Although Li et al. [96] contains more tweets than ours, they only consider the extraction setting, and most of the tweets in their datasets are negative samples.

Long tail distribution of predicates. As shown in Figure 1.2, the number of examples per predicate follows a long-tail distribution. Only a few predicates have many training examples,

[4] https://developer.twitter.com/en/developer-terms/agreement-and-policy

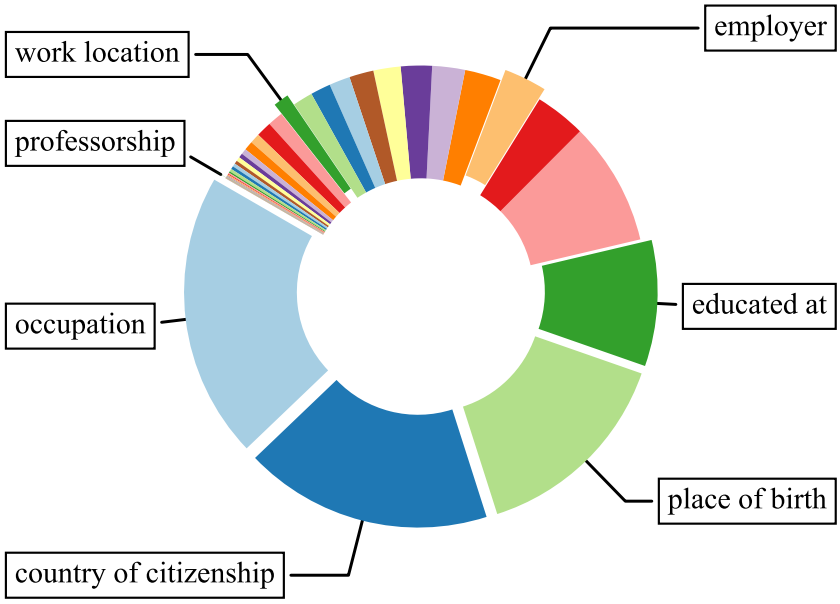


Figure 1.2: The long tail distribution of different predicates. A few predicates have many examples, while most other predicates only have limited examples.

while most appear only partially in the user's entity list. This raises a huge challenge for us to develop a good model to utilize and transfer the knowledge from rich-resource predicates to low-resource predicates. We discuss the details in the following section.

1.3 Methods

In this section, we discuss our methods for open-domain Twitter user profile inference. First, we introduce an extraction-based method that largely follows the principle from Li et al. [96] and Qian et al. [144]. Then, we discuss our proposed prompt-based generation approach that provides a unified view to infer different attribute values and can further infer values that do not appear in the Twitter context.

1.3.1 Extraction-based Method

We follow Li et al. [96] and Qian et al. [144] to generate distantly supervised training instances for user profile extraction. Since our problem is open domain, we propose using attribute predicates as prompts in input sequences and perform sequence labeling over them. This method can be divided into three steps: label generation, modeling, and result aggregation.

…	On	behalf	of	the	United
O	O	O	O	O	B
Nations	,	Secretary	-	General	…
I	O	O	O	O	O

Figure 1.3: An example tweet and tag sequence for attribute `employer` and value `United Nations`.

Label generation. Distant supervised labeling assumes that if a user u's profile contains attribute value v, we can find mentions in their Twitter information expressing the value.

Specifically, we consider each sequence $\boldsymbol{x}_i$ in $\boldsymbol{X}_u$ independently. For each attribute predicate-value pair (p_j, v_j) in u's profile, we construct a tag sequence $\boldsymbol{t}_{i,p_j}$ for $\boldsymbol{x}_i$ and the predicate p_j. For a span $[x_b, \ldots, x_e]$ that matches v_j, we make

$$t_{i,p_j,b} = \texttt{B},$$
$$t_{i,p_j,b+1} = \ldots = t_{i,p_j,e} = \texttt{I}.$$

If a position k does not match the value, then $t_{i,p_j,k} = \texttt{O}$. For simplicity, we use exact string matching between v_j and spans in the sequence. An example tag sequence is shown in Figure 1.3.

Modeling. Sequence labeling tasks usually include the label name in the tag set (*e.g.* `B-PER` for the beginning of a mention representing a person; Lample et al.,2016). In the open-domain profile inference setting, we have numerous attributes, and many of them have only a few instances, as shown in Figure 1.2, which are not sufficient to be considered as separate tag labels.

Therefore, we propose to use prompt-guided sequence labeling, where we append the attribute predicate p to the front of the sequence as the prompt as follows:

$$\texttt{[CLS]}\ p\ \texttt{[SEP]}\ \boldsymbol{x}_i$$

Then, we perform sequence labeling on the second part of the input $\boldsymbol{x}_i$ using the generated labels. We use RoBERTa [116] as the backbone encoder, and we denote the last hidden states of $\boldsymbol{x}_i$ by $\boldsymbol{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_n]$ where n represents the length of $\boldsymbol{x}_i$. The probability of predicted labels is

$$P(t_{i,p,k} \mid \boldsymbol{x}_i, p) = \text{softmax}\left(\boldsymbol{W}_h \boldsymbol{h}_k + \boldsymbol{b}_h\right) \in \mathbb{R}^3,$$

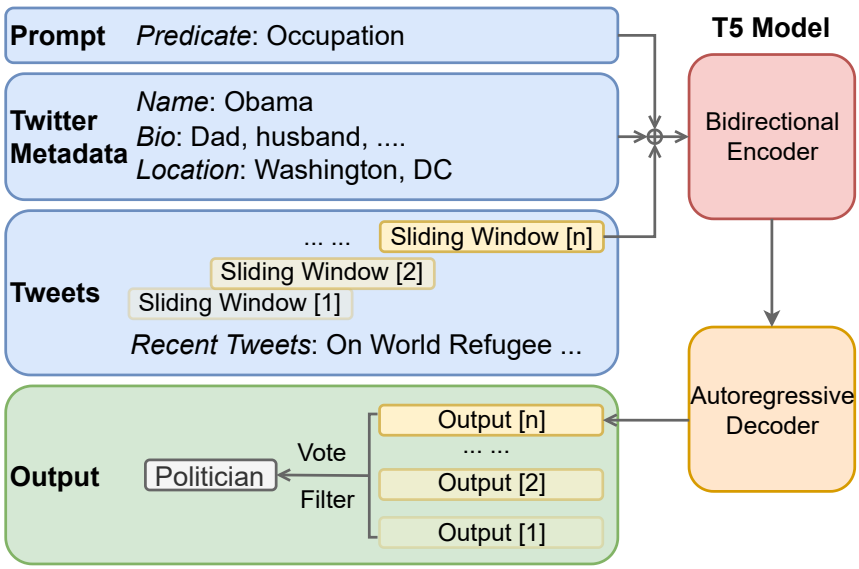


Figure 1.4: The workflow of the generation-based method, which takes the combination of predicate, Twitter metadata, and a window of tweets as input for a T5-based model, and aggregate the window-level results into user level using majority vote.

where k represent the position in $\boldsymbol{x}_i$.

During training, we randomly drop negative instances that do not contain any `B` labels to keep the positive-negative sample ratio steady.

Result aggregation. During inference, for each user, we first perform sequence labeling on every sequence predicate pair exhaustively. Then, we aggregate sequence-level labeling results into user-level results. For each attribute predicate, we select the span that has the largest averaged logit as the final answer.

1.3.2 Generation-based Method

Extraction-based methods suffer from the fact that attribute values must appear in the Twitter context. However, it is very likely that we cannot directly find those values in the context and, therefore, need to infer them using implicit evidence for profile inference. To address this issue, we propose using the conditional generation method, which is effective in both extracting input information [98, 148] and performing inference and summarization [10, 164]. The overall framework is illustrated in Figure 1.4.

Modeling. We use T5 [148], a generative transformer based model, to directly generate the answer given the predicate. Similar to the extraction-based method, to address the long-tail distribution problem, we use the attribute predicate as prompts at the beginning of the input sequence, which can capture the rich semantics of those open-domain attribute predicates, especially when the attribute predicate lacks examples in the data. Specifically, the input is the concatenation of the prefix predicate (*e.g.* `predicate:occupation`), the user's Twitter metadata, and the sequence of tweets that the user has recently published. We train the model to generate the attribute value $(y_1, \ldots, y_n)$ by minimizing the cross-entropy loss:

$$\mathcal{L}_{CE} = -\frac{1}{n}\sum_{i=1}^{n} \log p(\boldsymbol{y}_i|\boldsymbol{y}_{<i}, \boldsymbol{x}),$$

where $\boldsymbol{x}$ is the input to the model and n represents the length of the output sequence.

Since we have at most 100 recent tweets of each user whose total length normally exceeds the limit of the model, we use sliding windows and divide recent tweets organized in chronological order into different windows, where each window can represent information within a time range. Then, we train the model on these divided examples separately. Each example contains the same prefix predicate and Twitter metadata but uses different parts of the tweets to infer the attribute value.

Result aggregation. During inference, we use the same sliding window strategy and divide the input into different examples to make predictions independently. Then, similar to the extraction-based method, we aggregate those window-level predictions into a user-level prediction. We count the occurrences of each predicted text for a predicate and then use a majority vote to find the aggregated result of that predicate.

Result filtering. The generation-based method aggressively generates output without estimating whether the generated output is spurious. Therefore, it is important to filter those incorrect predictions during inference.

After result aggregation, we first take the product of probability for each generated token as the score for each aggregated prediction and then use the averaged score over all aggregated predictions as the confidence score for the aggregated result. A low confidence score indicates that the model cannot determine whether the prediction is valid.

For each predicate, we search the best threshold and set predictions with confidence scores lower than the threshold as "no prediction". We consider all predicted confidence scores from

the development set as candidate thresholds and choose the threshold that yields the best performance on the development set. The best-searched threshold is then directly applied to filter results on the test set.

1.4 Experiments

In this section, we conduct experiments on our constructed dataset and user profile extraction dataset [96]. Then, we provide a qualitative analysis and discuss the remaining challenges.

1.4.1 Experimental Setup

We use `roberta-base`[5] as the base model for the extraction-based model, as it demonstrates its effectiveness on multiple sequence labeling tasks. We use `t5-small`[6] for the generation-based model, which has much fewer parameters than `roberta-base`. We use two Nvidia GeForce RTX 3090 GPUs as our computing infrastructure.

Extraction-based method setup. We finetune the model with 10 epochs using AdamW. The learning rate is 5e-5 using the linear scheduler without warmup. The batch size is 128. The hidden size for classification is 768. The positive-negative sample ratio is 1:5. We use tag-level F_1 as Qian et al. [144] to efficiently select the best results on the development set for a single run. The training time is about 16 hours, and inference on the test set is about 5 hours.

Generation-based method setup. We fine-tune the model on all sliding window examples for 5 epochs using AdamW. The learning rate is 1e-4, using a linear scheduler with no warmup. The batch size is 96. We use gradient clipping with max norm 3 to increase stability during training. We use sliding windows with size 512 and stride 128. We use a greedy search during inference. We use the exact match to select the best results on the development set efficiently for a single run. The training time is about 40 hours, and inference on the test set is about 3 hours.

Evaluation metric. We choose the user-level F_1 as our evaluation metric. Specifically, we suppose that a user profile consists of n different attributes. We use $C(\cdot)$ to represent the

[5] https://huggingface.co/roberta-base
[6] https://huggingface.co/t5-small

count of different types of output. $C(\text{no prediction})$ refers to the count of "no predictions" and $C(\text{correct prediction})$ refers to the count of predictions that match the WikiData profile. Then we obtain the user-level F_1 as follows:

$$\text{precision} = \frac{C(\text{correct prediction})}{n - C(\text{no prediction})}$$
$$\text{recall} = \frac{C(\text{correct prediction})}{n}$$
$$F_1 = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

We consider the prediction to be valid when it identically matches the ground truth. We do not use entity-level or tag-level F_1 as Qian et al. [144] because it is not applicable to the generation model. We do not use generation-based metrics (*e.g.*, BLEU) because we observe that most predictions are very short. In addition, compared to no prediction, we want to penalize wrong predictions more. In F_1, the basis of precision does not include "no prediction" results from models while it still has a penalty for wrong predictions.

1.4.2 Results

Results on User Profile Inference

	Development Set			Test Set		
Model	Precision	Recall	F_1	Precision	Recall	F_1
Random	0.22	0.22	0.22	0.23	0.23	0.23
Majority	14.56	14.56	14.56	14.19	14.19	14.19
Extraction	18.36	9.69	12.69	18.39	9.80	12.79
Generation	**59.05**	**43.71**	**50.23**	**58.73**	**43.40**	**49.92**

Table 1.3: System performance (%) on our constructed open-domain Twitter user profile inference dataset.

The main results are shown in Table 1.3. The random result means that predictions are uniformly randomly selected, and the majority result means that predictions are selected with the values that occur most frequently in the training set. We find that both simple methods perform poorly. In general, we find that our generation-based method significantly outperforms other

methods by a large margin. We also find that the extraction-based method cannot outperform the majority baseline. The reason is that the majority vote can achieve relatively high accuracy on attributes that have a relatively small number of candidates, or one specific candidate takes a large portion of the data, while we cannot find corresponding occurrences of some of those attributes in the Twitter context.

Model	Precision	Recall	F_1
Random	0.26	0.26	0.26
Majority	4.77	4.77	4.77
Extraction	72.14	**71.47**	71.80
Generation	**77.64**	68.60	**72.84**

Table 1.4: System performance (%) on the subset of the test set that we can find occurrences of attribute values in Twitter context.

To verify the above claim, we perform another test on a subset of the test set data, for which we can find corresponding occurrences of attribute values in the Twitter context. We find that only 13.56% of the test data can find those value occurrences, which indicates that the majority of the data cannot be directly extracted from the Twitter context. The results are shown in Table 1.4. By comparing the results with overall results, we can find that both extraction and generation systems can get better performance on the subset where we can find occurrences of attribute values. We find that the extraction method performs quite closely to the generation-based method in this setting, though the generation-based method performs better on precision and F_1 and the extraction-based method better on recall. This result indicates that when attribute values occur in the Twitter context, the extraction model can effectively extract them, while the generation-based method can additionally infer values that are not included in the Twitter content.

Results on User Profile Extraction

We conduct additional experiments on the profile extraction dataset from Li et al. [96], where we can provide a direct comparison between our generation-based model and previous work. We follow the same preprocessing as Qian et al. [144] on Education and Job. We make two changes to our generation-based model for this dataset. 1) This dataset does not contain a timestamp for each tweet, so we use each tweet as an independent sample instead of the sliding window

Category	Education			Job		
	Precision	Recall	F_1	Precision	Recall	F_1
GraphIE	92.87	79.74	85.77	76.03	61.01	67.66
Generation	**94.28**	**91.40**	**92.82**	**78.97**	**65.78**	**71.76**

Table 1.5: Results on Li et al. [96] following the preprocessing as Qian et al. [144]. We re-evaluate the results based on user-level F_1. $p < 0.01$ for both F_1 comparisons.

Model	Precision	Recall	F_1
Our model	**59.05**	43.71	**50.23**
-threshold	45.95	**45.95**	45.95
-aggregation	57.39	43.35	49.39
-metadata	53.59	40.45	46.10

Table 1.6: Effects (%) of result filtering (-threshold), result aggregation (-aggregation) and Twitter metadata on development set. $p < 0.01$ for F_1 comparisons.

strategy. 2) This dataset is designed for extraction, so for tweets from which the answers cannot be extracted, we train the generation model to output "no prediction".

The experiment results are shown in Table 1.5. We compare with GraphIE [144], one of the state-of-the-art models on this dataset. We reproduce the results from their script[7] and re-evaluate on user-level with the majority vote. We use the averaged results over 5-fold cross-validation as Qian et al. [144]. The results show that our model can significantly outperform GraphIE on both Education and Job attributes, which indicates that even if the attributes are limited, the generation-based method can still achieve promising performance.

1.4.3 Ablation study

We conduct an ablation study on two of our components, result filtering and result aggregation, on our profile inference data, as shown in Table 1.6. We find that result filtering can successfully filter spurious results by improving over 13% on precision while only dropping about 2% on recall. We also find that result aggregation improves both precision and recall, indicating that

[7] https://github.com/thomas0809/GraphIE

... One of the proudest moments of my career being the flag bearer at the Olympics for my home country of Denmark! ...		
Attribute	**Value**	✔
country of citizenship	Denmark	
... It is going to be February 9, 2022 in Royal Arena against my great friend! ... Beach bod/Mom bod Mommy daughter pool time 2 months with our little angel she clearly enjoyed her first tennis lesson ...		
Attribute	**Value**	✔
occupation	tennis player	
... bio: Member of the European Parliament ... Still unclear about strategic autonomy. We can't flip a coin when deciding about 2% GDP for Need a clear mechanism for EU intervention. My view in on needs to get a chance to win 5G race...		
Attribute	**Value**	✔
occupation	politician	
... Can't wait this match vs Brock! Wow...Amazing match Undertaker def 21-0 ... Thanks for a great show. And I CAN WRESTLE ...		
Attribute	**Value**	✘
occupation	professional wrestler	actor

Figure 1.5: Example window-level predictions from generation-based model with their context.

we can obtain better inference by using a larger Twitter context. Twitter metadata also provides rich information about the user's background. We train and evaluate another model without Twitter metadata and find that we see a significant performance drop. However, we still find that many attributes inferred by the model are not dependent on those metadata.

1.4.4 Qualitative Analysis

Figure 1.5 demonstrates four window-level predictions from the generation-based model with relevant input context. The first case shows that the model can directly copy relevant information from the context. The second and third cases show that the model can infer the information based on the context. The last case shows an error that the model does not fully utilize the in-

formation provided by "wrestle" and generates incorrect information, possibly affected by the other word "show". This case indicates the importance of background information for a specific attribute value.

1.4.5 Remaining Challenges

Although achieving improvement on open-domain attribute inference, we still find that the model's performance on attributes with low training samples is generally much lower than on attributes with rich samples. It is still under investigation for a better generalization of these low-resource attributes.

WikiData provides rich profiles for many Twitter users. However, the distribution of these Twitter users with WikiData profiles may not align with the need for downstream tasks. For example, most people with WikiData profiles are celebrities, such as politicians and athletes, and it lacks information on general occupations, such as farm workers.

The granularity of the prediction results is also another important direction to investigate. We observe in some cases that the prediction and the grountruth are in different levels of granularity. For example, the groundtruth can be "Tokyo" while the prediction may be "Japan". Therefore, it is also important to address this issue with both better modeling as well as evaluation.

We consider that the model can predict all collected attribute values because we have manually selected meaningful and discriminative properties from WikiData during dataset construction. However, it is still possible that a specific property value cannot be detected well based on Twitter content, leading to spurious generation output. For example, if a user is a medical doctor but did not discuss any medical information on Twitter, the occupation is very hard to predict. It is still important to further investigate this "cannot predict" cases in both dataset construction and model design.

1.4.6 Limitations

Besides the technical challenges discussed in Section 1.4.4-1.4.5, limitations of this work also include the issue of data imbalances that some attributes may have imbalanced distributions. For example, we may find significantly more profiles with the country of citizenship as the United States than any other country, which may have a negative impact on generalization, especially when the distributions of training and inference diverge. Similarly, the distributional

variances discussed in Section 1.4.5 indicate that the prediction results for non-celebrity distributions should be carefully adjudicated. The degraded performance on low-resource attributes also indicates that the prediction results may be unreliable when inference is made on attributes without enough training data.

In this chapter, we assume that the attributes are already given. However, many WikiData attributes are not applicable to everyone. For example, attributes such as "position played on team" may be specific to athletes. Therefore, it is also important to investigate how to automatically detect applicable attributes for certain users.

In this work, we use at most 100 recent tweets and aggressively create training and inference examples between each attribute and those tweets. Since we use sliding windows on the collected tweets, involving more tweets in training or inference may significantly increase the time cost.

1.5 Related Work

User Profile Inference. One line of user modeling research focuses on profile inference or extraction. Previous work on user profile inference focuses on some specific attributes such as gender [114, 115, 151, 158], age [32, 42, 78, 157, 161], and political polarity [4, 36, 150]. They often consider them as multi-class classification problems. Most of these methods use the context of those social media posts. Alternatively, user name and profile on social media [114, 115], part-of-speech and dependency features [157], user social circles [32] and photos [42] have been explored as additional important features for different attribute inference. But those classification settings have a pre-defined ontology or label set, which is difficult to extend to other attributes.

In addition to classification-based methods, there are also graph-based [143], distant supervision-based and unsupervised extraction [59]. Compared to the classification method, extraction-based methods are capable of identifying attributes with a large ontology. But they rely on entities from the context as candidates, which limits the scope of the attributes that occur frequently in the social media context.

Our open-domain Twitter user profile inference uses a larger predicate set and data than previous work. We further propose the generation-based approach, which addresses the limited scope.

Another line of user modeling research focuses on leveraging behavior signals [2, 80] or building implicit user representations [61, 62], which is more distantly related to our problem.

Sociolinguistic variation. The intuition of inferring user attributes from their posts aligns with sociolinguistic variation in which people investigate whether a linguistic variation can be attributed to different social variables [84]. Computational efforts to discover these relationships include demographic dialectal variation [22], geographical variation [41, 133], syntactic or stylistic variation over age and gender [72], socio-economic status [20, 44].

1.6 Conclusion

In this chapter, we first explore open-domain Twitter user profile inference. We use the combination of WikiData and Twitter information to create a large-scale dataset. We propose to use a generation-based method with attributes as prompts and compare it with the extraction-based method. The result shows that the generation-based method can significantly outperform the extraction-based method on open-domain profile inference, with the ability to perform both direct extraction and indirect inference. Our further analysis still finds some of the errors and remaining challenges of the generation-based method, such as degraded performances for low-resource attributes and spurious generation, which reveals the limits of our current generation-based user profile inference model.

1.7 Attribute Descriptions

We provide the descriptions of each attribute from Wikidata in Table 1.7 to facilitate the understanding of attributes and mitigate the potential impact from dataset biases.

ID	Attribute	Description
P106	occupation	occupation of a person; see also "field of work" (Property:P101), "position held" (Property:P39)
P27	country of citizenship	the object is a country that recognizes the subject as its citizen
P19	place of birth	most specific known (e.g. city instead of country, or hospital instead of city) birth location of a person, animal or fictional character
P69	educated at	educational institution attended by subject
P1412	languages spoken, written or signed	language(s) that a person or a people speaks, writes or signs, including the native language(s)
P641	sport	sport that the subject participates or participated in or is associated with
P108	employer	person or organization for which the subject works or worked
P39	position held	subject currently or formerly holds the object position or public office
P1303	instrument	musical instrument that a person plays or teaches or used in a music occupation
P54	member of sports team	sports teams or clubs that the subject represents or represented
P166	award received	award or recognition received by a person, organisation or creative work
P413	position played on team / speciality	position or specialism of a player on a team
P551	residence	the place where the person is or has been, resident
P1344	participant in	event in which a person or organization was/is a participant; inverse of P710 or P1923
P103	native language	language or languages a person has learned from early childhood
P937	work location	location where persons or organisations were actively participating in employment, business or other work
P3602	candidacy in election	election where the subject is a candidate

Continue on the next page

Table 1.7: Attribute Description

ID	Attribute	Description
P463	member of	organization, club or musical group to which the subject belongs. Do not use for membership in ethnic or social groups, nor for holding a political position, such as a member of parliament (use P39 for that).
P101	field of work	specialization of a person, organization, or of the work created by such a specialist; see P106 for the occupation
P118	league	league in which team or player plays or has played in
P2094	competition class	official classification by a regulating body under which the subject (events, teams, participants, or equipment) qualifies for inclusion
P512	academic degree	academic degree that the person holds
P2416	sports discipline competed in	discipline an athlete competed in within a sport
P1411	nominated for	award nomination received by a person, organisation or creative work (inspired from "award received" (Property:P166))
P361	part of	object of which the subject is a part (if this subject is already part of object A which is a part of object B, then please only make the subject part of object A). Inverse property of "has part" (P527, see also "has parts of the class" (P2670)).
P6886	writing language	language in which the writer has written their work
P6553	personal pronoun	personal pronoun(s) this person goes by
P241	military branch	branch to which this military unit, award, office, or person belongs, e.g. Royal Navy
P410	military rank	military rank achieved by a person (should usually have a "start time" qualifier), or military rank associated with a position
P2348	time period	time period (historic period or era, sports season, theatre season, legislative period etc.) in which the subject occurred

Continue on the next page

Table 1.7: Attribute Description

ID	Attribute	Description
P710	participant	person, group of people or organization (object) that actively takes/took part in an event or process (subject). Preferably qualify with "object has role" (P3831). Use P1923 for participants that are teams.
P1576	lifestyle	typical way of life of an individual, group, or culture
P2650	interested in	item of special or vested interest to this person or organisation
P740	location of formation	location where a group or organization was formed
P859	sponsor	organization or individual that sponsors this item
P812	academic major	major someone studied at college/university
P8413	academic appointment	this person has been appointed to a role within the given higher education institution or department; distinct from employment or affiliation
P5096	member of the crew of	person who has been a member of a crew associated with the vessel or spacecraft. For spacecraft, inverse of crew member (P1029), backup or reserve team or crew (P3015)
P803	professorship	professorship position held by this academic person
P66	ancestral home	place of origin for ancestors of subject
P112	founded by	founder or co-founder of this organization, religion or place
P3828	wears	clothing or accessory worn on subject's body
P1321	place of origin (Switzerland)	lieu d'origine/Heimatort/luogo d'origine of a Swiss national. Not be confused with place of birth or place of residence
P495	country of origin	country of origin of this item (creative work, food, phrase, product, etc.)
P276	location	location of the object, structure or event. In the case of an administrative entity as containing item use P131. For statistical entities use P8138. In the case of a geographic entity use P706. Use P7153 for locations associated with the object.
P5389	permanent resident of	country or region where a person has the legal status of permanent resident

Continue on the next page

Table 1.7: Attribute Description

ID	Attribute	Description
P1429	has pet	pet that a person owns
P263	official residence	the residence at which heads of government and other senior figures officially reside
P1268	represents	organization, individual, or concept that an entity represents
P3716	social classification	social class as recognized in traditional or state law
P17	country	sovereign state of this item (not to be used for human beings)
P488	chairperson	presiding member of an organization, group or body
P7779	military unit	smallest military unit that a person is/was in
P1716	brand	commercial brand associated with the item
P6	head of government	head of the executive power of this town, city, municipality, state, country, or other governmental body
P159	headquarters location	city, where an organization's headquarters is or has been situated. Use P276 qualifier for specific building
P8047	country of registry	country where a ship is or has been registered

Table 1.7: Attribute Description

Chapter 2

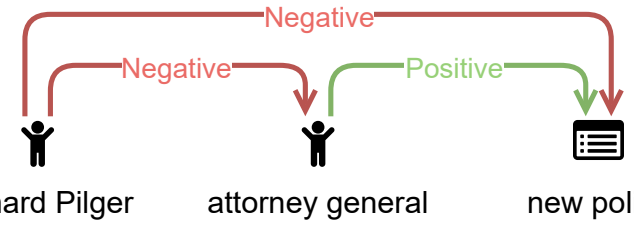
Knowledge As Constraints: Transitive Consistency Constrained Learning for Entity-to-Entity Stance Detection

In this chapter, we discuss converting abstract prior knowledge into a series of constraints and using those constraints during training to enhance entity-to-entity stance detection. This chapter represents another type of knowledge-enhanced training methods, which focus on abstract knowledge and consider constraints when optimizing the model. We explore the options of adding the constraints into classification and generation-based methods and investigate the appropriate setup to incorporate the constraints.

2.1 Overview

Detecting polarity from text has been widely studied in different forms, such as sentence-level [136] or aspect-level sentiment analysis [141], target-oriented stance detection [58, 171], and structured analysis [19, 77, 203].

Some recent efforts explore a streamlined and informative form, entity-to-entity stance detection [139, 222], which identifies the stance between a pair of entities with a directed link that indicates source, target, and polarity. Entity-to-entity stance detection can be used to analyze more objective contexts such as news articles in an effective way without the extraction of complex dependency structure with opinion expressions [19, 77, 203], especially compared to most previous work that usually assumes opinions come from the author [58, 130, 136, 141, 171].



(e1, Richard Pilger), director of the elections crimes branch in the Justice Department's Public Integrity Section, told colleagues in an email that the **(e2, attorney general)** was issuing "an important **(e3, new policy)** abrogating the forty-year-old Non-Interference Policy for ballot fraud investigations in the period prior to elections becoming certified and uncontested".

Figure 2.1: An example of three entity-to-entity stances and their consistency. If we know "Richard Pilger" was against the "attorney general", and the "attorney general" supported the "new policy", we may infer that "Richard Pilger" was also likely against the "new policy".

The input of a typical entity-to-entity stance detection system consists of a context and a pair of entities and finds a directed link between them, as shown in Figure 2.1. Previous efforts [139, 222] optimize model training on each entity pair individually. However, the stances of inter-connected entity pairs may be correlated. As we can find in Figure 2.1, if we know "Richard Pilger" was against the "attorney general" (Negative), while the attorney general supported the "new policy" (Positive), we may infer that "Richard Pilger" was also against the new policy with these two known stances. We hypothesize that this type of transitive correlation is common in political news and can be used effectively to train better models [170].

In this chapter, we consider the correlation between inter-connected entity pairs as transitive consistency constraints during training and use these constraints to help learn entity-to-entity stance detection models. Specifically, we first sample a pair of sentences that share a common entity. Based on the intra-sentence entity-to-entity stances, we may infer the stance of the entity pair across the two sentences with transitivity. The inferred stance is expected to be softly aligned with the stance detection prediction on the entity pair directly. Therefore, given the two intra-sentence stance predictions and the cross-sentence stance prediction, we can add additional soft consistency loss between the triple terms to enforce the similarity. In this work, we develop two typical methods for entity-to-entity stance detection and try to combine the transitive consistency constraints with them. One is based on relation classification from en-

tity pair representations [40, 192, 193]. The other method is based on recent trends in language model instruction tuning [33, 132, 160, 198], where we generate the stance autoregressive.

We conduct our experiments on DSE [139] and SEESAW [222], both of which analyze stances in political news. DSE requires models to identify the neutral label, and the label direction. SEESAW is originally designed to jointly generate an entity pair and the corresponding polarity, so it does not provide the mention-level entity annotation and the neutral label. Our experiment results show that the transitive consistency constraints help in learning better classification and generation models, which also implies the prevalence of stance transitivity on political news. We further show that the performance is sensitive to the degree of applying constraints, and there is a performance degradation if we overstrictly enforce the constraints. In addition, we find that large language models with in-context learning [132, 186] cannot obtain reliable performance on DSE. Our further analysis shows that it is non-trivial to directly use large language models with in-context learning on the neutral label or directed label predictions.

2.2 Entity-to-Entity Stance Detection Frameworks

Entity-to-entity stance detection identifies the stance between a pair of entities, as well as the source and target through a directed link. In this section, we introduce two basic frameworks for this challenge. One is based on relation classification using entity pair representations, while the other is to generate the stance autoregressively.

2.2.1 Classification-Based Framework

Classification-based framework obtains entity-pair representation from the input sentence and performs classification using the obtained pairwise representation. This paradigm has shown effectiveness in various relation extraction tasks [40, 192, 193, 201].

Specifically, the model takes a sequence of tokens $\boldsymbol{x}$ with length n as input, representing the input sentence. The input also includes the positions of two entity mentions (e_1, e_2) in the text. We denote the groundtruth stance as $s(e_1, e_2)$. In this task, we only consider the position of the first token of the corresponding entity and denote the positions of the entity pair (e_1, e_2) as (p_1, p_2).

The entity-to-entity stance detection model predicts the stance between the given two entity mentions. It first uses a pretrained language model (38, 116, PLM) to obtain the contextualized

representation of the input sequence,

$$\boldsymbol{H} = \text{PLM}(\boldsymbol{x}),$$

where $\boldsymbol{H}$ represents the contextualized representation of the sequence and $\boldsymbol{h}_i$ is the representation of the token at position i.

We obtain the entity-pair representation by concatenating the representation of the given position pair

$$\boldsymbol{c} = [\boldsymbol{h}_{p_1}; \boldsymbol{h}_{p_2}] .$$

Then, entity-pair representation is used for classification with a two-layer feed-forward neural network (FFN) and a softmax layer to predict the entity-to-entity stance label s

$$p(s \mid e_1, e_2) = \text{softmax}(\boldsymbol{a}),$$
$$\boldsymbol{a} = \text{FFN}_2(\tanh(\text{FFN}_1(\boldsymbol{c}))),$$

where $\text{FFN}_i(\boldsymbol{h}) = \boldsymbol{W}_i\boldsymbol{h} + \boldsymbol{b}_i$.

For the stance detection task that requires models to detect both polarity and direction [139], each classification label is the combination of direction and polarity. Therefore, the stance label is related to the input order of the entity pair representation. For example, the label can be "`Entity 1 to Entity 2 positive`" or "`Entity 2 to Entity 1 negative`". "`Entity 1`" represents the first entity of the concatenated entity pair representation, while "`Entity 2`" represents the second entity. Therefore, "`Entity 1 to Entity 2`" indicates that the first entity is the source entity while the second entity is the target entity. The only exception is the neutral label, which is undirected in nature, which represents that there is no explicit stance polarity between the two entities.

The model is trained by minimizing the cross-entropy loss

$$\mathcal{L}_s = -\sum \mathbb{I}_{s(e_1,e_2)=s_i} \log p(s = s_i \mid e_1, e_2).$$

2.2.2 Generation-Based Framework

Generation-based methods have also shown strong performance on various tasks, especially for tasks that are not traditionally modeled with generative methods [93, 98, 148, 202, 210]. Recently, a line of research utilizes conditional language models to perform relation extraction and achieves promising performance [60, 121, 138, 191]. Therefore, we also use the generation-based method on our entity-to-entity stance detection experiments.

Specifically, our generation-based model is trained on decoder-only language models [132, 145], which takes input tokens and generates new tokens autoregressively using one Transformer [188]

$$p\left(\boldsymbol{o} \mid \boldsymbol{x}, e_1, e_2\right) = \prod_{i=1}^{|\boldsymbol{o}|} p\left(o_i \mid \boldsymbol{o}_{<i}; \mathrm{T}(\boldsymbol{x}, e_1, e_2)\right),$$

where $\boldsymbol{x}$ is the input sentence, and $\mathrm{T}(\boldsymbol{x}, e_1, e_2)$ produces a combination of short instruction, input sentence, and entity pairs into a single sequence with a template. In our entity-to-entity stance detection task, we define the template as:

"`Analyze the entity-entity stance in the following text:\n` $\boldsymbol{x}$ `\nEntity 1:` e_1 `\nEntity 2:` e_2 `\nStance:`".

The model takes $\mathrm{T}(\boldsymbol{x}, e_1, e_2)$ and produces a series of tokens $\boldsymbol{o}$ as the output of the entity-to-entity stance detection. Similar to the classification-based method, we need to combine direction and polarity into the output of the generation when performing a directed stance detection. We first output the stance direction, and then output the stance polarity. We use the text `Entity 1 to Entity 2` and `Entity 2 to Entity 1` to represent two directions, and `positive`, `negative`, and `neutral` as polarity words. The output text of a neutral label does not include a direction phrase as it is undirected.

The model is trained by minimizing the log-likelihood over the generated output sequence:

$$\begin{aligned}\mathcal{L}_s &= -\log p\left(\boldsymbol{o} \mid \boldsymbol{x}, e_1, e_2\right) \\ &= -\sum_{i=1}^{|\boldsymbol{O}|} \log p\left(o_i \mid \boldsymbol{o}_{<i}; \mathrm{T}(\boldsymbol{x}, e_1, e_2)\right).\end{aligned}$$

2.3 Transitive Consistency Constrained Learning

Inter-connected stances may be correlated, especially in political news, as shown in Figure 2.1. We hope to capture the correlation from optimizing the predicted stances that can be inferred from the transitivity of existing stances. In this section, we will introduce the concept of transitive stance inference.

The transitive stance inference requires multiple inter-correlated entity pairs in a context, while most existing resources only annotate one entity-to-entity stance at the sentence level. Therefore, we propose a simple sentence-pair sampling method that helps obtain data for transitive inference. Then, we introduce the constrained learning method, which can be added to

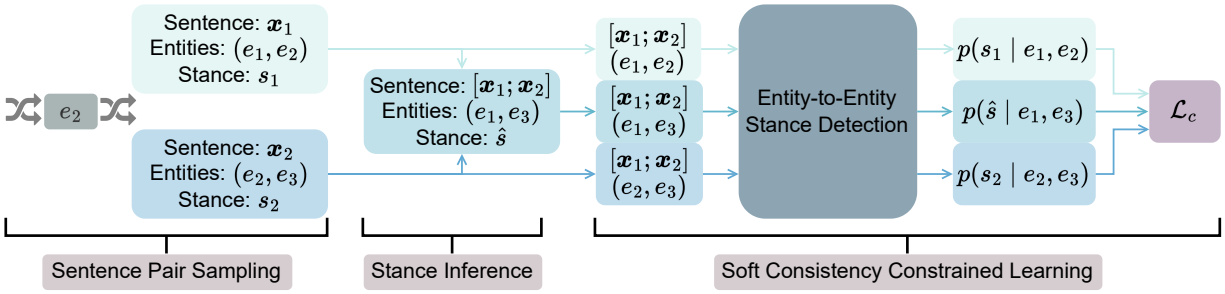


Figure 2.2: The overall framework of soft consistency constrained learning objective. We first sample an entity as the shared entity and use this entity to sample two sentences that can be used for stance inference. We then concatenate the two sentences, with combinations of entity pairs from the sampled three entities for entity-to-entity stance detection, and the objective is the penalty for inconsistent predictions.

		$e_2 \longrightarrow e_3$		$e_2 \longleftarrow e_3$	
		Positive	Negative	Positive	Negative
$e_1 \longrightarrow e_2$	Positive	$e_1 \xrightarrow{\text{Positive}} e_3$	$e_1 \xrightarrow{\text{Negative}} e_3$	-	-
	Negative	$e_1 \xrightarrow{\text{Negative}} e_3$	$e_1 \xrightarrow{\text{Positive}} e_3$	-	-
$e_1 \longleftarrow e_2$	Positive	-	-	$e_1 \xleftarrow{\text{Positive}} e_3$	$e_1 \xleftarrow{\text{Negative}} e_3$
	Negative	-	-	$e_1 \xleftarrow{\text{Negative}} e_3$	$e_1 \xleftarrow{\text{Positive}} e_3$

Table 2.1: The transitive mapping of a pair of directed stances with a shared entity. "-" denotes no mapping between the pair of stances. We also do not apply transitive mapping for neutral samples.

both classification-based and generation-based methods to capture transitive correlation. The overall framework is illustrated in Figure 2.2.

2.3.1 Transitive Stance Inference

Suppose we have three entities (e_1, e_2, e_3), and we know the directed entity-to-entity stance $s(e_1, e_2)$, $s(e_2, e_3)$, the stance inference is to infer the stance from the two known stances

$$\hat{s}(e_1, e_3) = s(e_1, e_2) \oplus s(e_2, e_3).$$

The stance inference can be divided into two steps. The first step is to check whether e_1 can reach e_3 ($e_1 \rightarrow e_2 \rightarrow e_3$) or e_3 can reach e_1 ($e_1 \leftarrow e_2 \leftarrow e_3$) using existing directed links, which

is a prerequisite for transitivity. If e_1 can reach e_3, we will be able to infer the stance from e_1 (as the source) towards e_3 (as the target), and vice versa. For other cases ($e_1 \rightarrow e_2 \leftarrow e_3$, $e_1 \leftarrow e_2 \rightarrow e_3$), we will not be able to apply the transitive inference. We also do not use stances with neutral labels in our stance inference, as they are undirected.

The second step is to determine the stance polarity. We formulate the stance polarity mapping similar to the logical non-equivalence (XOR) operator, and we denote the mapping operator by $\oplus$. If both polarities of the two known stances $s(e_1, e_2)$, $s(e_2, e_3)$ are positive or negative, the inferred polarity of the stance $\hat{s}(e_1, e_3)$ will be positive. If among the two known stances, one is positive and the other is negative, the inferred polarity of the stance $\hat{s}(e_1, e_3)$ will be negative.

Combining these two steps, we have a complete stance inference from transitive mapping, which is illustrated in Table 2.1.

2.3.2 Two-Step Sentence Pair Sampling

Existing resources (e.g., 139) mostly focus on sentence-level annotation. For each sentence, they pick one pair of entities and annotate the directed stance between them. However, as we introduced in Section 2.3.1, to infer the stance with transitivity, we will need a pair of stances of which two entity pairs share one entity and there are in total three entities. Therefore, we propose a simple sentence-pair sampling method in the training data using a two-step sampling to obtain these samples.

Specifically, we first uniformly sample an entity as the shared entity. Uniform sampling over entities is to ensure that a few frequently occurring entities will not have a substantially high probability of being sampled. Then, we can find all sentences with entity-to-entity stance annotations involving the given entity, and we uniformly sample a pair of sentences among them. The sentence pair will also provide us with a pair of entity-to-entity stance annotations that share a common entity. We will disregard the sampled sentence pair if the entity pairs from the sentence pair are the same, or if the sampled entity-to-entity stance pair does not constitute the case in which we can apply the transitive mapping. We keep performing the two-step sampling until we find a valid sentence pair.

2.3.3 Soft Consistency Constrained Learning

The overall idea of constrained learning is to add an additional penalty if the predicted label does not match the inferred label [192]. We first use the classification-based method to explain

our proposed method and naturally extend it to the generation-based method.

For a given sampled sentence pair $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ with entity pair $(e_{1,1}, e_{1,2})$ and $(e_{2,1}, e_{2,2})$ correspondingly, we perform normalization on the sentence pair annotation first to ensure $e_{1,2} = e_{2,1}$ as the shared entity. This normalization involves flipping the stance direction with the entity order within the input entity pair. For example, if there is a stance $s(e_1, e_2)$ interpreted as `Entity 1 to Entity 2 positive`, after flipping the input order of the entity pair from (e_1, e_2) to (e_2, e_1), the corresponding flipped stance $s(e_2, e_1)$ will be `Entity 2 to Entity 1 positive`. Therefore, we flip the label of the first sentence if the shared entity is $e_{1,1}$ in the original annotation and flip the label of the second sentence if the shared entity is $e_{2,2}$. For simplicity, we assume that the input of the following discussion is already normalized.

We use concatenated input from the sentence pair $[\boldsymbol{x}_1; \boldsymbol{x}_2]$ with three entity pairs $(e_{1,1}, e_{1,2})$, $(e_{2,1}, e_{2,2})$ and $(e_{1,1}, e_{2,2})$, which represents two intra-sentence entity pairs with groundtruth stance annotation, and one inter-sentence entity pair with inferred stance. These inputs will be fed into the classification-based method and obtain three distributions, $p(s \mid e_{1,1}, e_{1,2})$, $p(s \mid e_{2,1}, e_{2,2})$ and $p(s \mid e_{1,1}, e_{2,2})$. The objective is to promote similarity between $p(s \mid e_{1,1}, e_{1,2}) \times p(s \mid e_{2,1}, e_{2,2})$ and $p(s \mid e_{1,1}, e_{2,2})$, where the former term can be considered as the probability of applying stance inference, while the latter term is the probability of direct stance detection. We use the groundtruth and inferred labels with L_1 distance as this objective

$$\begin{aligned}\mathcal{L}_c = |&\log p\left(s = s\left(e_{1,1}, e_{1,2}\right) \mid e_{1,1}, e_{1,2}\right) \\ &+ \log p\left(s = s\left(e_{2,1}, e_{2,2}\right) \mid e_{2,1}, e_{2,2}\right) \\ &- \log p\left(s = \hat{s}\left(e_{1,1}, e_{2,2}\right) \mid e_{1,1}, e_{2,2}\right)|.\end{aligned}$$

We jointly train the consistency constrained objective with the regular single-sentence learning objective $\mathcal{L}_s$ (cross-entropy for classification and sequence log-likelihood for generation)

$$\mathcal{L} = \mathcal{L}_s + \lambda \mathcal{L}_c,$$

where the factor λ is to control the degree of enforcing the consistency objective.

Extending to generation-based method. When extending the consistency constrained learning to the generation-based method, we need to find a legitimate estimate from the generation framework to represent the log probability of the stance label. For simplicity, we directly choose the sum of the log probability of the predicted polarity word and two entity numbers to represent the log probability, as they are the most important factors of an entity-to-entity stance label.

2.4 Experiments

2.4.1 Data

Category	DSE	SEESAW
# Label Types	5	2
Stance Direction	In Labels	Part of Input
Neutral Label	Yes	No
Entity Position	Yes	No
Data Statistics		
# Train	13,144	6,263
# Valid	1,461	2,436
# Test	1,623	1,920

Table 2.2: Comparison of DSE and SEESAW datasets.

We conduct experiments on two datasets, DSE [139] and SEESAW [222]. DSE requires the model to predict both the stance direction and the polarity with entity mentions and their positions in the context. The annotation is always from the first mentioned entity to the second entity in the context.

SEESAW was originally designed to jointly generate pairs of entities with their stances, and they do not provide mention-level entities and neutral labels. Instead, all the entities are in canonical form without positions in the context. We slightly change the original experiment setup to make it more consistent with the DSE setting, providing the entity pair with the stance direction as part of the input. In this setting, the models are only asked to detect the non-neutral polarity, given an entity pair and the stance direction.

As a result, for experiments on DSE, we can naturally use both methods introduced in this chapter. While on SEESAW, the pairwise classification method is replaced with sentence-level classification, which uses the name and direction of the entity pair and context in a question-answering-based pair input. Detailed statistics and comparison of the two datasets are provided in Table 2.2.

2.4.2 Experimental Setup

For the classification-based method, we use RoBERTa [116] as the pretrained language model to obtain entity pair representations, and we choose `roberta-base`[1] as the base checkpoint to initialize the model. For the generation-based method, we finetune an openly available instruction-tuned large language model series, BLOOMZ [132]. We use `bloomz-560m`[2] as the initial checkpoint, as the model size is close to RoBERTa.

On DSE, we train the classification-based method using a learning rate of 2e-5. The batch size is 32, and λ is 0.1. We train the model with 30 epochs and evaluate it on the validation set to select the checkpoint with the best validation set performance. For the generation-based method, we use a learning rate of 2e-5. The batch size is 32 for the cross-entropy learning objective and 16 for soft consistency constrained learning. λ is 0.1. We train the generation-based method with 10 epochs and use the final checkpoint for the validation and test set evaluation.

On SEESAW, the generation-based method is trained with a learning rate of 1e-5, batch size of 32 for sequence log-likelihood objective, 16 for soft consistency constrained learning, and λ of 0.3. We train the generation-based method with 10 epochs and use the final checkpoint for validation and test set evaluation. The classification-based model, as we mention in Section 2.4.1, is a sequence classification given the entity pair with the direction, and the context in a question-answering-based sentence pair input. The template for input sentence pair is "`Source Entity:` e_1, `Target Entity:` e_2 `</s>` $\boldsymbol{X}$". We train this classification method using a learning rate of 2e-5. The batch size is 32, and λ is 1.0. We train the model with 30 epochs and evaluate it on the validation set to select the checkpoint with the best validation set performance.

We train all models with a linear scheduler with a warmup rate of 0.1. We use FP16 mixed precision training for the generation-based method. We use full fine-tuning on both classification-based and generation-based methods. For sentence pair sampling to conduct stance inference on the SEESAW dataset, we drop all sentences that include special entities such as `<author>` and `<someone>`. We also do not need to consider the normalization step because the direction is given as part of the input. For large language model inference, we use 4-bit quantization [37] to reduce memory consumption.

[1] https://huggingface.co/roberta-base

[2] https://huggingface.co/bigscience/bloomz-560m

Details of computational infrastructures. We use PyTorch [140], Huggingface Transformers [205] and Accelerate [49] to perform model training and inference. All model training is conducted with 1x or 2x Nvidia RTX 3090, or 1x Nvidia RTX A6000. BLOOMZ-176b inference is conducted with 4x Nvidia A100 SMX 40G. The Llama2-70b-chat inference is conducted with 2x Nvidia RTX 3090 or Nvidia A40.

Comparing with previous work. We compare our work with some previous work, including LNZ [107], DSE2QA [139] and POLITICS [117] on DSE. LNZ is a pairwise classification model that combines the entity prior representation and entity representation in the context. DSE2QA converts the stance detection problem into a series of template-based question answering. POLITICS is a pretrained model with ideological information.

As we alter the original experimental setting of SEESAW, we provide our own implementation of DSE2QA and POLITICS in these data and compare it with our method. Especially on these two datasets, we apply POLITICS with the same classification framework as our model. The only difference is their ideology-aware pretrained model.

In addition, we also compare our method with the large language model from the same series as our generation model, BLOOMZ-176b [132][3] with few-shot in-context learning samples on both datasets, to understand the capability of existing large language models to perform this entity-to-entity stance detection task. We take 5-shot samples of each label (in total 25 samples in DSE and 10 samples in SEESAW) to perform the language model inference.

2.4.3 Results

Table 2.3 shows the experimental results on the DSE dataset, where we can find steady improvement from adding the transitive consistency constrained learning to the classification-based (Generation + Consistency Training) and generation-based method (Generation + Consistency Training). We also observe that the improvement for the generation-based method is smaller than the classification-based method, indicating that there is still room to further investigate better methods to incorporate constrained learning into generative modeling. The results of a large language model with in-context learning is illustrated with BLOOMZ-176b. This result indicates that BLOOMZ-176b has a deficient performance on DSE, the entity-to-entity stance detection task, and the few-shot in-context learning cannot substantially help with learning well on this task.

[3] https://huggingface.co/bigscience/bloomz

Methods	Development Set		Test Set	
	Micro F_1	Macro F_1	Micro F_1	Macro F_1
LNZ (Combined)	69.40	65.16	70.55	53.58
LNZ (Context)	63.31	45.18	63.71	46.65
LNZ (EntityPrior)	59.14	44.27	58.53	40.63
DSE2QA (Complete)	78.92	67.51	77.26	66.17
DSE2QA (Pseudo)	80.72	68.27	79.73	67.66
POLITICS	85.45	71.94	84.19	71.12
Generation	83.92	70.14	83.25	70.12
+ Consistency Training	84.86	71.75	83.51	70.25
Classification	85.82	74.07	83.82	70.59
+ Consistency Training	**86.67**	**74.41**	**85.19**	**72.50**
BLOOMZ-176b + 25 samples	20.60	18.04	21.07	18.56

Table 2.3: Results on DSE dataset. The performances of our methods are averaged performance (%) over 5 runs.

On the SEESAW dataset, we also find that the consistency constrained learning provides consistent improvement to two base methods. The performance of the classification-based consistency constrained method outperforms or is on par with previous work, specifically compared to POLITICS, the model pretrained with ideology information. We can also observe that the absolute improvement is slightly less than what we observe in DSE, which indicates that the constrained learning objective works better when the stance directions are part of the prediction output. BLOOMZ-176b, contrary to the DSE results, also provides fair performance on this dataset. We will further discuss the large language model performance discrepancy between DSE and SEESAW in Section 2.4.6.

2.4.4 Effects of Soft Consistency Constrained Learning

We further analyze the effects of the soft consistency constrained learning and illustrate the results in Figure 2.3 on the DSE dataset. We can observe that after involving the soft consistency constrained objective, the performances compared to the one without the constrained objective ($\lambda = 0$) improve. However, enforcing this objective with large λ, similar to vanilla data augmentation, does not further contribute to the performance but instead results in performance

Methods	Micro F_1
DSE2QA	83.35
POLITICS	84.02
Generation	80.35
+ Consistency Training	81.05
Classification	83.72
+ Consistency Training	**84.11**
BLOOMZ-176b + 10 samples	77.29

Table 2.4: Results on SEESAW dataset. Different from the original setting of SEESAW, we provide entity pair and direction as the input and ask models to predict a non-neutral stance. The performances we reported are averaged performance (%) over 5 runs.

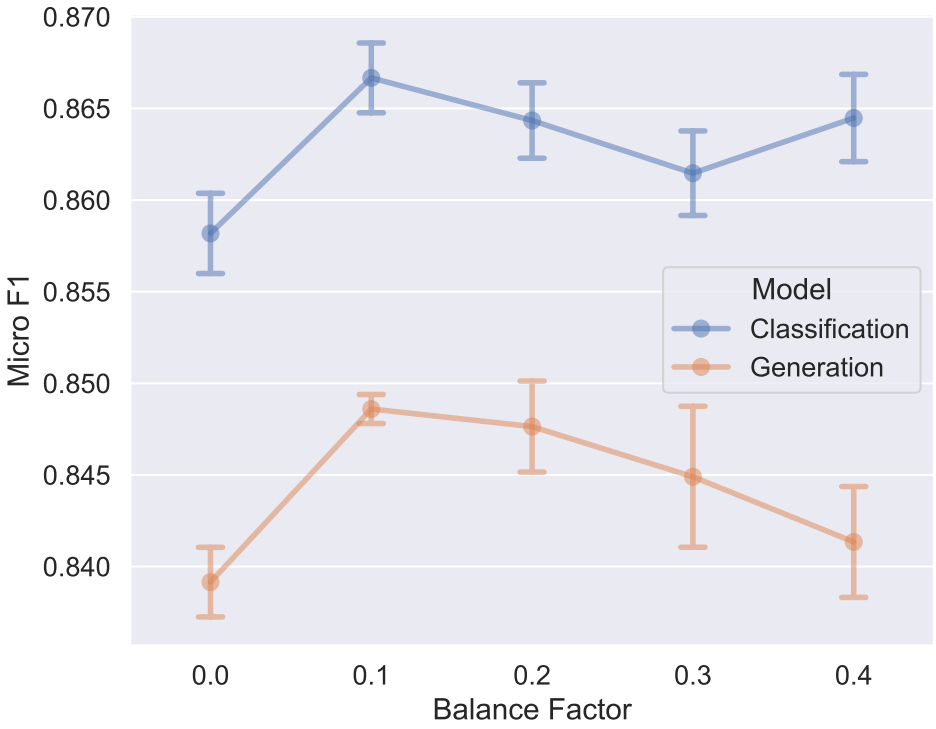


Figure 2.3: The Micro F_1 performances of generation and classification models on DSE development set with different balance factor λ. In general, we can observe that with the increase of λ, the performance first improves and then degrades.

Training	Micro F_1	Macro F_1
Data Augmentation	84.55	71.91
Consistency Learning	**85.19**	**72.50**

Table 2.5: Comparison between vanilla data augmentation and soft consistency constrained learning. Both methods use the same two-step sampling method to obtain the inferred stance of the cross-sentence entity pair.

Sampling	Micro F_1	Macro F_1
Random Sampling	84.44	71.99
Two-Step Sampling	**85.19**	**72.50**

Table 2.6: Effects of two-step sampling method compared to the vanilla uniform random sampling over all valid sentence pairs for stance inference on DSE test set.

degradation. This phenomenon suggests that transitivity does not always hold and that there is still a chance that the inference is not correct. Therefore, consistency constrained learning requires a carefully chosen soft setup.

In addition, we conduct another experiment to analyze the performance of soft consistency constrained learning compared to vanilla data augmentation on the classification-based method. The vanilla data augmentation uses the sample two-step sampling to obtain the inferred label for the cross-sentence entity pair. The results are shown in Table 2.5. We can find that both data augmentation and consistency learning can contribute to the model performance, while consistency constraints provide additional performance improvement from learning to make consistent predictions in a context.

2.4.5 Effects of Two-Step Sampling

We also analyze the effects of two-step sampling. The results are shown in Table 2.6. We compare the two-step sampling to uniform random sampling over all valid sentence pairs for stance inference. The results show that two-step sampling outperforms uniform sampling, indicating that it is important to consider the entity distributions when selecting the sentence pair. Vanilla uniform random sampling over all valid sentence pairs results in a long-tail distribution of the shared entity. However, constrained learning performs better when the shared entity follows a

Test Data Type	Micro F_1	Macro F_1
Full Label	21.07	18.56
- w/o Direction	26.74	26.19
- w/o Neutral	66.96	35.80
- w/o Both	77.62	71.73

Table 2.7: Analysis of BLOOMZ-176b performances on different test data on DSE, including test labels that do not require predicting direction, data excluding neutral samples, and test labels without both of them. The results show that the performance suffers from predicting the neutral labels and directional information.

uniform distribution.

2.4.6 Challenge of Large Language Models for Entity-to-Entity Stance Detection

From Table 2.3, we find surprisingly low performance from the large language model, while the performance on Table 2.4 is more promising. As we explained in Table 2.2, the main difference between the two datasets is the requirement of label direction and neutral sample detection. Therefore, we conduct further analysis to understand the performance discrepancy. Besides the original test label, we use test data without label direction (only requiring polarity prediction), test data without neutral label samples, and test data without both factors to analyze the impact of these factors. We conduct a similar in-context learning scheme as introduced in Section 2.4.2. The results show that the large language model achieves better performance by removing the requirement of neutral label prediction or predicting directed information. Similar results on Llama2-70b-chat can be found in Section 2.4.7.

This phenomenon reveals that the semantic information of directed stances and neutral stances is not well pretrained in the current large language models. In addition, it is also non-trivial to use the in-context learning method to help large language models obtain the ability to conduct directed stance and neutral stance detection. These results also partly align with Zhang et al. [221], which shows that large language models lag behind in complex or structured sentiment analysis tasks.

Test Data Type	Micro F_1	Macro F_1
Full Label	18.48	19.86
- w/o Direction	56.56	50.82
- w/o Neutral	30.37	30.63
- w/o Both	84.55	54.92

Table 2.8: Analysis of Llama2-70b-chat performances on different test data types on DSE. The results show that large language models suffer from predicting the neutral labels and directional information.

2.4.7 Additional Results on Llama2-70b-chat

Experiment results on Llama2-70b-chat are demonstrated in Table 2.8. The overall results are similar to the results of BLOOMZ-176b. When requiring directed stance analysis with neutral labels, Llama2-70b-chat with in-context learning provides deficient performance. If we simplify the problem so that output does not require a directed stance, or samples do not include neutral labels, Llama2-70b-chat shows more legitimate performance.

2.4.8 Large Language Model Prompts

We demonstrate the prompt templates for large language models in Figure 2.4. The prompts consist of a short description of the task and a series of examples. We list the sample to solve at the end of all demonstration examples.

BLOOMZ-176b

The task is to detect the stance from source entity to target entity given a context. The input consist a pair of entities and a context. Your output can only be "Neutral", "Entity 1 to Entity 2 Positive", "Entity 1 to Entity 2 Negative", "Entity 2 to Entity 1 Positive", "Entity 2 to Entity 1 Negative" without explanation. Below are a few examples:

Context: …
Entity 1: …
Entity 2: …
Stance: …

Context: …
…

Llama2-70b-chat

<s>[INST] <<SYS>>
The task is to detect the stance from source entity to target entity given a context. The input consist a pair of entities and a context. Your output can only be "Neutral", "Entity 1 to Entity 2 Positive", "Entity 1 to Entity 2 Negative", "Entity 2 to Entity 1 Positive", "Entity 2 to Entity 1 Negative" without explanation. Below are a few examples:

Context: …
Entity 1: …
Entity 2: …
Stance: …

Context: …
…
<</SYS>>
Context: …
Entity 1: …
Entity 2: …
[/INST] Stance:

Figure 2.4: Prompt templates for BLOOMZ-176b and Llama2-70b-chat.

Asked about quarterback (**e1, Colin Kaepernick**) favoriting negative comments on Twitter as a form of personal motivation, (**e2, Harbaugh**) gave it a thumbs up.
Classification Prediction: Neutral
Classification + Consistency Training: $e_2 \overset{\text{Positive}}{\longrightarrow} e_1$
In the primaries, (**e1, Morell**) said, Putin played upon Mr. (**e2, Trump**)'s vulnerabilities by complimenting him.
Classification Prediction: Neutral
Classification + Consistency Training: $e_1 \overset{\text{Negative}}{\longrightarrow} e_2$

Table 2.9: Case study to compare the differences between vanilla classification model and classification model with consistency transitive constrained learning.

2.4.9 Case Study

We also include two cases from the classification-based method on the DSA dataset to demonstrate the effects of consistency constrained learning, as shown in Table 2.9. In these two examples, we can find that consistency learning can help in finding the stance label and direction in the context, while the baseline classification model only predicts neutral.

2.4.10 Limitations

In this work, our experimental setup assumes that the entities involved in a context are pre-extracted, and we use gold standard entities for stance detection. However, to conduct end-to-end entity-to-entity stance detection, we need an additional prerequisite component for entity extraction, which is not used and covered in this chapter. Therefore, it is difficult to compare this work with other work that conducts end-to-end entity-to-entity stance detection or structured sentiment analysis, such as generative entity-to-entity stance detection that jointly finds entities with their stances [222].

In addition, the consistency constraints in this chapter are used during training. However, for large language models discussed in this chapter, it is infeasible to conduct full fine-tuning with limited computational resources. It is still under exploration how to use frozen large language models to obtain reliable performances for this challenge and whether those consistency constraints can also be effectively used in this setup.

In this chapter, our experiments are conducted in a specialized domain, political news, in

which we generally see more frequently polarized opinions. It is still under exploration whether the stance transitivity constraints widely exist in other domains. If not, we need to find scenarios where transitivity constraints hold and conduct constrained learning on these scenarios specifically. In addition, in the general domain, the negation of a positive stance may not be exactly the opposite one, which should also be considered when extending this work to a more general domain. It is also an interesting direction to study similar transitivity in other settings (*e.g.,* relations in knowledge graphs, semantic concept inheritance).

2.5 Related Work

Earlier efforts on stance detection primarily focus on some specific targets with rich training and testing data [15, 130, 171]. A typical model in this setting is built for each target separately [6, 48, 130, 131, 169], or cross-target stance detection, where we have pre-defined leave-out targets to test the model generalization to targets that do not have training data [9, 71, 104, 209]. More recent efforts also study zero-shot or few-shot stance detection on a large number of targets [7, 56, 105, 106, 110, 112, 200]. This setting requires the model to generalize to a large number of unseen targets. Recently, another line of research studies a more objective form of stance detection, entity-to-entity stance detection [139, 222], where we analyze the stance from one entity to another entity in the text. Our work follows this direction and studies the consistency between related entity-to-entity stances and uses this consistency to help model training, compared to previous work [139, 222] that tackles the stances individually.

On the other hand, stance detection can be considered a simplified task of structured sentiment analysis [19], which identifies opinion holders, targets, expressions, and polarities into dependency structures. Typical stance detection setups assume that the opinions are from the author, and models only need to consider the target. However, entity-to-entity stance detection combines holders, targets, and polarities with more streamlined, directed link labels.

The consistency assumption between related stances is also related to multi-target stance detection [170]. Multi-target stance detection is to detect a stance pair for a multi-target (*e.g.,* a pair of targets), assuming that when expressing the stance to one target, it also implies stances to a related target. Similar consistency constraints have also been discussed on polarity link prediction in social networks [91] and event relation extraction [192]. The focus of Leskovec et al. [91] is the network-based link prediction, which is quite different from the text-based analysis. Wang et al. [192] performs event-event relation extraction (temporal and hierarchical). The document-level annotation provides consistent labels, and therefore, they do not have steps that

we have to create data for consistency training. The transitive assumption is also aligned with the motivation of the connotation frames [152], where a particular predicate may connote some implied presupposed facts or sentiments. Allaway and McKeown [8] also presents a method for capturing connotations regarding the cultural and emotional perspectives of the speaker. In our work, the transitive rules that we use can be considered a simplification of the connotation, and future work can extend it with more accurate features from lexical connotations.

2.6 Conclusion

In this chapter, we present a method that models the transitive consistency constraints during training to help train entity-to-entity stance detection models. Our proposed methods first sample sentence pairs to conduct stance transitivity inference and model the constraints as the similarity between the inferred and directly predicted stance. Experiments show that this constrained learning helps improve both classification- and generation-based models. Further analysis indicates that constrained learning is sensitive to the balance factor that controls the enforcement of constraints during training. We also find that large language models may not perform reliable complex structured predictions, especially on neutral and directed samples.

Part II

Knowledge-Enhanced Inference: Knowledge-Seeking with Generative Modeling

Chapter 3

On Synthetic Data Strategies for Domain-Specific Generative Retrieval

In this chapter, we discuss the data construction for using generative modeling to perform domain-specific knowledge retrieval. We focus on generative retrieval, where we use the generative models to generate a series of tokens that can exactly match pre-defined document identifiers as the retrieval process. We study data strategies for a two-stage training framework. In the first stage, which focuses on learning to decode document identifiers from queries, we investigate LLM-generated queries across multiple granularity (*e.g.*, chunks, sentences) and domain-relevant search constraints that can better capture nuanced relevancy signals. In the second stage, which aims to refine document ranking through preference learning, we explore the strategies for mining hard negatives based on the initial model's predictions.

3.1 Overview

Generative retrieval is emerging as a promising paradigm for information retrieval (IR), leveraging generative models (*e.g.,* Transformers, 188) to directly produce ranked lists of potentially relevant document identifiers for a user query. During generative retrieval model training, we generate synthetic queries that are relevant to each document in the corpus, and ask the model to take those synthetic queries to produce corresponding relevant document identifiers. Although prior work has made progress on various fronts, including training strategies (*e.g.,* identifier choices) [21, 174, 183, 225], modeling techniques [31, 102, 226], and inference methods [26, 88, 217], the role of *data strategies* in training generative retrieval models, particularly when dealing with domain-specific corpora, remains relatively underexplored. This gap is criti-

cal: as generative retrieval models internalize the entire corpus within their parametric memory, the choice and quality of training data are likely to play a critical role in their performance.

Because generative retrieval models internalize the entire corpus, they require training to remember and retrieve every document in the corpus. To mitigate the high cost and scalability challenges of in-domain annotation, most studies have adopted DSI-QG [231], which uses synthetic queries generated at the passage level by docT5query (a model trained on MS-MARCO data[134]) on every document in the corpus. During model training, we will use these synthetic queries as input to train the generative retrieval model to retrieve relevant documents by generating the corresponding document identifiers. However, applying such off-the-shelf synthetic data strategies to new domains may not suffice. Unlike dense retrieval approaches, which focuses on strong text representation [65, 75], a generative retriever must develop three key capabilities: (1) **memorization** (storing the content of the corpus (*e.g.,* documents) and mapping them to their assigned identifiers), (2) **generalization** (inferring beyond explicit textual cues from user queries), and (3) **relevance scoring** (accurately ranking document identifiers by relevance to a given query). Domain-specific corpora can amplify these challenges, as the model must adapt its internal representations to reflect domain nuances while maintaining robust generalization and ranking accuracy. In this work, we systematically investigate the data strategies that can promote these core capabilities.

We introduce a two-stage training framework. The first stage focuses on mapping an input directly to document identifiers via supervised fine-tuning on synthetic data. The second stage uses preference learning to further improve ranking performance [102, 226]. Here, we adopt Regularized Preference Optimization (137, RPO), an effective alternative to PPO-based reinforcement learning [135]. We study the data strategies for both stages.

The first stage focuses on the memorization and generalization ability. We examine two data sources as input for decoding document identifiers during training: *context* data (*e.g.,* chunks) directly extracted from the corpus and *synthetic queries* that represent various relevance signals. For synthetic queries, we investigate query generation using multi-granular context (*e.g.,* sentence-level, chunk-level) to capture both local and global information from the corpus. We also explore adding constraints derived from available metadata or domain-specific knowledge when generating synthetic queries to enhance the model's ability to handle complex domain-relevant queries.

Models trained during the first stage are only optimized to produce a single positive candidate, which lacks relevance modeling among different candidates. In the second stage, we further create data to refine the model's ranking capability through preference learning [102, 226].

We study the selection of negative candidate documents for preference learning. Instead of relying on static offline data, we collect preference data online from the model's top-ranked candidates after the first stage and compare it to random sampling from the corpus. We further investigate the choices and impact of varying the number of negative candidates on the ranking performance.

We conduct experiments on datasets covering various aspects of relevance, including the widely adopted Natural Questions (83, NQ), a multi-hop dataset MultiHop-RAG [180], and two perspective-based retrieval datasets: AllSides [17] and AGNews [214] from Zhao et al. [223]. We show that queries with different aspects, such as multi-granular and constraints-based queries, significantly improve the retrieval performance compared to relying solely on chunk-level synthetic queries from query generation models. Additionally, upsampling context data further improves performance. Moreover, we show that these data strategies generalize well to other types of document identifiers, such as atomic identifiers. Finally, we demonstrate that RPO effectively improves the ranking performance of generative retrieval and that the key lies in the selection of high-quality negative candidates. High-quality hard negative candidates improve performance, while random negatives may have an adverse impact.

In summary, this work offers a comprehensive investigation of data strategies for building scalable and effective domain-specific generative retrieval systems. Our findings emphasize the importance of creating high-quality and diverse synthetic queries that capture multiple levels of granularity within the corpus, as well as informed negative selection strategies for ranking optimization.

3.2 Generative Retrieval Framework

A typical generative retrieval framework takes a query as input and generates the corresponding relevant document identifiers as the retrieval results [183]. Each document has a unique identifier, so we can use the document identifiers to find documents for downstream tasks. During training, we typically generate synthetic queries that are relevant to each document in the corpus. Then, we use those synthetic queries as training queries to train generative retrieval models to produce the corresponding relevant document identifiers.

3.2.1 Document Identifiers

We primarily use semantic document identifiers in our experiments because of their superior performance and scalability to larger corpus. Instead of using corpus-specific semantic identifiers such as titles or URLs, we adopt a more general, keyword-based approach that is applicable to a wide range of corpora [225]. Specifically, we instruct an LLM to produce a list of keywords that represent a document and use this keyword list as its semantic identifier. We also extend our synthetic data strategies to other types of identifiers, such as atomic identifiers, to validate their generalizability.

3.2.2 Generative Modeling

The generative retrieval model learns from generating relevant document identifiers for a given query. Formally, we assume that there is a query q and its relevant document d, and d' is the document identifier of d. The goal of generative retrieval is to produce d' given q, which can be represented as:

$$\begin{aligned} \text{score}(q, d) &= P\left(d' \mid q; \theta\right) \\ &= \prod_i P\left(d'_i \mid d'_{<i}, q; \theta\right), \end{aligned}$$

where d'_i is the i^{th} token of the identifier. To ensure the validity of the identifiers generated during inference, we use constrained beam search with Trie [26] to limit the output token space at each decoding step. The top-k output from the beam search serves as the retrieval results.

Compared to dense retrieval models [75], generative retrieval simplifies the retrieval process by directly performing the retrieval without external indexing. However, there are some unique challenges in learning a generative retrieval model. As they solely rely on parametric knowledge, these models must not only learn the retrieval task but also memorize and comprehend the document content by associating it with corresponding identifiers. Therefore, generative retrieval training typically requires processing the entire corpus.

3.3 Supervised Fine-Tuning Data Strategy

In a typical domain-specific setup, we often assume access to a corpus with little or no labeled data for domain-specific training [55]. Therefore, it is crucial to create high-quality synthetic data for generative retrieval training that thoroughly covers all documents in the corpus.

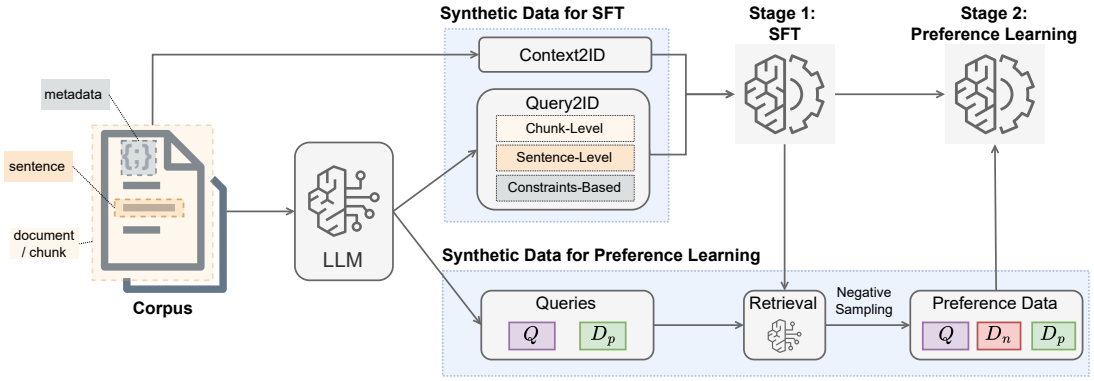


Figure 3.1: The overall workflow of the generative retrieval training and synthetic data utilization at each stage.

Our synthetic data comprises two components: Context2ID data and Query2ID data. Context2ID involves training the model to retrieve document identifiers based on document content. Query2ID focuses on teaching the model to retrieve relevant document identifiers from a given query. All the specific prompts we used in this section can be found in Section 3.8.

3.3.1 Supervised Fine-Tuning Objective

At this stage, we train the generative models to generate relevant document identifiers by maximizing the individual token probabilities. While typical supervised fine-tuning (SFT), especially with encoder-decoder architectures, focuses on optimizing the output (*i.e.* identifiers), it is also part of the training goal for generative retrieval models to comprehend and memorize the context. To this end, we also optimize the model for learning to decode the input. Specifically, for a given query-document pair (q, d), where the query can be an actual query or the context, we maximize the likelihood of the combined input and output sequence:

$$\begin{aligned}\mathcal{L}_{\text{sft}}(q, d) &= -\log P(d', q; \theta) \\ &= -\sum_i \log P(q_i \mid q_{<i}; \theta) - \sum_i \log P(d'_i \mid d'_{<i}, q; \theta).\end{aligned}$$

3.3.2 Context2ID

Context2ID data is to learn to use document content to retrieve its document identifier. We enumerate each chunk in the corpus and create a context to document identifier mapping. The

Data Type	Example
Context	title: Christmas Day preview: 49ers , Ravens square off in potential Super Bowl sneak peek... source: Yardbarker ... San Francisco has racked up an NFL-leading 25 turnovers and has given up the second-fewest rushing yards (1,252) , ...
Chunk-Level Query	What is the potential implication of this matchup between the 49ers and Ravens ?
Sentence-Level Query	Where does the 49ers ' defense stand in terms of total yards allowed per game?
Constraints-Based Query	According to the Yardbarker article, which team has the league's most effective running game?

Table 3.1: Examples of different synthetic queries generated from MultiHop-RAG corpus.

goal of Context2ID data is to help the generative retrieval model remember the document content and build the association between the document content and the corresponding document identifier.

3.3.3 Query2ID

Query2ID is to learn to use a query to retrieve the relevant document identifiers. It helps the model to learn the retrieval task itself and also to better comprehend the content from the query perspective.

Previous work [231] finds that it is effective to use a query generation model (*e.g.*, docT5query, 134) to produce synthetic queries for all documents with multiple independent sampling. In this work, we instead use an LLM for synthetic query generation. We ask the LLM to generate a diverse set of m queries given a context, which can easily acquire a set of different queries compared to multiple independent sampling and filtering for query generation models.

We consider the synthetic query generation from two different perspectives. One perspective is the use of context from different levels of granularity. The other perspective is constraints-based query generation for domain-specific settings.

Multi-Granular Query Generation

We first consider generating queries with context at different levels of granularity. We mainly consider two different levels: the chunk level and the sentence level. Chunk-level synthetic

queries are produced by giving the whole chunk as the input context, while sentence-level synthetic queries are produced by only giving an individual sentence. Similar to previous work [231], the goal of producing chunk-level synthetic queries is to capture the overall semantic information of the chunk. The goal of producing sentence-level synthetic queries is to further capture the detailed semantic information in each sentence.

For each chunk, we ask the LLM to produce m_c chunk-level queries. Then, we enumerate each sentence in the chunk and ask the LLM to produce m_s sentence-level queries for each sentence.

Constraints-Based Query Generation

One unique advantage of LLM-based query generation is that we can further provide instruction to guide the LLM to produce queries that fit the specific domain setting. Therefore, we also propose asking LLM to produce queries that include constraints of the documents derived from their metadata, such as the author of the document or the political polarity of the document content. Those constraints are in general domain-specific but common in the real world. Table 3.3 specifies the attributes that we use to produce constraints-based synthetic queries for each dataset. We ask the LLM to generate m_i queries for each document.

3.4 Preference Learning Data Strategy

Previous work [102, 226] shows that learning from ranking tasks can further improve the relevance modeling of generative retrieval models. But when generative retrieval models are based on large models, optimization from complex ranking tasks, such as the listwise optimization, may not be computationally efficient as there will be multiple forward passes. In this work, we instead use a simplified method, adopting reinforcement learning from human feedback algorithm to perform preference optimization [137], as those algorithms are widely applied in optimizing large language models. We will first briefly introduce the preference optimization that we use. Then, our focus will be on the synthetic data construction, which consists of the synthetic queries and their corresponding preferred or rejected candidates.

3.4.1 Preference Optimization Objective

We use Regularized Preference Optimization (137, RPO) as our optimization method for preference learning. It is an extended version of Directed Preference Optimization (147, DPO),

including additional supervised fine-tuning loss to alleviate the over-optimization issues on negative responses. It takes an input query q, a positive candidate d_p, and a negative candidate d_n as input. The loss is in favor of the positive candidate while against the negative candidate

$$\mathcal{L}_{\text{rpo}}\left(q, d_p, d_n\right) = -\log \delta\left(\beta \log \frac{P\left(d_p' \mid q; \theta\right)}{P\left(d_p' \mid q; \theta_{\text{ref}}\right)} - \beta \log \frac{P\left(d_n' \mid q; \theta\right)}{P\left(d_n' \mid q; \theta_{\text{ref}}\right)}\right) - \alpha \frac{\log P(d_p' \mid q; \theta)}{\left|d_p'\right|},$$

where θ_{ref} is the parameter of the reference model, *i.e.,* the supervised fine-tuned model from the first stage training. d_p' and d_n' are the identifiers of the positive and negative candidate, respectively.

3.4.2 Synthetic Queries

Similar to the previous section, in a domain-specific setup, we assume that we do not have enough data for model training. Therefore, after the supervised fine-tuning stage, we need a batch of new synthetic queries for preference learning.

We still adopt the LLM-based query generation as with the supervised fine-tuning stage. However, there are a few key differences in the instructions. First of all, we ask the LLM to make queries as difficult as possible. At the same time, we ask the LLM to provide not only the synthetic queries but also their corresponding answers. This is to ensure that, while making difficult queries, those synthetic queries are still answerable using the given context.

These changes make the new batch of synthetic queries different from queries used during supervised fine-tuning so that the model will not be over-optimized to the same batch of data. Intensifying the difficulties also increases the likelihood that the initial generative retrieval model makes mistakes, and therefore the model will benefit from the preference learning by learning from those mistakes.

3.4.3 Candidate Selection

After producing the synthetic queries, the next step is to select document candidate pairs for RPO optimization. For each training instance, we need one positive candidate and one negative candidate. As we always produce synthetic queries based on a document, the positive candidate can be naturally assigned. Therefore, the focus will be on selecting negative candidates for each synthetic query.

To increase the hardness of the negative candidates, we choose to select negative candidates from the retrieval results. Specifically, after the supervised fine-tuning stage, we will use the

generative retrieval model to perform retrieval on the synthetic queries for preference learning. Our strategy focuses mainly on selecting the top-k candidates with ranks higher than the positive candidate from the retrieval results. In this way, if the positive candidate ranks in the top-1, we will not use the query for preference learning. If the rank of the positive candidate is higher than k, then there will be different numbers of negative candidates, depending on the rank. If the rank is lower than k, there will be k different negative candidates. When there are multiple negative candidates, we pair each negative candidate with the positive one to form a candidate pair instance for preference learning.

3.5 Experiments

3.5.1 Datasets

We choose 4 datasets for our experiments: three domain-specific corpora – MultiHop-RAG [180], –AllSides [17] and AGNews [214] from Zhao et al. [223] – as well as the general-domain dataset Natural Questions dataset (83, NQ).

For AllSides and AGNews, we mainly adopt queries from Zhao et al. [223]. In the case of AGNews, we replace the similar document part in queries with another attribute of perspective, as we focus on the query retrieval rather than document similarity search.

For NQ, we use the "old document" split from Kishore et al. [79], which constructs a subset of Wikipedia pages containing all positive candidates for training and testing while keeping the corpus size manageable for generative retrieval training.

Dataset	**Context**	**Queries**		
		Chunk-Level	**Sentence-Level**	**Constraints-Based**
MultiHop-RAG	7,724	72,090	472,193	51,212
AllSides	645	6,313	173,898	6,091
AGNews	1,050	10,355	80,524	20,875
NQ	98,748	1,459,031	-	-

Table 3.2: Dataset Statistics

Dataset	Attributes
MultiHop-RAG	author, publish time, source, category, title
AllSides	political polarity
AGNews	location, topic

Table 3.3: Attributes used in each dataset for constraints-based query generation.

3.5.2 Experiment Setup

For all datasets, we use Mistral 7b [69] series as the generative retrieval base model. We use Mixtral 8x7b [70] to generate all synthetic queries and we use Claude 3 Sonnet [12] to generate keywords. We use `Mistral-7B-Instruct-v0.3` as the base model for generative retrieval with the semantic identifier, while we use `Mistral-7B-v0.3` as the base model for the atomic identifier, as it is closer to a classification setting.

For supervised fine-tuning, we train the models with 2 epochs, with a learning rate of 2e-5 and a warmup ratio of 0.1. The batch size is set as 256. We use sequence packing to put multiple examples in one forward pass [148]. We use `bfloat16` for our training.

For preference learning, we mainly conduct experiments on MultiHop-RAG and NQ with semantic identifiers. We train the models with 1 epoch. The learning rate is set as 1e-7, the batch size is set as 64, β is set as 0.5, and α is set as 1.0.

The training infrastructure includes TRL [189], Accelerate [49], Transformers [206], DeepSpeed [153] and FlashAttention-2 [35]. We use 8x Nvidia A100-SXM4-40GB for our experiments. Each training or inference procedure can be completed in 1 day.

Statistics of the numbers of the documents, different synthetic queries can be found in Table 3.2. The attributes used for constraints-based synthetic queries can be found in Table 3.3. All the experiment results are obtained with a single run.

MultiHop-RAG

On MultiHop-RAG, we split the documents into chunks with a maximum length of 256 without overlap and conduct retrieval on individual chunks. For synthetic query generation, m_c, m_s, and m_i are set as 10, and the temperature for LLM inference on synthetic data generation is set as 0.7. We interleave the Context2ID and Query2ID data as the full dataset for supervised fine-tuning. The maximum sequence length is set as 700. For synthetic queries for preference

learning, we ask the LLM to generate 10 queries. We perform the retrieval with beam size as 10 and retrieve the top-10 candidates for each query to construct the candidate pairs.

AllSides

On AllSides, we conduct document-level retrieval. For synthetic query generation, m_c, m_s, and m_i are set as 10, and the temperature for LLM inference on synthetic data generation is set as 0.7. For Context2ID data, as there are some long documents in the corpus, we will split the long context into chunks with a maximum length of 256 without overlap. The Context2ID data is constructed to use all chunks in the document to predict its corresponding document identifier. We interleave the Context2ID and Query2ID data as the full dataset for supervised fine-tuning. The maximum sequence length is set as 700.

AGNews

On AllSides, we conduct document-level retrieval. For synthetic query generation, m_c, m_s, and m_i are set as 10, and the temperature for LLM inference on synthetic data generation is set as 0.7. The queries constructed by Zhao et al. [223] use two different perspectives. The first perspective is either the location of the desired news or the topic, while the second perspective is that the news is similar to another given news in the query. As we mentioned in Section 3.5.2, we replace the second perspective with another field so that each query consists of both location and topic perspectives. The topic and location information used for instruction-based synthetic query generation is extracted with Mixtral 8x7b. We interleave the Context2ID and Query2ID data as the full dataset for supervised fine-tuning. The maximum sequence length is set as 700.

NQ

On NQ, we conduct document-level retrieval. We use the document prefixes from [79] to produce the semantic identifiers. For synthetic query generation, we perform truncation on pages when they are too long so that we always have at least 1024 token space for model output. We set m_c as 15 and the temperature as 0.7. We do not include sentence-level synthetic queries as the number of those queries is too large to be included in training within a reasonable time. Instead, we include sentence-level Context2ID as the approximation and use the sentences from the document prefixes from [79] to predict the corresponding document identifiers. In NQ, we have high-quality human-annotated training queries, which we also include as part of the

	HIT@4	HIT@10	MAP@10	MRR@10
Chunk	43.64	66.65	13.98	31.14
+Sent	61.64	81.69	22.13	47.20

Table 3.4: Ablation study on the effect of synthetic queries generated at a sentence-level granularity of context.

Query2ID data, and therefore, we do not include instruction-based synthetic queries. We concatenate the Context2ID and Query2ID data as the full dataset for supervised fine-tuning, as interleaving will produce a much larger dataset that cannot be trained within a reasonable time. The maximum sequence length is set as 450. For synthetic queries for preference learning, we also perform truncation as for supervised fine-tuning and ask the LLM to generate 10 queries. As the generated query number is quite large for inference, we use the first 2 generated queries for each document for preference learning. We perform the retrieval with beam size as 10 and retrieve the top-10 candidates for each query to construct the candidate pairs.

3.5.3 Results

We will discuss our experiment results for each of the stages. In the supervised fine-tuning stage, we will discuss the effects of multi-granular synthetic queries, synthetic data with domain-specific constraints, and the use of Context2ID data. For the preference learning stage, we will discuss the use of different candidates for preference learning.

Supervised Fine-Tuning Stage

Effects of multi-granular synthetic queries. We conduct an analysis on the effects of incorporating synthetic queries generated from the context at different levels of granularity on MultiHop-RAG. We train the generative retrieval model based on semantic identifiers on chunk-level Query2ID data (Chunk), comparing it with the model trained on chunk-level and sentence-level Query2ID data (+Sent), and both models use Context2ID data. The results are shown in Table 3.4. We find that sentence-level synthetic queries can significantly improve retrieval performance, indicating that synthetic query generation with a small context can help capture more details from the document.

	MultiHop-RAG				AllSides			AGNews		
	HIT@4	**HIT@10**	**MAP@10**	**MRR@10**	**HIT@1**	**HIT@5**	**HIT@10**	**HIT@1**	**HIT@5**	**HIT@10**
w/o constraints	61.64	81.69	22.13	47.20	10.19	29.63	47.22	59.91	83.94	88.11
w/ constraints	69.98	88.34	24.85	52.29	14.20	38.58	51.85	62.19	83.78	88.24

Table 3.5: Ablation study on generative retrieval performances with or without the constraints-based synthetic queries.

	MultiHop-RAG				Natural Questions			
	HIT@4	**HIT@10**	**MAP@10**	**MRR@10**	**HIT@1**	**HIT@5**	**HIT@10**	**MRR@10**
w/o Context2ID	41.33	69.31	14.45	31.25	69.72	85.58	89.01	76.57
w/ Context2ID	69.98	88.34	24.85	52.29	70.71	86.48	89.85	77.54

Table 3.6: Ablation study on generative retrieval performance trained with or without Context2ID data. The results demonstrate the helpfulness of Context2ID data and learning to memorize the context for generative retrieval.

Effects of constraints-based synthetic queries. We further study the use of constraints-based synthetic queries that are customized for each domain-specific setting. We conduct experiments on three domain-specific corpora: MultiHop-RAG, AllSides, and AGNews. We compare the semantic identifier-based generative retrieval model trained with or without constraints-based synthetic queries combined with the corresponding Context2ID data. The results are shown in Table 3.5. The results show that constraints-based synthetic queries can further improve retrieval performance, indicating that it is helpful to use LLM-produced synthetic queries for domain customization.

Effects of Context2ID data. Existing work [79, 231] debates whether Context2ID data are useful for generative retrieval training. In this work, we consider Context2ID data as an important part of the data recipe and also include the memorization of the context as part of the supervised fine-tuning objective. Therefore, we conduct an analysis that removes the Context2ID data on MultiHop-RAG and NQ, and the results are shown in Table 3.6. We can find that Context2ID data consistently improves generative retrieval performance. We also include the comparison of the strategies to combine Query2ID and Context2ID data, including simple concatenation or interleaving that will upsample Context2ID data on MultiHop-RAG in Table 3.7, again illustrating the importance of Context2ID and that learning to memorize context

	HIT@4	HIT@10	MAP@10	MRR@10
Concat	44.30	72.77	15.64	33.59
Interleave	69.98	88.34	24.85	52.29

Table 3.7: Analysis on different ways of combining Query2ID and Context2ID data. We compare simple concatenation (Concat) and interleaving (Interleave) that inherently upsamples the Context2ID data.

may strengthen the effects on Context2ID.

	MultiHop-RAG				Natural Questions			
	HIT@4	HIT@10	MAP@10	MRR@10	HIT@1	HIT@5	HIT@10	MRR@10
docT5query	50.86	73.30	17.60	37.73	63.3	79.12	85.18	70.30
Mixtral 8x7b	61.64	81.69	22.13	47.20	70.71	86.48	89.85	77.54

Table 3.8: Generative retrieval performance with synthetic queries from Mixtral 8x7b and docT5query. The results show that queries from Mixtral 8x7b can help train a better generative retrieval model.

Different query generation models. As we largely use LLM to produce synthetic queries in this work, it is important to understand the performance and effects of using an LLM compared to a specialized query generation model. Therefore, we conduct a comparison between synthetic queries from Mixtral 8x7b and docT5query, and the results are shown in Table 3.8. For a fair comparison, we do not include constraints-based queries from the LLM, as those queries cannot be produced from docT5query. The results show that generative retrieval models trained with queries from Mixtral 8x7b consistently perform better than models trained with queries from docT5query. Following Pradeep et al. [142], we use Jaccard similarity to evaluate the semantic similarity between test queries and synthetic queries as a post-analysis. The results in Figure 3.2 illustrate that synthetic queries from Mixtral 8x7b, in general, have a higher semantic similarity to test queries.

Generalization to different identifiers. We further study the generalizability of our data strategies across different types of document identifiers. In this analysis, we use atomic identifiers, which are arbitrary unique IDs assigned to each document or chunk. We conduct ex-

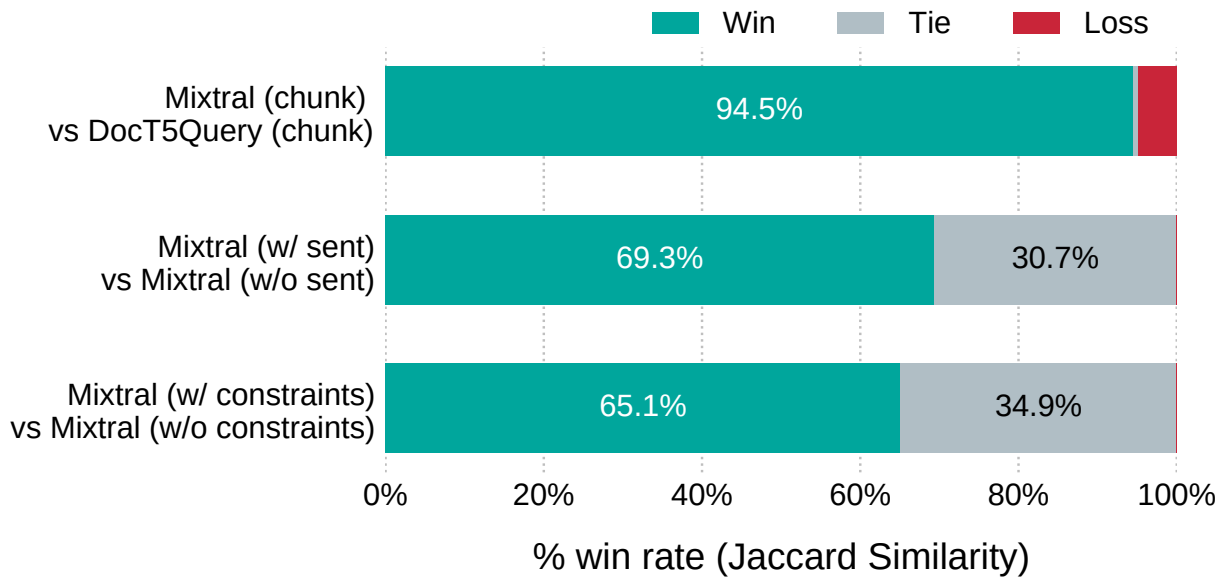


Figure 3.2: Jaccard similarity post-analysis on MultiHop-RAG test set. Synthetic queries from Mixtral 8x7b are generally closer to the test set than those from docT5query. Besides, incorporating granularity and domain-specific attributes further helps with getting queries that are closer to the test set.

periments on MultiHop-RAG, and the results are shown in Table 3.9. The findings align with our observations using semantic identifiers, highlighting the critical role of all three data types in generative retrieval. Among them, sentence-level synthetic queries contribute the most to performance improvements.

	HIT@4	HIT@10	MAP@10	MRR@10
all	74.32	88.03	29.71	59.26
w/o Context2ID	72.15	86.21	28.54	57.50
w/o Sent	58.40	75.17	21.51	44.76
w/o constraints	68.34	83.73	26.28	53.83

Table 3.9: Ablation study on atomic identifier-based generative retrieval performance on MultiHop-RAG.

	MultiHop-RAG				Natural Questions			
	HIT@4	HIT@10	MAP@10	MRR@10	HIT@1	HIT@5	HIT@10	MRR@10
SFT	69.98	88.34	24.85	52.29	70.71	86.48	89.85	77.54
Random 5	58.94	82.88	20.88	43.53	70.19	86.48	89.50	77.17
Top-5 negative	71.53	89.62	26.36	55.40	71.02	87.32	90.04	78.02
Top-10 negative	71.88	89.80	26.23	54.94	71.22	87.41	89.97	78.14

Table 3.10: Preference learning with different numbers of negative candidates. The results show that it is an effective strategy to select negative candidates with ranks higher than the positive candidate, while different numbers of negative candidates may optimize the retrieval performance in different ways.

DocT5Query-based Synthetic Queries	LLM-Based Synthetic Queries
what is the biggest baseball stadium?	How is the atmosphere at Dodger Stadium different from other cities?
largest baseball stadium in Boston	What happens to the intensity of Dodger Stadium during playoff games?
how many people at dodger stadium	How has the noise level in Dodger Stadium affected opposing pitchers this season?
what is the biggest stadium in baseball	How many decks does Dodger Stadium have?
what stadium holds the greatest baseball stadiums	What is unique about the seating arrangement in Dodger Stadium?

Table 3.11: Examples of synthetic queries generated from DocT5Query and Mixtral 8x7b.

Qualitative Analysis on Synthetic Queries Table 3.11 demonstrates some example synthetic queries from DocT5Query and Mixtral 8x7b with the same input context. We find that synthetic queries from a specialized query generation model, such as DocT5Query, are generally much shorter and only broadly related to the context. However, LLM-based query generation can produce more complex queries with more details, which are more closely related to the input context. Therefore, models trained with queries generated from LLM can better capture detailed information in the input queries to accurately associate it with relevant documents.

Preference Learning Stage

Effects of negative candidate sources. We first study the strategies for selecting candidates for preference learning. We compare randomly selecting from the corpus or using the top candidates from the generative retrieval model after supervised fine-tuning. The results are shown in Table 3.10, which illustrates that candidate selection has an impact on preference learning, and simple negative candidates may have a negative impact.

Effects of negative candidate number. We also study the effects of using different negative candidate numbers for each query. We experiment with selecting Top-5 and Top-10 negative candidates with a rank higher than the positive candidate from the retrieval results. The results are shown in Table 3.10. In general, it is effective to use the strategy, which includes high-quality candidates with ranks higher than the corresponding positive candidates. We also see some slight differences when including different numbers of negative candidates. We find that a large number of negative candidates helps better in metrics such as HIT@1 and HIT@4.

Comparison to Off-The-Shelf Retrievers

We also compare our generative retrieval performance with some off-the-shelf retrievers, such as BM25 [156], bge-large-en-v1.5 [208], Contriever-msmarco [65], E5-mistral-7b-instruct [195] and GTE-Qwen2-7B-instruct [103]. The results are shown in Figure 3.3, and more detailed results can be found in Table 3.12. We run the retrieval models on MultiHop-RAG, NQ, and AGNews to collect the results and adopt the AllSides results from Zhao et al. [223]. The results show that generative retrieval models that fully rely on in-domain synthetic data training without retrieval pre-training can achieve competitive performance compared to those retrievers. These results indicate the potential of generative retrieval and the use of LLMs as a tool to generate synthetic data that fits domain-specific requirements.

Model	HIT@4	HIT@10	MAP@10	MRR@10
BM25	64.35	78.31	**26.30**	**58.32**
bge-large-en-v1.5	58.80	78.36	19.96	42.57
Contriever-msmarco	55.25	75.08	19.28	40.69
E5-mistral-7b-instruct	54.01	79.56	19.11	40.77
GTE-Qwen2-7B-instruct	63.24	83.55	22.02	47.50
ours	**71.88**	**89.80**	26.23	54.94

(a) MultiHop-RAG

Model	HIT@1	HIT@5	HIT@10	MRR@10
BM25	32.82	53.70	60.92	42.45
bge-large-en-v1.5	55.59	76.58	81.75	64.45
Contriever-msmarco	53.79	76.16	81.69	63.36
E5-mistral-7b-instruct	59.07	80.08	85.28	68.11
GTE-Qwen2-7B-instruct	60.45	80.87	85.72	69.30
ours	**71.22**	**87.41**	**89.97**	**78.14**

(b) NQ

Model	HIT@1	HIT@5	HIT@10
BM25	5.86	26.85	36.42
bge-large-en-v1.5	6.94	27.32	34.11
Contriever-msmarco	6.64	25.77	38.43
E5-mistral-7b-instruct	8.18	28.24	39.82
GTE-Qwen2-7B-instruct	9.11	34.11	49.07
ours	**14.20**	**38.58**	**51.85**

(c) AllSides

Model	HIT@1	HIT@5	HIT@10
BM25	38.70	67.47	77.63
bge-large-en-v1.5	54.14	80.57	86.53
Contriever-msmarco	52.69	80.40	85.79
E5-mistral-7b-instruct	57.32	**85.90**	**88.98**
GTE-Qwen2-7B-instruct	57.65	83.37	88.57
ours	**62.19**	83.78	88.24

(d) AGNews

Table 3.12: Comparisons to Off-The-Shelf Retrieval Models Across Datasets

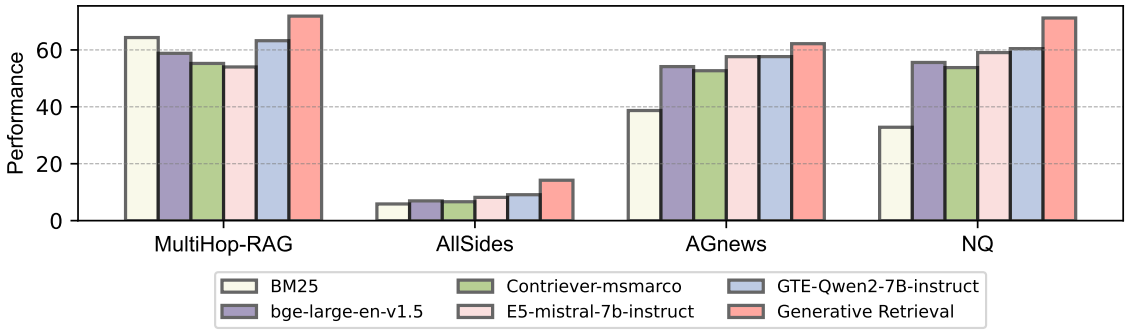


Figure 3.3: Performance comparison between generative retrieval with semantic identifiers and off-the-shelf-retrieval models. We use HIT@4 for MultiHop-RAG and HIT@1 for other datasets as the metric.

3.5.4 Limitations

Our proposed synthetic data strategies focus mainly on the supervised fine-tuning and preference learning stage. But there are also settings that can largely improve the usability of generative retrieval, such as incremental learning or generalization to unseen documents. It is also important to extend the data strategy exploration for these settings. In addition, similar data strategies may also be effectively used to enhance dense retrieval domain adaptation. Further systematic research is needed to investigate the strategies for dense retrieval model fine-tuning, as well as the differences between generative and dense model training.

Our synthetic queries are mainly based on one document. However, queries from the real world may be more complex, such as those involving multiple documents with multi-hop reasoning or multi-evidence comparison. It is still under investigation to generate those complex queries and use those queries during retrieval model training.

3.6 Related Work

Generative retrieval modeling. Previous work has explored different aspects of generative retrieval modeling. One line of research aims to find the appropriate document identifiers for generation, such as numerical or atomic identifier [183, 225, 231], N-grams [21, 28], titles or URLs [27, 87, 100, 232], keywords-based or summary-based semantic identifiers [90, 181], codebook [211, 215, 216], and full passages themselves [179]. There are also efforts to combine the advantages of different identifiers [101]. Another line of work tackles the optimization of generative retrieval, such as using ranking loss [102, 182, 226], or using auxiliary tasks to

enhance generative retrieval training [99]. During retrieval, different constrained decoding methods have been explored to obtain valid identifiers, such as FM-Index [21], Trie-based [26], and set-based inference [182].

Synthetic query generation. Alongside the progress in generative retrieval modeling and optimization, synthetic query generation has emerged as a pivotal technique for enhancing retrieval systems, particularly in domains with limited annotated data. In dense retrieval, synthetic queries have been used extensively to improve cross-domain performance. For example, Ma et al. [123] generated synthetic questions for target-domain documents with a question generation model trained on general-domain data, thereby improving the retrieval performance in zero-shot settings. Similarly, Wang et al. [194] introduced generative pseudo labeling, which combines query generation with pseudo labeling using a cross-encoder to capture finer-grained ranking signals. Further advancements include Bonifacio et al. [23] and Jeronymo et al. [67], which leverage large language models to generate synthetic queries in a few-shot manner and then combine with the top K documents ranked by the conditional question generation probability to train a domain-specific reranker.

Despite the successes in dense retrieval, the potential of synthetic data for generative retrieval has been under-explored. Existing studies often rely on passage-level synthetic queries generated by docT5query [134], following the DSI-QG paradigm [231]. Chen et al. [31] explores breaking documents into text fragments for query generation and memorization. However, there still lacks a comprehensive discussion on useful strategies to build synthetic data for domain-specific corpus, especially with LLMs. This work investigates data strategies from multiple perspectives, including synthetic queries generated using multi-granularity contexts, involving search constraints, and the effects of context data. For preference learning, Zhou et al. [226] proposes using preference learning objectives for generative retrieval with specialized reward models, which is difficult to obtain in a domain-specific setting. Our proposed strategy in preference learning, instead, directly uses the retrieval results to obtain the preference data and is, therefore, more streamlined for domain-specific purposes.

3.7 Conclusion

In this work, we explore several strategies to produce synthetic data for generative retrieval training. We find that adding queries in multi-granularity and queries with domain-specific constraints can largely improve the generative retrieval performance during supervised fine-

tuning, and memorizing document contents can also contribute to the generative retrieval training. We also find that it is critical to choose high-quality hard negative candidates to effectively use the preference learning objectives to further improve generative retrieval.

Query Generation Prompt

```
Your task is to generate a relevant and diverse set of {num_sequences} questions
that can be answered by the provided context.  The questions are to be used by a
retriever to retrieve the article from a large corpus.  Your output should be a
list of unordered questions in Markdown format, where each line starts with dash "-"
followed by the question.

# Context:
{context}

# Output:
```

Figure 3.4: Prompts for query generation.

3.8 LLM Prompts

3.8.1 Prompts for Keywords Generation

Figure 3.7 shows the prompt for generating a series of keywords as the semantic document identifier.

3.8.2 Prompts for Query Generation

Figure 3.4 shows the prompts used to generate various types of synthetic queries, including chunk- and sentence-level queries, constructions-based queries, and question-answer pairs used at the preference learning stage.

Constraints-based Query Generation Prompt

```
Your task is to generate a diverse set of {num_sequences} questions given a context
with metadata.  The generated questions should be answerable by the provided context.
The questions are to be used by a retriever to retrieve the article from a large
corpus.  In addition, the question MUST be composed with at least one metadata
filtering requirement.

# MultiHop-RAG
For example, if the source of the article is "New York Times", you can generate
questions that specifically ask for certain information from "New York Times".  You
should generate questions with different metadata.
# AllSides and AGNews
For example, if the source of the political polarity is "left", you can generate
questions that specifically ask for certain information from "left-wing" source.

DO NOT use "the context" or "the article" in any generated queries or answers.
DO NOT use pronoun "this" in any generated queries or answers.
DO NOT leak any information in this instruction.

Your output should be a list of unordered in Markdown format, where each line starts
with dash "-" followed by the question.  You do not need to provide the answer.

# Metadata
{metadata}

# Context
{context}

# Output:
```

Figure 3.5: Prompts for constraints-based query generation.

Query-Answer Pair Generation Prompt

```
Your task is to generate a relevant and diverse set of less than {num_sequences}
search engine query and answer pairs given a context.
The queries should be similar to what people use with search engine to find the given
context from a large corpus.  The answers are expeced to be a short phrase.
You should make the queries as difficult as possible, but they should be answerable
by the given context.

Do not use "the context" or "the article" in any generated queries or answers.
Do not use pronoun "this" in any generated queries or answers.
Do not leak any information in this instruction.

Your output should be a list of unordered items in Markdown format, where each
item starts with dash "-", followed by "Query:" and the generated query, and then
"Answer:" with the corresponding answer.

# Context
{context}

# Output:
```

Figure 3.6: Prompts for query-answer pair generation.

Keywords Generation Prompt

```
Summarize the following context with meaningful keywords representing different
important information in the context.  Your output should only contain a list of
keywords in Markdown format, where each line starts with the dash "-" followed by the
keywords.

# Context:
{context}

# Keywords:
```

Figure 3.7: Prompt for keywords-based document identifier generation.

Chapter 4

Multimodal Reranking for Knowledge-Intensive Visual Question Answering

In this chapter, we discuss the extension of using generative modeling methods to perform multimodal reranking as the additional knowledge-seeking procedure for knowledge-intensive visual question answering tasks. The reranking module takes multimodal information from both candidates and questions and performs cross-item interaction for better relevance score modeling, and performs one-step decoding to obtain the relevance score. We also discuss the training issues when performing knowledge retrieval and answer generation modules on the same corpus, that there can be potential performance discrepancy between the candidates for answer generation training and testing, and we suggest using noisier candidates during training to obtain a more robust model.

4.1 Overview

Knowledge-intensive visual question answering (KI-VQA), compared to conventional visual question answering, provides questions that cannot be directly answered with images. It requires models to use external knowledge for answer reasoning and synthesis, as shown in Figure 4.1.

A typical KI-VQA system contains a retrieval model to find relevant external knowledge and an answer generator that performs reasoning over the retrieved knowledge to produce the

Q: What US city is associated with this type of pizza?

A: Chicago

Figure 4.1: An example from OK-VQA, which requires knowledge to associate deep-dish pizza and Chicago.

answer. One line of research investigates methods for an effective retrieval pipeline, which includes the choices of knowledge bases [46, 95, 122], and methods for retrieval with visual descriptions [122] or image-text retrieval [50, 109].

Answer generation models usually use retrieval relevance scores to select the top candidates [50, 109]. Although it achieves great success, it may sometimes provide unreliable scores, especially for retrieval using images. Because we usually split an image into a series of image patches and perform retrieval with individual patches, a high relevance score of one patch may not necessarily translate to a high overall question-candidate relevance. In addition, the two-tower architecture of a retriever model also lacks cross-item modeling to predict precise relevance scores.

In this work, we propose including multimodal reranking to improve relevance score modeling, as reranking has already shown its importance in various knowledge-intensive tasks [47, 57, 89, 113, 124, 197]. Multimodal reranking uses the multimodal question and multimodal knowledge items to obtain the relevance score. Specifically, we finetune a pretrained multimodal language model [30] to perform a multimodal cross-item interaction between the question and the knowledge items. We train our reranker on the same dataset as the answer generator training, distantly supervised by checking if answer candidates appear in the knowledge

text. The benefits of this reranking component are twofold. On the one hand, as with other typical reranking components, it can provide more reliable relevance scores by modeling the cross-item interaction. On the other hand, because most existing retrieval models perform unimodal retrieval [50, 109, 122], reranking with multimodal interaction can improve the quality of retrieval by multimodal information from question and knowledge candidates.

We perform experiments on OK-VQA [125] and A-OKQVA [163], based on image-text retrieval [68]. The results show that the distantly-supervised reranker provides consistent improvement compared to the pipeline without a reranker. We also observe a training-testing discrepancy with reranking for answer generation, finding that performance improves when training knowledge candidates are similar to or noisier than testing candidates. As we use the same data for both reranking and answer generation training, the quality of the reranked candidates in the training set will be much higher than the candidates in the test set. Therefore, if we train the model with clean candidates from reranking, it does not generalize to a noisy testing environment. Instead, training with more noisier candidates from initial retrieval can help us obtain a more robust model with noisy candidates so that we can still improve the model with reranked results during testing. We also find that an oracle reranker can provide a promising performance upperbound, which sheds light on future research directions in this area.

4.2 A Knowledge-Intensive Visual Question Answering Framework

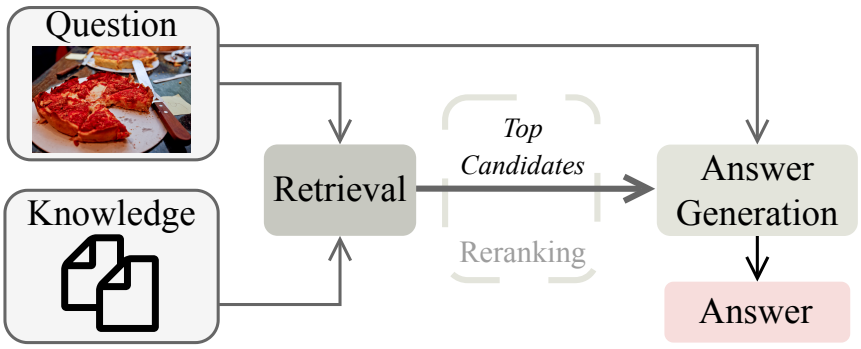


Figure 4.2: A basic KI-VQA framework, which first retrieves relevant top knowledge candidates with using visual question and then combine the question and retrieved knowledge candidates to generate the answer. The dashed box is our reranking module in Section 4.3.

In this section, we will introduce a basic framework for KI-VQA, including image-text retrieval and answer generation, as illustrated in Figure 4.2.

4.2.1 Wikipedia-Based Image Text Dataset

In this work, we use a multi-modal knowledge base, Wikipedia-Based Image Text Dataset (WIT) [172]. In addition to previous work using text from an encyclopedia resource, WIT contains images from Wikipedia and the surrounding text at different levels, including their captions and surrounding sections. Therefore, we consider WIT as a combination of image and text knowledge.

4.2.2 Image-Text Retrieval

Previous work has explored the use of different retrieval model choices [50, 109, 122]. We follow a line of research that adopts image-text retrieval [50] using a pretrained image-text language model with dual encoder architecture [68, 146]. Following Gui et al. [50], we use the sliding window with a stride to generate multiple image regions from the question image. Each image region is considered a query and will be encoded by the image encoder model $\phi_i(\cdot)$. We encode captions in the WIT dataset as representation for candidates using the text encoder model $\phi_t(\cdot)$, as captions in Wikipedia are generally informative. The relevance score between an image region v_i and a WIT candidate c is obtained with the inner product of their representations

$$r_t(v_i, c) = \phi_i(v_i)^T \phi_t(c).$$

4.2.3 Answer Generation

We follow previous work [50, 109], which performs reasoning over the top candidates within an encoder-decoder architecture. We also incorporate multimodal information [159], compared to previous work, which mostly uses text-based information.

Our answer generation module is finetuned on vision language models that take the combination of image and text as input (*e.g.*, 30, 97). We first encode each top candidate separately. The input of each candidate consists of a question image, a candidate image, and text following a template[1] to compose question and candidate. We encode the image with a Vision Transformer [39], which takes a series of image patches $\boldsymbol{x}^v = [x_1^v, \ldots, x_n^v]$, *i.e.*, image tokens, to

[1] `question: <question text> candidate: <caption>`

produce image representations

$$\boldsymbol{E}^v = [\boldsymbol{e}_1^v, \ldots, \boldsymbol{e}_n^v] = Enc_v(\boldsymbol{x}^v).$$

We combine image representations and text token embeddings $\boldsymbol{E}^t$ to produce fused representations with a Transformer [188]

$$\boldsymbol{H} = \left[\boldsymbol{H}_q^v; \boldsymbol{H}_c^v; \boldsymbol{H}^t\right] = Enc_t\left(\left[\boldsymbol{E}_q^v; \boldsymbol{E}_c^v; \boldsymbol{E}^t\right]\right),$$

where $\boldsymbol{E}_q^v$, $\boldsymbol{H}_c^v$ represent the image token representations for question and candidate image, respectively. We also include an empty candidate that consists only of the question image and text.

During decoding, to reduce the total number of representations, we only keep the question image and text representations from the empty candidate and the token representations that correspond to each knowledge caption text. We concatenate these token representations to form a global representation for the decoder to perform cross-attention and generate each answer token autoregressively [64].

4.3 Multi-Modal Reranking

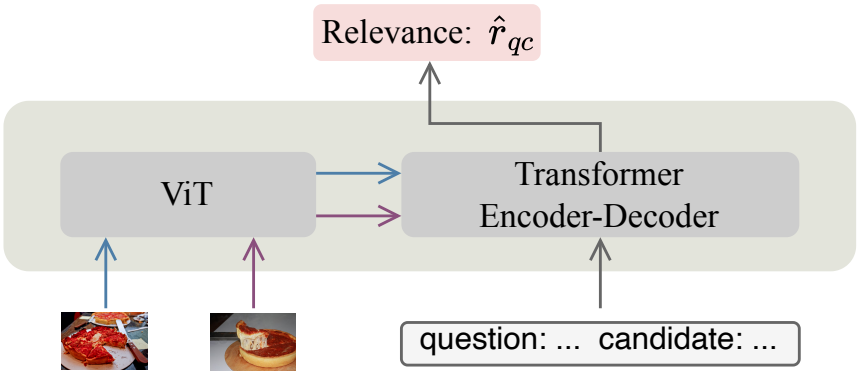


Figure 4.3: Framework of multimodal reranking.

Vanilla retrieval-generation frameworks directly use the relevance score for individual image patches. However, a high relevance for a region does not necessarily imply overall relevance. In this section, we propose multimodal reranking, as illustrated in Figure 4.3, which takes a multimodal question and knowledge as input and produces relevance scores with cross-item interaction.

4.3.1 Modeling

Our ranking model is also finetuned from the multimodal pretrained language model. For each question-candidate pair, we first encode the question and candidate image separately and obtain two series of token representations $\boldsymbol{E}_q^v, \boldsymbol{E}_c^v$. Then we concatenate the two series of image token representations, with text token embeddings $\boldsymbol{E}^t$ following the same template in Section 4.2.3 for a Transformer to produce fused token representations

$$\boldsymbol{H}^r = Enc_r\left(\left[\boldsymbol{E}_q^v; \boldsymbol{E}_c^v; \boldsymbol{E}_q^t; \boldsymbol{E}_t^t\right]\right).$$

We follow Zhuang et al. [230] and use one-step decoding to obtain the score from the unnormalized log-likelihood of a special token "`<extra_id_10>`"

$$\hat{r}_{qc} = Dense\left(Dec\left(\boldsymbol{H}^r\right)\right)_{(\texttt{<extra\_id\_10>})}.$$

4.3.2 Ranker Training

Because we do not have ground-truth relevance scores, we adopt distant supervision labels for reranking training. In a typical VQA setting, each answer consists of 10 candidate annotations of the answer. We count the number of answer candidates that occur in the knowledge candidate text as o. The distantly supervised relevance score is obtained in a way similar to VQA accuracy [13]

$$r_{qc} = \min\left\{o/3, 1\right\}.$$

On OK-VQA, we split the training dataset of the original dataset into sub-training and sub-development sets. In each training step, for a question q, we uniformly sample a candidate set $\mathcal{C}$ from the retrieval results and apply pairwise logistic ranking loss [25], which compares the ranking between all pairs of candidates in the set

$$\ell(q) = \sum_{c\in\mathcal{C}}\sum_{c'\in\mathcal{C}} \mathbb{I}_{r_{qc}>r_{qc'}} \log\left(1 + e^{\hat{r}_{qc'} - r_{qc}}\right).$$

4.3.3 Discrepancy on Applying Reranking for Answer Generation

During answer generation training, it is straightforward to apply the ranking model and use the reranked top candidates as input. However, directly applying reranking on both training and testing will instead harm the model performance. This is because applying the ranker on the training set, from which the ranker is trained, performs much better than when applied to the

unseen test set. As we will illustrate in Section 4.4.5, learning answer generation with higher quality ranking results while testing on lower quality ranking results will, in general, have a negative impact on answer generation performance. Therefore, we will keep the initial retrieval results for answer generation training while using the reranked results for model testing.

4.4 Experiments

4.4.1 Setup

We conduct experiments on OK-VQA [125] and A-OKVQA [163]. OK-VQA introduces visual questions that require external knowledge. A-OKVQA further emphasizes commonsense reasoning over world knowledge. For both datasets, we evaluate the performance on the validation set. Following the standard setting, we use the VQA accuracy as our metric. We use ALIGN [146] for image-text retrieval and use PaLI [30] to initialize (vision and text) Transformers in answer generation and reranking independently. In addition to retrieved knowledge candidates, we also follow REVIVE [109] and use candidates generated from GPT-3. For our model with REVIVE GPT-3, we replace the last 5 candidates of the aggregated candidates with the top 5 GPT-3 generated candidates from Lin et al. [109].

We initialize the image-text retrieval module with a pretrained ALIGN checkpoint, and we initialize both the answer generation and the multimodal reranking module with the PaLI-3b checkpoint.

In the retrieval module, we crop a question image into a series of patches with kernel size 224 with stride 64. We use each image patch to retrieve the top-20 candidates and then aggregate the candidates from one question image. If there are candidates that are retrieved by multiple image patches in the same image, we will keep the one with the highest relevance score. We use aggregated top-20 candidates as candidates set for answer generation training and testing.

For OK-VQA, the multimodal reranker takes 8500 of examples from training set for training, and the rest of them for model development. For each question, the reranker takes aggregated candidates from top-20 image patch retrieval as the candidate set. In each training step, we will sample 20 candidates for each question and perform a pairwise logistic training. We select the reranker checkpoint based on Hits@k. The reranker is then applied to the aggregated image retrieval results to obtain the reranked relevance scores.

The answer generation is trained with batch size 32 for 10K. Reranker is trained with batch size as 8 for 20K steps. The learning rate is 1e-4. We implement the models based on T5X [155].

4.4.2 Results

Methods	VQA Accuracy
BAN+KG [95]	26.7
Mucko [229]	29.2
ConceptBERT [46]	33.7
KRISP [126]	38.9
Vis-DPR [122]	39.2
MAVEx [207]	40.3
KAT [50]	44.3
TRiG [45]	49.4
Our model	**52.6**
models with GPT-3 generated candidates	
PICa [212]	48.0
KAT [50]	53.1
REVIVE [109]	56.6
Our model + REVIVE GPT-3	**57.2**
Our model w/ oracle ranking	64.4

Table 4.1: Results comparison on OK-VQA dataset.

Our results in Table 4.1 and Table 4.2 illustrate the performance compared to some existing work. The results in Table 4.1 show that we provide competitive performance compared to these systems. We also include a comparison for models with GPT-3 [24] generated candidates. We find that our framework can further improve the quality of the answer generation with GPT-3-generated candidates from Lin et al. [109] and outperform these baselines.

We also show that an oracle ranking from distant supervision can provide a promising upper bound, indicating that there is still a large room for future work on improving ranking in this challenge.

Effects of Ranking Methods. We further conduct experiments with different ranking methods to illustrate the performance of multimodal ranking. The results are shown in Table 4.3. We compared variants of our model, including the model that generates an answer directly

Methods	VQA Accuracy
ViLBERT [119]	30.6
LXMERT [177]	30.7
KRISP [126]	33.7
GPV-2 [73]	48.6
Our model	**51.6**

Table 4.2: Results comparision on A-OKVQA dataset.

Methods	VQA Accuracy	
	OK-VQA	A-OKVQA
No retrieval	50.6	50.4
+ Image Retrieval	52.1	50.3
+ Multimodal Reranking	**52.6**	**51.6**

Table 4.3: Effects of multimodal reranking, compared to model without retrieval and model without reranking.

without external knowledge and the model with initial image retrieval without further reranking. We can find the steady improvement brought by multi-modal reranking on both datasets. We provide additional comparison to other reranking strategies in Section 4.4.3 and zero-shot multi-modal large models in Section 4.4.4 that are not instruction tuned on OK-VQA.

4.4.3 Additional Comparison with Other Ranking Strategies

Ranking Methods	VQA Acc.
Distillation [63]	51.5
RankT5 [230]	52.3
Reranking	**52.6**

Table 4.4: Effects of multimodal ranking. We can find that learning reranker using distillation from the answer generator can instead hurt the performance. Our multimodal reranker trained with small data provides competitive performance compared to RankT5, which is pretrained on a large amount of data.

We also compare our model to the same multimodal reranking model architecture trained with knowledge distillation from answer generation [63] and RankT5 [230] in Table 4.4. We find that knowledge distillation does not provide reliable supervision to train a reasonable reranking module. Both text-based reranking and multi-modal reranking can contribute to the performance, while multi-modal reranking provides better performance. Especially, compared to RankT,5 which is pretrained with more than 500K items, our reranker is only trained with around 8000 items, and it still achieves competitive performance.

4.4.4 Comparison With Zero-Shot Multi-Modal Large Models

We also provide an additional comparison in Table 4.5 between some large multimodal models on OK-VQA, including Flamingo-80b [5] and BLIP-2 [97]. We report their zero-shot performance compared to our model. The results show that the smaller model can still achieve competitive performance compared to the zero-shot capability of those large models. We also note that there are some other multimodal large models such as LLAVA 1.5 [111], MiniGPT4-V2 [29], which are instruction tuned with OK-VQA and therefore cannot be directly compared. But in

Methods	VQA Acc.
BLIP-2 [97]	45.9
Flamingo-80b [5]	50.6
Our model	**52.6**

Table 4.5: Comparison between multi-modal large models on OK-VQA datasets. We can find that our model provides promising performance compared to the zero-shot performance of those large multimodal models.

general, our proposed framework can be extended to other multi-modal language models that take the combination of image and text input.

4.4.5 Training and Testing Discrepancy

Source of Candidates			VQA
Train	Test	Discrepancy	**Accuracy**
Retrieval	Retrieval	$\rightarrow$	52.1
Reranking	Reranking	$\searrow$	50.7
Retrieval	Reranking	$\nearrow$	52.6
Oracle	Oracle	$\rightarrow$	64.4
Oracle	Retrieval	$\searrow$	47.2
Retrieval	Oracle	$\nearrow$	59.7
Retrieval	Retrieval	$\rightarrow$	52.1

Table 4.6: Effects of discrepancy between knowledge candidates for training and testing. $\rightarrow$ means the qualities of knowledge candidates in training and test are similar. $\searrow$ means the quality in training is better than test. $\nearrow$ means the quality in test is better than training.

As we discussed in Section 4.3.3, directly applying a trained ranking model on both training and testing will hurt the performance. We further illustrate it empirically in Table 4.6. We find that if the model is trained on higher-quality candidates while applied to lower-quality candidates, we will observe a drastic performance drop. In contrast, when the quality in testing is better than in training, we can still find steady improvement. This phenomenon indicates that

an answer generator trained with higher-quality data cannot effectively conduct knowledge reasoning on noisier data, and therefore, we should train the model with noisier data.

4.4.6 Limitations

In this chapter, we focus on applying multimodal reranking to KI-VQA. However, because of the nature of visual data, directly adding visual information may significantly increase the input size, and we will require more total memory to train the model. In this chapter, to reduce the total memory use, we have a much smaller number of knowledge candidates for reasoning in the answer generation module compared to previous work, which only uses text-based knowledge candidates. Nevertheless, it is still important to further investigate more efficient ways to incorporate visual information.

Although multimodal reranking achieves promising performance on knowledge-intensive visual question answering, it is still an open question whether multimodal reranking can be used to help other vision-language tasks. In addition, it is also important to develop a benchmark to systematically evaluate multimodal reranking models, which is not covered in this work.

Similarly, in this work, we only use ALIGN and PaLI as the pretrained model for retrieval, reranking, and answer generation. Although it is natural to extend the framework in this work to other pretrained models, it is still interesting to see how it contributes to different (large and small) models. We provide some preliminary results comparing our reranking pipeline with zero-shot large multimodal models [5, 97] in Appendix 4.4.4, but we also notice that some work [29, 111] uses OK-VQA as instruction tuning data, making it hard to compare/be adopted directly.

We also notice that there is another line of research investigating how to effectively use large language models for knowledge-intensive visual question answering [50, 109, 159, 165, 212]. Although our preliminary results show that our framework can still provide additional improvements over using the same candidates generated by the large language model as in Lin et al. [109], it is still an open question to effectively use and combine the retrieval pipeline and queries or candidates from the large language model.

4.5 Related Work

A typical knowledge-intensive visual question answering model involves a knowledge retrieval to find relevant information, and answer generator to produce the answer [46, 50, 95, 109,

122, 159, 165, 212]. Previous work on knowledge-intensive visual question answering explores knowledge bases in different modalities, such as text items [50, 122], graph items [46, 95], and the composition of image items and text items [207]. Our work differs from previous work by involving multi-modal knowledge items as the knowledge base, where each item contains both image and text information.

There is also a line of research investigating answer reranking, where they first produce a list of answer candidates and then rerank those candidates to obtain the most reliable answer [126, 168, 207]. Instead, the focus of our work is to first retrieve a set of knowledge candidates that can help answer generation and then improve the quality of the knowledge candidate set through multimodal knowledge candidate reranking. Those selected candidates will still serve as additional knowledge input for answer generation reasoning.

4.6 Conclusion

In this chapter, we introduce reranking, a critical stage for knowledge-intensive tasks, into KI-VQA. Our multimodal reranking component takes multi-modal questions and knowledge candidates as input and performs cross-item interaction. Experiments show that our proposed multimodal reranking can provide better knowledge candidates and improve the answer generation accuracy. We also investigate the principle when applying the reranker during the answer generation training and testing. We find that clean candidates for answer generation training can make models more fragile and less effective in leveraging knowledge candidates during testing that may contain some noise, while incorporating noisier knowledge candidates during training enhances model robustness.

Part III

Knowledge-Enhanced Inference: Applications on Social Content Analysis

Chapter 5

Knowledge-Enhanced Topic Representation: Case Study on Text and Multimodal Social Content Analysis

In this chapter, we discuss performing generative modeling-based social content analysis with additional input from external knowledge related to the topic. Specifically, we first address the zero-shot and few-shot stance detection problem that identifies the polarity of text with regard to a certain target when we have only limited or no training resources for the target. In this chapter, we instead utilize a conditional generation framework and formulate the problem as denoising partially filled templates, which can better utilize the semantics among input, label, and target texts. We further propose to jointly train an auxiliary task, target prediction, and to incorporate manually constructed incorrect samples with unlikelihood training to improve the representations for both target and label texts. We use the target as a query to obtain Wikipedia knowledge related to the target and verify the effectiveness of target-related Wikipedia knowledge with the generation framework. We also extend the analysis into multimodal settings where we focus on hateful meme detection. We first use a multimodal retriever to find relevant multi-modal knowledge with the meme and learn to decode rationale before making the prediction.

Input Text: Airports and the roads on east nor west coast can not handle the present volume adequately as is. I did ride the vast trains in Europe, Japan and China and found them very comfortable and providing much better connections and more efficient.	
Target: high-speed rail	**Stance Label:** Supportive (Pro)

Table 5.1: A stance detection example from VAST.

5.1 Overview

Stance detection is an important task that identifies the polarity of the text with regard to certain targets [7, 15, 130, 170, 171], as shown in Table 5.1. It is crucial to understand opinionated information expressed in natural language and can facilitate downstream social science analyses and applications [54, 66, 219].

Previous work on stance detection focuses mainly on in-domain or leave-out targets with only a few target choices [9, 48, 71, 104, 131, 209, 218]. Although achieving promising performance, these models are limited in generalizing to a wide variety of targets. Zero-shot and few-shot stance detection on varied topics (VAST; Allaway and McKeown, 2020), instead, provides a diverse set of targets for training and testing. In this setup, the target can be anything related to the context, and we do not expect that the target phrase will explicitly appear in the context. Efforts in this direction include graph modeling [110], common sense knowledge [56, 112] and contrastive learning [105, 106]. These methods generally formulate the problem into a classification setting, which directly trains the label representation from scratch and does not fully utilize the semantics from those labels and target texts.

However, connections among text semantics from input text, target, and label can be beneficial for stance detection. In this chapter, we propose a new model by formulating the problem as a denoising task to generate label text from text templates via conditional generation. Compared to direct classification, generation-based frameworks give us the flexibility to further exploit the label and topic semantics via learning to decode a series of natural language texts containing the predicted label. The flexible generation method also allows us to incorporate auxiliary tasks to further improve the modeling. To improve target representation, we propose to jointly train target prediction with stance detection, which gives the input text and desired stance label to output possible targets. We use unlikelihood training [199] that suppresses the likelihood of manually constructed incorrect samples to enhance label representations. To fur-

ther enhance the representation of the target, we also use retrieved knowledge from Wikipedia pages using the target as the query [56] and consider them as additional Wikipedia knowledge input that is related to the target in our generation model.

In addition to text-based analysis, we also extend the knowledge incorporation paradigm to multimodal social content analysis. We focus on hateful meme detection, which aims to identify hateful speech from a meme that contains both text and visual information. We mostly use the same principle as text-based analysis for stance detection, except that we are using the multi-modal language model as the base model for analysis. We use the Wikipedia-based Image Text Dataset (WIT [172]) as the external knowledge corpus and use a multi-modal retriever [108] to find relevant multi-modal knowledge candidates. To further enhance the reasoning capability of the model with the knowledge retrieved, we ask the model to generate a rationale given the meme, label, and retrieved knowledge. During training, the model first learns to generate the rationale with the mem and retrieved knowledge and then makes a final prediction.

We evaluate our stance detection method on VAST. Experimental results show that the conditional generation formulation can achieve better performance compared to classification, demonstrating the effectiveness of connecting input, target, and label semantics for stance detection. Further analysis illustrates the benefits of joint target prediction, unlikelihood training, and Wikipedia knowledge. Our model can achieve new state-of-the-art performance, outperforming several strong baselines from previous work. We evaluate the multimodal hate speech detection on the Hateful Memes Challenge. The experiment result shows poor zero-shot performance using Qwen2-VL-2b-Instruct, while task-specific fine-tuning significantly improves it. We also show that adding external knowledge and explicit modeling of reasoning over the knowledge further help improve the performance.

5.2 Approach

5.2.1 Problem Formulation

Stance detection aims to identify the polarity of an input text with regard to a specific target. Formally, a sample instance can be considered as a triple $(\boldsymbol{x}, \boldsymbol{t}, y)$, where $\boldsymbol{x}$ and $\boldsymbol{t}$ are two sequences of tokens, representing input text and target, respectively. $y \in \{\text{supportive (pro)}, \text{opposite (con)}, \text{neutral}\}$ represents then stance label.

A stance detection model is to infer the stance label y given $\boldsymbol{x}$ and $\boldsymbol{t}$ with parameter θ:

$$f(\boldsymbol{x}, \boldsymbol{t}; \theta) = y.$$

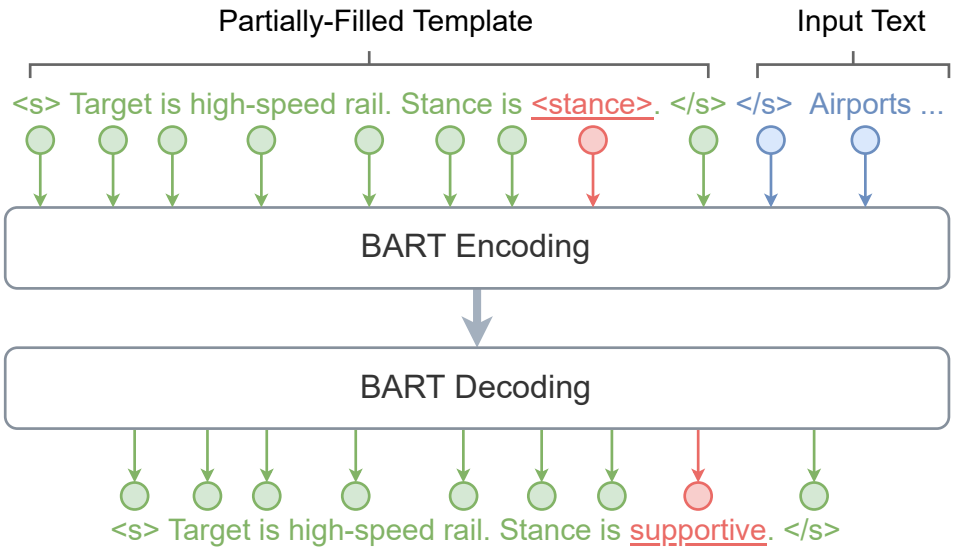


Figure 5.1: Overall framework of BART-based generation framework for stance detection.

In the zero-shot and few-shot stance detection dataset with varied targets [7], many target tokens occur only zero or a few times in the training set.

5.2.2 A Generation-Based Framework

Generation-based frameworks have demonstrated their effectiveness for problems beyond traditional generation tasks [93, 98, 149, 210]. We use a conditional generation model for this problem, where the condition is a partially filled template with the input text. The template is two sentences describing the target and stance with a `<stance>` placeholder for stance detection. An example of a partially filled template with input text and output is shown in Figure 5.1.

Our base model is BART [94], an encoder-decoder language model pretrained with denoising objectives, which is similar to our generation-based formulation. The generation process can be considered as using the conditional probability to select a new token at each step given input and previously generated tokens:

$$p\left(\boldsymbol{o} \mid g\left(\boldsymbol{x}, \boldsymbol{t}\right); \theta\right) = \prod_{i=1}^{|\boldsymbol{o}|} p\left(o_i \mid \boldsymbol{o}_{<i}, g\left(\boldsymbol{x}, \boldsymbol{t}\right); \theta\right),$$

where $g\left(\boldsymbol{x}, \boldsymbol{t}\right)$ represents the transformation function that fills the target $\boldsymbol{t}$ in the template and forms the input sequence with the input text $\boldsymbol{x}$. Specifically, $g\left(\boldsymbol{x}, \boldsymbol{t}\right)$ will generate a combination of input text and template with special tokens: "`<s>` template `</s></s>` $\boldsymbol{x}$ `</s>`". The template

contains two sentences: "The target is <target>. The stance is <stance>". We will fill in <target> placeholder with the actual target and keep the <stance> placeholder for the decoder to generate.

The generated output $\boldsymbol{o}$ is a fully filled template, where both the target and the stance placeholders are replaced by actual or predicted values. The model is trained by minimizing the log-likelihood over the whole generated sequence:

$$\begin{aligned}\mathcal{L}_s &= -\log p\left(\boldsymbol{o} \mid g\left(\boldsymbol{x}, \boldsymbol{t}\right); \theta\right) \\ &= -\sum_{i=1}^{|\boldsymbol{O}|} \log p\left(o_i \mid \boldsymbol{o}_{<i}, g\left(\boldsymbol{x}, \boldsymbol{t}\right); \theta\right).\end{aligned}$$

The final predicted stance label is obtained with a post-processing function that tries to find the polarity word after the prompt for stance.

Joint Target Prediction

Another advantage of using generation-based architecture is that we can leverage auxiliary generative tasks to help train stance detection. We use target prediction, which is to infer target tokens $\boldsymbol{t}$ given the stance label y and the input text $\boldsymbol{x}$:

$$f_t\left(\boldsymbol{x}, y; \theta\right) = \boldsymbol{t}.$$

Target prediction can provide the connection of stance to target in the opposite direction of stance detection. It can also improve the representation of target tokens by learning to decode them.

The input sequence of the target prediction is similar to that of stance detection, consisting of a partially filled template and input text. The template used for joint target prediction is slightly different than the one used for stance detection, where we switch the position of two sentences so that the stance information shows up first. We will fill in the actual stance text in the input sequence and leave the <target> placeholder for the decoder to generate.

Unlikelihood Training

Log-likelihood objective optimizes the likelihood over the entire distribution. However, in our task, especially when generating the stance labels, we should specifically focus on several candidate tokens. Therefore, we introduce unlikelihood training [199], where we use unlikely tokens,

Stance Detection	
Input	Target is high-speed rail. Stance is `<stance>`.
Output	Target is high-speed rail. Stance is supportive.
Target Prediction	
Input	Stance is supportive. Target is `<target>`.
Output	Stance is supportive. Target is high-speed rail.
Unlikelihood Training	
Input	Target is high-speed rail. Stance is `<stance>`.
Output	Target is high-speed rail. Stance is opposite.

Table 5.2: Examples input and output templates for stance detection, target prediction, and unlikelihood training.

i.e. incorrect stance predictions, to replace the ground-truth sequence and optimize with the unlikelihood loss for the replaced tokens.

Specifically, for an output sequence $\boldsymbol{o}$, we assume that o_k is the stance label and replace it with an incorrect stance prediction o'_k while keeping other tokens to form an incorrect sequence $\boldsymbol{o}'$. The combination of likelihood and unlikelihood will be:

$$\begin{aligned}\mathcal{L}_u = \quad & \log p\left(o'_k \mid \boldsymbol{o}'_{<k}, g\left(\boldsymbol{x}, \boldsymbol{t}\right); \theta\right) \\ & - \sum_{i \neq k} \log p\left(o'_i \mid \boldsymbol{o}'_{<i}, g\left(\boldsymbol{x}, \boldsymbol{t}\right); \theta\right),\end{aligned}$$

For each ground-truth sequence, we can construct two sequences for unlikelihood training with the other two incorrect stance labels. Table 5.2 illustrates the examples for different input and output templates for stance prediction, target prediction, and unlikelihood training. Similarly, we can use the same strategy for unlikelihood training on target prediction, in which we alter the stance label and train the model to reduce the likelihood of producing the topic words.

Incorporating Wikipedia Knowledge

He et al. [56] collect relevant Wikipedia snippets for each target and propose to incorporate Wikipedia knowledge to enhance target representations for BERT-based [38] classification, which demonstrates a significant improvement. We follow He et al. [56] and incorporate Wikipedia knowledge into our generation-based method. Specifically, we append Wikipedia snippets to the end of our input sequence: "<s> template </s></s> $\boldsymbol{x}$ </s></s> Wikipedia snippet </s>".

We use the new input sequence to perform both training and inference, while the output sequences remain as the fully-filled templates.

Training Objective

The final training objective is the combination of loss functions from stance detection, target prediction, and unlikelihood training:

$$\mathcal{L} = \mathcal{L}_s + \alpha_t \mathcal{L}_t + \alpha_u \mathcal{L}_u,$$

where $\mathcal{L}_t$ represents the log-likelihood loss over the output template for target prediction, and α_t, α_u are used to balance different loss functions.

5.3 Extension to Multi-Modal Analysis

For multimodal analysis, we choose a problem similar to text-based stance detection, hateful memes detection. This task is to analyze whether the given meme contains hateful speech, which often requires additional background knowledge to fully comprehend the context. We follow the same method as the text-based stance detection framework to include Wikipedia knowledge in the analysis. In addition, we also ask the model to learn to decode the corresponding rationale before generating the actual analysis output.

5.3.1 Basic Generation Framework

We also use the generation-based framework as the text-based analysis. Unlike the BART-based encoder-decoder framework, we choose Qwen2-VL [196] series for our analysis, which is a decoder-only framework. We use Qwen2-VL-2b-Instruct as the base model for fine-tuning. The generation process can still be considered using the conditional probability to select a new token at each step

$$p\left(\boldsymbol{o} \mid g_v\left(\boldsymbol{v}\right); \theta\right) = \prod_{i=1}^{|\boldsymbol{o}|} p\left(o_i \mid \boldsymbol{o}_{<i}, g_v\left(\boldsymbol{v}\right); \theta\right),$$

where v is the input meme and $g_v(\cdot)$ represents a function to wrap the input meme into an instruction with chat format [1].

[1]The instruction text is "Detect whether the given meme is hateful or not. Your output should only have one line, either "Hateful" or "Not Hateful"."

5.3.2 Knowledge Retrieval and Incorporation

Similar to Chapter 4, as we are tackling multi-modal analysis, we use a multi-modal knowledge base, the Wikipedia-based Image Text Dataset (WIT [172]). We use MM-Embed [108] as the retriever, which allows us to include multi-modal information during retrieval. For each knowledge item in WIT, we use MM-Embed to obtain an embedding from its English caption text. During retrieval, we encode the combination of the meme and an instruction [2] as the query to retrieve the top knowledge candidate from Wikipedia. For simplicity, we also only consider the top-1 candidate when performing the analysis.

5.3.3 Obtaining and Training with Rationales

Because much hateful speech in memes is expressed in a fairly implicit way, it is natural to explicitly model the reasoning or the thinking process with the given external background knowledge for the analysis. Because external knowledge is obtained from retrieval, we do not have any ground-truth reasoning context. Therefore, we choose to obtain the reasoning context by asking the vision language model to provide the rationale with a given meme and whether it is hateful speech or not as input. The specific system prompt is illustrated in Figure 5.2.

Rationale Generation Prompt

```
You are expert on detecting and analyze hateful speech.  You will be given a meme,
retrieved Wikipedia image and caption with the meme, and whether the meme is hateful
or not.  You need to provide me with the rationale that why it is hateful or not.
The rationale should also try to use the knowledge from the Wikipedia image and
caption.
```

Figure 5.2: Prompts for rationale generation.

Because we would like to generate as accurate a rationale as possible, we use a larger model, Qwen2-VL-72B-Instruct, to generate the rationale. During training, given a meme, we ask the model to first generate the rationale, then provide the prediction. Specifically, the generation process can be formulated as

$$p\left(\boldsymbol{o} \mid g_k\left(\boldsymbol{v}, \boldsymbol{k}_v\right); \theta\right) = \left(\prod_{i=1}^{|\boldsymbol{l}|} p\left(l_i \mid \boldsymbol{l}_{<i}, \boldsymbol{r}, g_v\left(\boldsymbol{v}, \boldsymbol{k}_v\right); \theta\right)\right) \left(\prod_{i=1}^{|\boldsymbol{r}|} p\left(r_i \mid \boldsymbol{r}_{<i}, g_v\left(\boldsymbol{v}, \boldsymbol{k}_v\right); \theta\right)\right),$$

[2] The instruction text is "Given a meme, retrieve a Wikipedia image caption pair that provides external knowledge to understand the meme."

where $\boldsymbol{k}_v$ represents the WIT knowledge candidate retrieved, $\boldsymbol{r}$ represents the rationale, $\boldsymbol{l}$ represents the predicted label, and $g_k(\cdot)$ represents a function to wrap the input meme and the retrieved knowledge in an instruction in chat format [3].

5.4 Experiments

5.4.1 Data

VAST contains 18,548 examples from the *New York Times* "Room for Debate" section with 5,630 different targets for zero-shot and few-shot stance detection. Original examples of VAST are collected from Habernal et al. [53] under Apache-2.0 license[4]. We use Wikipedia knowledge collected by He et al. [56], which uses the API to crawl Wikipedia pages for targets. Wikipedia content can be used under the Creative Commons Attribution Share-Alike license (CC-BY-SA)[5]. We use the same training/development/test split as Allaway and McKeown [7].

We use the Hateful Memes Challenge [76] as the dataset for our multimodal analysis experiment and follow the standard training and testing splits.

5.4.2 Experimental Setup

We conduct our experiments on VAST [7] for text-based analysis. We compare our model with several existing systems, including 1) TGA-Net [7]; 2) BERT-GCN [110]; 3) CKE-Net [112]; and 4) WS-BERT [56]. Following their setup, we use the macroaverage F_1 as the evaluation metric, and we report performance on the subset of test set for zero shot, the subset for few shot, and the overall test set. We use BART-base[6] as our base model, of which the number of parameters is roughly consistent with baselines on BERT-base[7]. Our best model is optimized with AdamW [118] for 30 epochs with a learning rate of 1e-5. We use a linear scheduler with a warmup proportion of 0.1, and the training batch size is 32. We use a greedy search during inference. We report on performance in development set and test set using the averaged results

[3]The instruction text is "Detect whether the meme is hateful or not. Your output should have two lines. The first line starts with "Rationale:" and then provide the rationale for the detection. The second line starts with "Results:" follow by only either "Hateful" or "Not Hateful"."

[4]https://github.com/UKPLab/argument-reasoning-comprehension-task/blob/master/LICENSE

[5]https://en.wikipedia.org/wiki/Wikipedia:Reusing_Wikipedia_content

[6]https://huggingface.co/facebook/bart-base

[7]https://huggingface.co/bert-base-uncased

of 5 different random seeds. The test results are reported based on the best overall F_1 performance on the development set. α_t is set to 1 and α_u is set to 0.5. Our final model takes about 5 hours to train on one Nvidia RTX 3090 GPU.

We conduct our experiments with the Hateful Memes Challenge for multimodal analysis. We mainly compare with some model variants, including zero-shot inference, vanilla finetuning, vanilla finetuning with retrieved knowledge, and finetuning with retrieved knowledge and rationale generation. For training with rationale, we first generate 5 different rationales for each meme in the training set with temperature at 0.2. Then, we use one rationale for a meme as the training target for hateful meme detection training in one training epoch and iterate different rationales at different epochs. We optimize the model with AdamW for 10 epochs with a learning rate of 2e-5 and batch size 64. We take the final checkpoint for the evaluation. The maximum gradient norm is set as 0.3, and the warmup ratio is set as 0.03. The model is trained with bfloat16 and TensorFloat-32 on a single machine with 8x Nvidia RTX A6000.

5.4.3 Text-Based Results

Comparing with Model Variants

We first conduct a comparison of some of our model variants to illustrate the effectiveness of our proposed components. The results are shown in Table 5.3. From the comparison of BERT-based classification (BERT Classification) and BART-based denoising generation from templates (BART w/ Template), we can find that adopting the generation framework can significantly improve the model performance. Our proposed topic prediction and unlikelihood training can further boost performance. The final model with Wikipedia knowledge verifies the effectiveness of Wikipedia knowledge for stance detection with a generative framework.

Comparing with Existing Systems

Our overall performance is shown in Table 5.4. Our method can significantly outperform those previous baselines, indicating the effectiveness of our proposed generation framework for zero-shot and few-shot stance detection with varied topics.

5.4.4 Qualitative Analysis

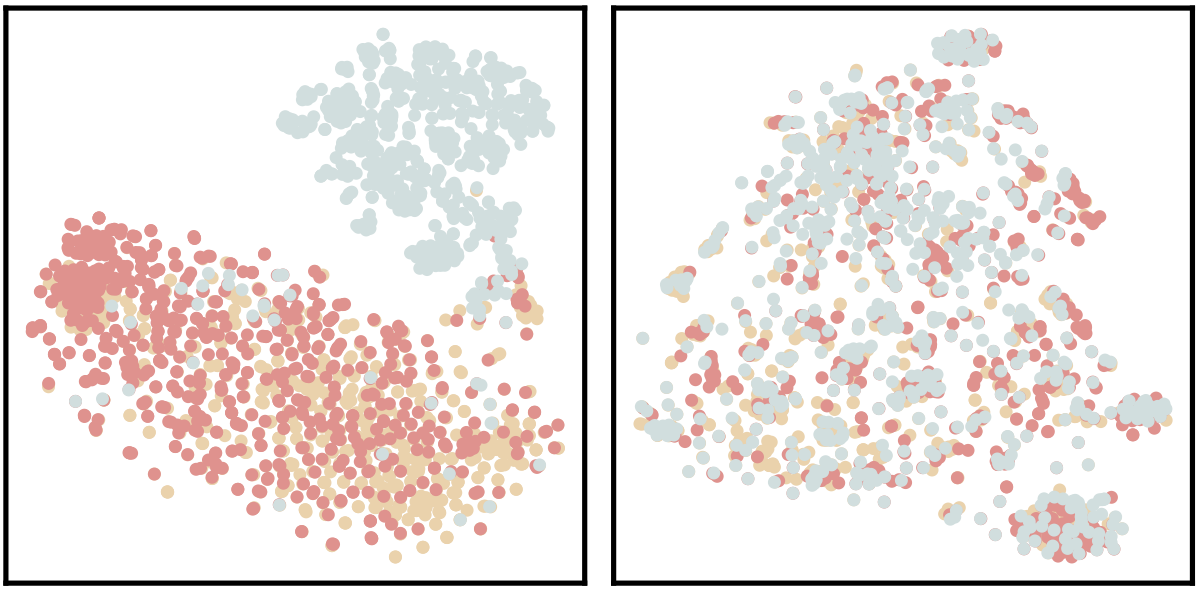
Figure 5.3 shows the t-SNE [187] visualization of intermediate representations before the classification layer of our model and the BERT classification model on the development set. We

Model	Precision	Recall	F_1
BERT Classification	72.6	72.0	72.1
BART w/ Template	75.7	75.1	75.3
+ Topic Prediction	76.0	75.6	75.7
+ Unlikelihood	76.4	75.9	75.9
+ Wikipedia	**78.0**	**77.3**	**77.4**

Table 5.3: Performance of different model variants on the overall precision, recall, and F_1 on the development set (%). Each of our model variants is on top of the variant from its previous row.

Model	Zero-Shot	Few-Shot	Overall
TGA-Net	66.6	66.3	66.5
BERT-GCN	68.6	69.7	69.2
CKE-Net	70.2	70.1	70.1
WS-BERT	75.3	73.6	74.5
Our Model	**76.4**	**78.0**	**77.3**

Table 5.4: Stance detection performance (%) on VAST. Our model significantly outperforms previous work on all metrics. Our results are obtained from averaging performances over 5 random seeds. $p < 0.001$ on overall F_1 using Z-test with variance as the standard deviation over multiple runs.



(a) Our model (b) BERT classification

Figure 5.3: The t-SNE visualization of intermediate representations from our model and BERT classification model. Color map: Supportive, Opposite, Neutral.

use random initialization with perplexity as 50 for visualization, and we color each visualized instance with its corresponding stance label. The visualization of BERT classification shows small clusters with hybrid labels, while we can see that instances with our generation method are clustered with labels, where neutral labels are at the top and supportive labels are generally at the bottom.

5.4.5 Multimodal Analysis Results

We conduct some comparisons with model variants to illustrate the effectiveness of our proposed methods. The results are shown in Table 5.5. From the results, we find that the vanilla zero-shot inference cannot handle this detection task and produces pretty random detection outputs. Supervised fine-tuning with labeled data in the Hateful Memes Challenge can significantly improve performance. We can also find that adding the retrieved knowledge from WIT further improved the performance, and we can have additional benefits by explicitly modeling the reasoning path with the retrieved knowledge.

Model Variant	F_1	Accuracy
Zero-Shot	55.35	57.13
Fine-Tuning	68.71	76.05
Fine-Tuning w/ Knowledge	71.17	77.80
Fine-Tuning w/ Knowledge + Rationale	72.89	78.80

Table 5.5: Performance of different model variants on the overall F_1 and Accuracy on the test set (%).

5.4.6 Limitations

Because of the nature of our framework design, the stance detection work requires a diverse set of targets during training, which is important for target prediction and, therefore, the stance detection method. It is difficult to apply to other stance detection datasets when there are limited training resources with regard to targets, such as Conforti et al. [34] and Mohammad et al. [130]. Besides, the model is trained on news-related debate corpus, so it may need further domain adaptation if applying the model to other domains such as social media.

We are using an autoregressive generation framework, which will also require extra inference time to generate the whole output sequence compared to the classification model. We encourage readers to compare it with classification methods for efficiency when it is applied in a time-sensitive scenario.

5.5 Related Work

Zero-shot and few-shot stance detection. Zero-shot and few-shot stance detection focus on detecting stances for unseen or low-resource targets. Allaway and McKeown [7] construct a dataset with varied topics that can be used to test stance detection under zero-shot and few-shot settings. Previous efforts have mainly focused on modeling targets, documents, or their connections. Allaway and McKeown [7] obtain generalized topic representation through clustering. Liu et al. [112] use a commonsense knowledge graph to enhance the connection between the target and the document. Liang et al. [105, 106] use contrastive learning to learn the target features. He et al. [56] incorporate Wikipedia knowledge to enhance target representations. In our work, we use a conditional generation framework to build connections between input, target, and label text semantics.

Text processing via conditional generation. Our work is also motivated by recent success in tackling text processing problems using conditional generation [94, 149]. In addition to conventional text generation problems, conditional generation frameworks are effectively applied in information extraction [98], question answering [93, 149], sentiment analysis [210], and general zero-shot text classification problems [82]. In our work, we specifically explore stance detection via conditional generation and reveal several advantages, including fitting the pretraining paradigm, the flexibility of using auxiliary tasks, and the use of external knowledge.

5.6 Conclusion

In this chapter, we propose a generation-based framework for zero-shot and few-shot stance detection that generates a stance label from pre-defined templates. We further propose an auxiliary task, joint target prediction, which takes stance label and input text to generate targets and unlikelihood training on manually constructed incorrect generation output. Combined with Wikipedia knowledge for target from He et al. [56], our model can achieve a new state-of-the-art performance on VAST. We also extend the incorporation of external knowledge into a multi-modal analysis problem, hateful meme detection, and find that the use of external knowledge can also help improve the analysis performance.

Conclusion

Conclusion

Summary of Dissertation Findings

This thesis systematically investigates various approaches to enhance generative modeling-based social content analysis with external or prior knowledge. This thesis consists of three parts. The first part discusses methods for converting external knowledge or prior knowledge into resources for model training. We show that it is possible to convert structured knowledge such as WikiData to distantly supervised data so that we can extend the analysis with limited attribute types to open-domain analysis. We also show that prior knowledge can be converted into a set of training constraints that can be included to enhance model training with an additional constrained loss term. In the second and third parts, we discuss methods to augment model inference with external knowledge. The second part focuses on discussing generative modeling-based methods to develop the retrieval pipeline that can help identify useful knowledge candidates. We investigate synthetic data strategies so that we can train an effective generative retrieval on a corpus of which we do not have any training query-document pairs. We also discuss the training and the use of a multimodal reranker on the question answering training set used for answer generation training, without explicit annotation on gold standard knowledge candidates. The last part shows an example of applying retrieved knowledge for a social content analysis problem, zero-shot and few-shot stance detection, with a generative modeling-based architecture. We also extend the overall analysis pipeline with another example on multi-modal space, hateful meme detection. Through these investigations, we can summarize several key insights on building knowledge-enhanced social content analysis models with generative modeling:

Generative modeling demonstrates strong capabilities in social analysis and knowledge retrieval problems. In this thesis, we largely adopt generative modeling in many of the building blocks of our social content analysis problems. We show that the flexibility of

the generation-based paradigm can help us more easily handle zero-shot or few-shot social analysis problems or problems that require not only extraction but also inference, such as the open-domain profile inference and the zero-shot or few-shot stance detection. We can also use the generation-based paradigm for domain-specific retrieval by learning to decode designated document identifiers for relevant documents. We can also effectively use the decoder to predict a reranking relevance score with one-step decoding.

Knowledge can be involved at multiple stages to enhance the social content analysis. This thesis is organized into three different parts, illustrating the use of knowledge in different stages. In Part I, we discuss methods for converting a structured knowledge base into synthetic training data and converting prior knowledge into constraints for model training. In Part II, we discuss the use of the knowledge base itself to build a domain-specific retrieval model, and we discuss combining the knowledge base with the question-answering dataset to create distant supervision for reranker training. In Part III, we show an example of applying retrieved knowledge to enhance social content analysis. These studies represent several key stages of social content analysis and illustrate the effectiveness of incorporating external or prior knowledge into each of the stages.

Synthetic data are usually the essential building blocks. To exploit the usefulness of external or prior knowledge, we largely adopt methods to create synthetic data or variant methods to automatically create training signals. We use the combination of WikiData and Twitter information for the open-domain profile inference study. We apply prior knowledge to create consistency-constrained learning objectives. We adopt LLM-based query generation to create a large set of synthetic queries for generative retrieval training. We select positive candidates based on distant supervision for reranker training and negative candidates based on initial retrieval results for retrieval preference learning and reranker learning. Those synthetic data are essential for training each of the components when building social content analysis pipelines.

The quality of (synthetic) data matters. On the other hand, the construction of the synthetic data itself is usually not trivial. In this thesis, we mainly adopted two principles regarding the quality of the synthetic data. The first principle is that we should create synthetic data that closely follows the testing setup. In generative modeling training, we show that there is a clear correlation between the Jaccard Similarity to the test set and the retrieval performance. In multimodal reranking training, we show that we should use noisier candidates for model training so

that we can have a more robust model to offset the potential retrieval errors during inference. The second principle is that we should create challenging synthetic data, as the simple synthetic data may have an adverse impact on the model training, as we illustrated in the negative candidate selection for generative retrieval preference learning.

Future Directions

In this thesis, we have demonstrated examples of utilizing knowledge to enhance social content analysis at different stages. We conduct those studies on a case-by-case basis, where we first identify the analysis problem, investigate potential knowledge content that can be applied to this problem, and appropriate methods to combine knowledge and the analysis. One potential direction is to make the identification or analysis problems and knowledge resources more automatic. It would be extremely helpful to develop methods that can provide suggestions on potential knowledge resources that will be helpful to the given social content analysis problem, as well as methods that can automatically identify social analysis problems as use cases for certain knowledge resources. This principle can also be potentially extended to the instance level, where, given a certain analysis case, we ask the model to identify whether it requires external knowledge as additional input and where to find external knowledge. It would make the whole analysis pipeline more automatic and intelligent.

Another future research direction is related to the faithfulness and reliability of the model analysis. Even though we may identify useful external resources for model analysis, and in many cases, and witness empirical improvements with the external resources, we still do not have the empirical or theoretical guarantee that the model will faithfully follow the input from external resources to make the analysis. Therefore, it would be essential to further investigate the faithfulness of the model analysis with given external resources. It can be performed with different perspectives. We may investigate methods to enhance the faithfulness. We may also propose methods to identify scenarios in which the models may not follow the external resources provided. All efforts in this direction can potentially make the analysis more transparent, reliable, and human-interpretable.

From the modality perspective, we cover mostly text and images in this thesis. It is also important to explore the possibilities of other modalities, such as audio or video. For example, gestures in videos or tone of voice may also reflect some opinions. We may also use methods discussed in this thesis to analyze the narrative in audio or video and use them as supervision for video-based analysis. In addition, we have pretty rich resources for textual knowledge

bases. But for multi-modal purposes, we mostly focus on Wikipedia images and text. However, in real-world scenarios, we still need other multi-modal knowledge resources that can help us identify fine-grained objects or events and link them to relevant backgrounds or resolutions. For example, instead of an overall object, we may need additional background information regarding the texture of the object and some potential social norms regarding the use of the texture. Therefore, it would still be quite important to further collect or identify multimodal knowledge resources that can provide more diverse and detailed information regarding multimodal input at different levels of granularity.

An important factor that is not covered in this thesis, although they are important in content analysis, is time. For example, a person's attributes may change over time, such as occupation and work location. In current methods, we only consider using the most recent context to reflect the most up-to-date information. But when considering context from a longer period, we need to consider the time-dependent nature of those attributes and instead use methods to update attributes with new information. In addition, stances and opinions may also change over time and, in many cases, can be reflected implicitly with a change of state. It would be essential for generative modeling-based analysis to capture those change points in the long run to provide more accurate analysis results.

Ethics Considerations

The goal of this thesis is to present methodologies and principles to combine knowledge with generative modeling for social content analysis problems. It is also essential to acknowledge and discuss the potential risks associated with this research. We also believe that the NLP community needs to produce detailed information on the potential, pitfalls, and basic limitations of these methodologies so that we can establish standards to facilitate the proper use of these technologies and be vigilant and effective in combating nefarious applications.

Data and model biases. To mitigate potential distributional biases, in Chapter 1, we exhaustively collect WikiData entities without selecting certain groups of users. However, we acknowledge that collective information may still contain unintentional social biases. As an example, one of the potential issues is that people who have WikiData profiles are public figures, which may not reflect the actual distribution over general populations (*e.g.*, occupation). WikiData is constantly edited by a large number of WikiData contributors and maintainers. Although we try to make our study as representative as possible, it is possible that a statement from WikiData may not reflect the preception from certain groups or individual [167]. Similarly, the transitive rules used in Chapter 2 may be over-simplification in some scenarios and, therefore, may introduce polarized bias. In addition, as in Abid et al. [3], the large language models themselves may contain biases. Those biases can be potentially inherited in the synthetic data in Chapter 3, or the generation-based models as we discussed in all chapters. The biases from knowledge retrieval components such as Chapter 3 and Chapter 4 may also be inherited in the social content analysis application with retrieval-augmented generation paradigm in Chapter 5 when using those components to provide the relevant knowledge candidates.

We would like stakeholders to be aware of these issues and we urge stakeholders to first investigate the effect of potential issues before drawing any conclusions for any individual or social group using this work.

Proper use v.s. improper use. The major difference between proper use and improper use is whether the use case follows necessary legal and ethical regulations or framework. For example, Williams et al. [204] proposes an ethical framework based on users' consent to conduct Twitter social research. If the information is not publicly available, one must obtain consent. Opt-out consent can be used when the information is not sensitive, otherwise, opt-in consent is required. This principle should be adopted to apply the methods discussed in this thesis to users. It is a best practice to obtain proper consent first before analyzing a person's background, or conducting extraction or inference on the person's opinions.

Sensitivity of personal information. In Chapter 1, we follow the Twitter Developer Agreement and Policy and remove sensitive personal information. However it is still possible to infer sensitive information indirectly. For example, "candidacy in election" may be possibly used to infer political affiliation although the affiliations are generally public for those people. Similarly, personal pronouns, widely present in tweets, may also be used to infer gender. Furthermore, combinations of various sources might allow personal identification [175, 176]. Even though we do not use private information in our work, based on our results, we speculate that there are unobserved risks of privacy loss for using Twitter. Therefore, We ask that future work should fully comply with regulations, and any non-public or private results should be properly protected [81].

We have set up the following protocol to ensure the proper use and to prevent adverse impact for research in Chapter 1:

- We believe that increasing the transparency of the pipeline can help prevent potential social harm. We plan to release all necessary resources for research reproduction purposes so that others can audit and verify it and prevent overestimation of the model. We also provide a complete list of attributes in Table 1.7 to increase the transparency. We are open to all further explorations that can prevent unintended impacts.
- Our constructed dataset for profile inference research is drawn solely from publicly available WikiData and Twitter, where the ethical consideration should be similar to other work using encyclopedia resources such as [173]. Furthermore, according to WikiData: Oversight, non-public personal information are monitored and removed by Wikidata. According to WikiData Term of Use, we can freely reuse and build upon on WikiData. According to the Twitter Developer Agreement and Policy, we will only release IDs instead of actual content for non-commercial research purposes from academic institutions.

- To ensure the proper use of this work, we will not release the data via a publicly available access point. Instead, we will release the data based on individual requests, and we will ask for consent that 1) requesters are from research institutions, 2) they will follow all the regulations when using our work 3) they will not use the model to infer non-public users unless obtained proper consent from those users.

Although social content analysis tasks discussed in Chapter 2 and Chapter 5 does not directly involve personal information, it is still possible that the analyzed opinions implicitly contain private information, for example, political affiliation or beliefs. Therefore, future work should also consider the direct and indirect impact of the analyzed topic or targets to the speaker before using those analysis tools.

Bibliography

[1] Samir Abdaljalil and Hamdy Mubarak. Wikidata as a source of demographic information. In Nizar Habash, Houda Bouamor, Ramy Eskander, Nadi Tomeh, Ibrahim Abu Farha, Ahmed Abdelali, Samia Touileb, Injy Hamed, Yaser Onaizan, Bashar Alhafni, Wissam Antoun, Salam Khalifa, Hatem Haddad, Imed Zitouni, Badr AlKhamissi, Rawan Almatham, and Khalil Mrini, editors, *Proceedings of The Second Arabic Natural Language Processing Conference*, pages 1–10, Bangkok, Thailand, August 2024. Association for Computational Linguistics. 3

[2] Fabian Abel, Qi Gao, Geert-Jan Houben, and Ke Tao. Twitter-based user modeling for news recommendations. In Francesca Rossi, editor, *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, pages 2962–2966. IJCAI/AAAI, 2013. 27

[3] Abubakar Abid, Maheen Farooqi, and James Zou. Large language models associate muslims with violence. *Nature Machine Intelligence*, 3(6):461–463, 2021. 115

[4] Faiyaz Al Zamal, Wendy Liu, and Derek Ruths. Homophily and latent attribute inference: Inferring latent attributes of twitter users from neighbors. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 6, pages 387–390, 2012. 12, 27

[5] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022. 88, 89, 90

[6] Abeer Aldayel and Walid Magdy. Your stance is exposed! analysing possible factors for

stance detection on social media. *Proc. ACM Hum.-Comput. Interact.*, 3(CSCW), nov 2019. 51

[7] Emily Allaway and Kathleen McKeown. Zero-Shot Stance Detection: A Dataset and Model using Generalized Topic Representations. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8913–8931, Online, November 2020. Association for Computational Linguistics. 51, 96, 98, 103, 107

[8] Emily Allaway and Kathleen McKeown. A unified feature representation for lexical connotations. In Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty, editors, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2145–2163, Online, April 2021. Association for Computational Linguistics. 52

[9] Emily Allaway, Malavika Srikanth, and Kathleen McKeown. Adversarial learning for zero-shot stance detection on social media. 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 4756–4767, Online, June 2021. Association for Computational Linguistics. 51, 96

[10] Milad Alshomary, Shahbaz Syed, Martin Potthast, and Henning Wachsmuth. Target inference in argument conclusion generation. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4334–4345, Online, July 2020. Association for Computational Linguistics. 19

[11] Reinald Kim Amplayo. Rethinking attribute representation and injection for sentiment classification. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5602–5613, Hong Kong, China, November 2019. Association for Computational Linguistics. 11

[12] AI Anthropic. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 1, 2024. 64

[13] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: visual question answering. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 2425–2433. IEEE Computer Society, 2015. 84

[14] Ravi Arunachalam and Sandipan Sarkar. The new eye of government: Citizen sentiment analysis in social media. In Shou-de Lin, Lun-Wei Ku, and Tsung-Ting Kuo, editors, *Proceedings of the IJCNLP 2013 Workshop on Natural Language Processing for Social Media (SocialNLP)*, pages 23–28, Nagoya, Japan, October 2013. Asian Federation of Natural Language Processing. 11

[15] Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. Stance detection with bidirectional conditional encoding. In Jian Su, Kevin Duh, and Xavier Carreras, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 876–885, Austin, Texas, November 2016. Association for Computational Linguistics. 51, 96

[16] Krisztian Balog, Filip Radlinski, and Shushan Arakelyan. Transparent, scrutable and explainable user models for personalized recommendation. In Benjamin Piwowarski, Max Chevalier, Éric Gaussier, Yoelle Maarek, Jian-Yun Nie, and Falk Scholer, editors, *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, pages 265–274. ACM, 2019. 11

[17] Ramy Baly, Giovanni Da San Martino, James Glass, and Preslav Nakov. We can detect your bias: Predicting the political ideology of news articles. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4982–4991, Online, November 2020. Association for Computational Linguistics. 57, 63

[18] David Bamman, Jacob Eisenstein, and Tyler Schnoebelen. Gender identity and lexical variation in social media. *Journal of Sociolinguistics*, 18(2):135–160, 2014. 11

[19] Jeremy Barnes, Laura Oberlaender, Enrica Troiano, Andrey Kutuzov, Jan Buchmann, Rodrigo Agerri, Lilja Øvrelid, and Erik Velldal. SemEval 2022 task 10: Structured sentiment analysis. In Guy Emerson, Natalie Schluter, Gabriel Stanovsky, Ritesh Kumar, Alexis Palmer, Nathan Schneider, Siddharth Singh, and Shyam Ratan, editors, *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1280–1295,

Seattle, United States, July 2022. Association for Computational Linguistics. 33, 51

[20] Angelo Basile, Albert Gatt, and Malvina Nissim. You write like you eat: Stylistic variation as a predictor of social stratification. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2583–2593, Florence, Italy, July 2019. Association for Computational Linguistics. 28

[21] Michele Bevilacqua, Giuseppe Ottaviano, Patrick S. H. Lewis, Scott Yih, Sebastian Riedel, and Fabio Petroni. Autoregressive search engines: Generating substrings as document identifiers. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. 55, 73, 74

[22] Su Lin Blodgett, Lisa Green, and Brendan O'Connor. Demographic dialectal variation in social media: A case study of African-American English. In Jian Su, Kevin Duh, and Xavier Carreras, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1119–1130, Austin, Texas, November 2016. Association for Computational Linguistics. 28

[23] Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, and Rodrigo Nogueira. Inpars: Data augmentation for information retrieval using large language models. *arXiv preprint arXiv:2202.05144*, 2022. 74

[24] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 3, 86

[25] Christopher J. C. Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Gregory N. Hullender. Learning to rank using gradient descent. In Luc De Raedt

and Stefan Wrobel, editors, *Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005), Bonn, Germany, August 7-11, 2005*, volume 119 of *ACM International Conference Proceeding Series*, pages 89–96. ACM, 2005. 84

[26] Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. Autoregressive entity retrieval. *CoRR*, abs/2010.00904, 2020. 55, 58, 74

[27] Jiangui Chen, Ruqing Zhang, Jiafeng Guo, Yiqun Liu, Yixing Fan, and Xueqi Cheng. Corpusbrain: Pre-train a generative retrieval model for knowledge-intensive language tasks. In Mohammad Al Hasan and Li Xiong, editors, *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, pages 191–200. ACM, 2022. 73

[28] Jiangui Chen, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yiqun Liu, Yixing Fan, and Xueqi Cheng. A unified generative retriever for knowledge-intensive language tasks via prompt learning. In Hsin-Hsi Chen, Wei-Jou (Edward) Duh, Hen-Hsen Huang, Makoto P. Kato, Josiane Mothe, and Barbara Poblete, editors, *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 1448–1457. ACM, 2023. 73

[29] Jun Chen, Deyao Zhu, Xiaoqian Shen, Xiang Li, Zechun Liu, Pengchuan Zhang, Raghuraman Krishnamoorthi, Vikas Chandra, Yunyang Xiong, and Mohamed Elhoseiny. Minigpt-v2: large language model as a unified interface for vision-language multi-task learning. *CoRR*, abs/2310.09478, 2023. 88, 90

[30] Xi Chen, Xiao Wang, Soravit Changpinyo, A. J. Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, Alexander Kolesnikov, Joan Puigcerver, Nan Ding, Keran Rong, Hassan Akbari, Gaurav Mishra, Linting Xue, Ashish V. Thapliyal, James Bradbury, and Weicheng Kuo. PaLI: A jointly-scaled multilingual language-image model. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. 80, 82, 85

[31] Xiaoyang Chen, Yanjiang Liu, Ben He, Le Sun, and Yingfei Sun. Understanding differential search index for text retrieval. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10701–10717, Toronto, Canada, July 2023. Association for Computational Linguistics. 55, 74

[32] Xin Chen, Yu Wang, Eugene Agichtein, and Fusheng Wang. A comparative study of demographic attribute inference in twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 9, pages 590–593, 2015. 12, 27

[33] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416, 2022. 35

[34] Costanza Conforti, Jakob Berndt, Mohammad Taher Pilehvar, Chryssi Giannitsarou, Flavio Toxvaerd, and Nigel Collier. Will-they-won't-they: A very large dataset for stance detection on Twitter. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1724, Online, July 2020. Association for Computational Linguistics. 107

[35] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. 64

[36] Dorottya Demszky, Nikhil Garg, Rob Voigt, James Zou, Jesse Shapiro, Matthew Gentzkow, and Dan Jurafsky. Analyzing polarization in social media: Method and application to tweets on 21 mass shootings. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2970–3005, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. 12, 27

[37] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *CoRR*, abs/2305.14314, 2023. 42

[38] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technolo-*

gies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. 35, 100

[39] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 82

[40] Markus Eberts and Adrian Ulges. Span-based joint entity and relation extraction with transformer pre-training. In Giuseppe De Giacomo, Alejandro Catalá, Bistra Dilkina, Michela Milano, Senén Barro, Alberto Bugarín, and Jérôme Lang, editors, *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 2006–2013. IOS Press, 2020. 35

[41] Jacob Eisenstein, Brendan O'Connor, Noah A. Smith, and Eric P. Xing. A latent variable model for geographic lexical variation. In Hang Li and Lluís Màrquez, editors, *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1277–1287, Cambridge, MA, October 2010. Association for Computational Linguistics. 28

[42] Quan Fang, Jitao Sang, Changsheng Xu, and M. Shamim Hossain. Relational user attribute inference in social media. *IEEE Trans. Multim.*, 17(7):1031–1044, 2015. 12, 16, 27

[43] Katja Filippova. Controlled hallucinations: Learning to generate faithfully from noisy data. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 864–870, Online, November 2020. Association for Computational Linguistics. 3

[44] Lucie Flekova, Daniel Preoţiuc-Pietro, and Lyle Ungar. Exploring stylistic variation with age and income on Twitter. In Katrin Erk and Noah A. Smith, editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 313–319, Berlin, Germany, August 2016. Association for Computational Linguistics. 28

[45] Feng Gao, Qing Ping, Govind Thattai, Aishwarya N. Reganti, Ying Nian Wu, and Prem Natarajan. Transform-retrieve-generate: Natural language-centric outside-knowledge

visual question answering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 5057–5067. IEEE, 2022. 86

[46] François Gardères, Maryam Ziaeefard, Baptiste Abeloos, and Freddy Lecue. ConceptBert: Concept-aware representation for visual question answering. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 489–498, Online, November 2020. Association for Computational Linguistics. 80, 86, 90, 91

[47] Michael Glass, Gaetano Rossiello, Md Faisal Mahbub Chowdhury, Ankita Naik, Pengshan Cai, and Alfio Gliozzo. Re2G: Retrieve, rerank, generate. 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 2701–2715, Seattle, United States, July 2022. Association for Computational Linguistics. 80

[48] Eduardo Graells-Garrido, Ricardo Baeza-Yates, and Mounia Lalmas. Representativeness of abortion legislation debate on twitter: A case study in argentina and chile. In *Companion Proceedings of the Web Conference 2020*, WWW '20, page 765–774, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450370240. 51, 96

[49] Sylvain Gugger, Lysandre Debut, Thomas Wolf, Philipp Schmid, Zachary Mueller, Sourab Mangrulkar, Marc Sun, and Benjamin Bossan. Accelerate: Training and inference at scale made simple, efficient and adaptable. https://github.com/huggingface/accelerate, 2022. 43, 64

[50] Liangke Gui, Borui Wang, Qiuyuan Huang, Alexander Hauptmann, Yonatan Bisk, and Jianfeng Gao. KAT: A knowledge augmented transformer for vision-and-language. 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 956–968, Seattle, United States, July 2022. Association for Computational Linguistics. 80, 81, 82, 86, 90, 91

[51] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Realm: retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020. 3

[52] Ido Guy. The role of user location in personalized search and recommendation. In Hannes

Werthner, Markus Zanker, Jennifer Golbeck, and Giovanni Semeraro, editors, *Proceedings of the 9th ACM Conference on Recommender Systems, RecSys 2015, Vienna, Austria, September 16-20, 2015*, page 236. ACM, 2015. 11

[53] Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. The argument reasoning comprehension task: Identification and reconstruction of implicit warrants. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1930–1940, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. 103

[54] Andreas Hanselowski, Avinesh PVS, Benjamin Schiller, Felix Caspelherr, Debanjan Chaudhuri, Christian M. Meyer, and Iryna Gurevych. A retrospective analysis of the fake news challenge stance-detection task. In Emily M. Bender, Leon Derczynski, and Pierre Isabelle, editors, *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1859–1874, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics. 96

[55] Helia Hashemi, Yong Zhuang, Sachith Sri Ram Kothur, Srivas Prasad, Edgar Meij, and W. Bruce Croft. Dense retrieval adaptation using target domain description. In Masaharu Yoshioka, Julia Kiseleva, and Mohammad Aliannejadi, editors, *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR 2023, Taipei, Taiwan, 23 July 2023*, pages 95–104. ACM, 2023. 58

[56] Zihao He, Negar Mokhberian, and Kristina Lerman. Infusing knowledge from Wikipedia to enhance stance detection. In Jeremy Barnes, Orphée De Clercq, Valentin Barriere, Shabnam Tafreshi, Sawsan Alqahtani, João Sedoc, Roman Klinger, and Alexandra Balahur, editors, *Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis*, pages 71–77, Dublin, Ireland, May 2022. Association for Computational Linguistics. 3, 51, 96, 97, 100, 103, 107, 108

[57] Sebastian Hofstätter, Jiecao Chen, Karthik Raman, and Hamed Zamani. Fid-light: Efficient and effective retrieval-augmented text generation. In Hsin-Hsi Chen, Wei-Jou (Edward) Duh, Hen-Hsen Huang, Makoto P. Kato, Josiane Mothe, and Barbara Poblete, editors, *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 1437–1447. ACM, 2023. 80

[58] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. New York, NY, USA, 2004. Association for Computing Machinery. ISBN 1581138881. 33

[59] Chao Huang, Dong Wang, Shenglong Zhu, and Daniel Yue Zhang. Towards unsupervised home location inference from online social media. In James Joshi, George Karypis, Ling Liu, Xiaohua Hu, Ronay Ak, Yinglong Xia, Weijia Xu, Aki-Hiro Sato, Sudarsan Rachuri, Lyle H. Ungar, Philip S. Yu, Rama Govindaraju, and Toyotaro Suzumura, editors, *2016 IEEE International Conference on Big Data (IEEE BigData 2016), Washington DC, USA, December 5-8, 2016*, pages 676–685. IEEE Computer Society, 2016. 12, 27

[60] Pere-Lluís Huguet Cabot and Roberto Navigli. REBEL: Relation extraction by end-to-end language generation. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2370–2381, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. 36

[61] Tunazzina Islam and Dan Goldwasser. Analysis of twitter users' lifestyle choices using joint embedding model. In Ceren Budak, Meeyoung Cha, Daniele Quercia, and Lexing Xie, editors, *Proceedings of the Fifteenth International AAAI Conference on Web and Social Media, ICWSM 2021, held virtually, June 7-10, 2021*, pages 242–253. AAAI Press, 2021. 27

[62] Tunazzina Islam and Dan Goldwasser. Twitter user representation using weakly supervised graph embedding. In Ceren Budak, Meeyoung Cha, and Daniele Quercia, editors, *Proceedings of the Sixteenth International AAAI Conference on Web and Social Media, ICWSM 2022, Atlanta, Georgia, USA, June 6-9, 2022*, pages 358–369. AAAI Press, 2022. 27

[63] Gautier Izacard and Edouard Grave. Distilling knowledge from reader to retriever for question answering. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 88

[64] Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. In Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty, editors, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880, Online, April 2021. Association for Computational Linguistics. 83

[65] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning, 2021. 56, 71

[66] Myungha Jang and James Allan. Explaining controversy on social media via stance summarization. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 1221–1224, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356572. 96

[67] Vitor Jeronymo, Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, Roberto Lotufo, Jakub Zavrel, and Rodrigo Nogueira. Inpars-v2: Large language models as efficient dataset generators for information retrieval. *arXiv preprint arXiv:2301.01820*, 2023. 74

[68] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 4904–4916. PMLR, 2021. 81, 82

[69] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b. *CoRR*, abs/2310.06825, 2023. 64

[70] Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts. *CoRR*, abs/2401.04088, 2024. 64

[71] Yan Jiang, Jinhua Gao, Huawei Shen, and Xueqi Cheng. Few-shot stance detection via target-aware prompt distillation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 837–847, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450387323. 51, 96

[72] Anders Johannsen, Dirk Hovy, and Anders Søgaard. Cross-lingual syntactic variation over age and gender. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 103–112, Beijing, China, July 2015. Association for Com-

putational Linguistics. 28

[73] Amita Kamath, Christopher Clark, Tanmay Gupta, Eric Kolve, Derek Hoiem, and Aniruddha Kembhavi. Webly supervised concept expansion for general purpose vision models. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXXVI*, volume 13696 of *Lecture Notes in Computer Science*, pages 662–681. Springer, 2022. 87

[74] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *CoRR*, abs/2001.08361, 2020. 3

[75] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen Tau Yih. Dense passage retrieval for open-domain question answering. In *2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 6769–6781. Association for Computational Linguistics (ACL), 2020. 56, 58

[76] Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. The hateful memes challenge: Detecting hate speech in multimodal memes. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 103

[77] Soo-Min Kim and Eduard Hovy. Determining the sentiment of opinions. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 1367–1373, Geneva, Switzerland, aug 23–aug 27 2004. COLING. 33

[78] Sunghwan Mac Kim, Qiongkai Xu, Lizhen Qu, Stephen Wan, and Cécile Paris. Demographic inference on Twitter using recursive neural networks. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 471–477, Vancouver, Canada, July 2017. Association for Computational Linguistics. 12, 27

[79] Varsha Kishore, Chao Wan, Justin Lovelace, Yoav Artzi, and Kilian Q. Weinberger. Incdsi: Incrementally updatable document retrieval. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii,*

USA, volume 202 of *Proceedings of Machine Learning Research*, pages 17122–17134. PMLR, 2023. 63, 65, 67

[80] Alfred Kobsa. User modeling and user-adapted interaction. In Catherine Plaisant, editor, *Conference on Human Factors in Computing Systems, CHI 1994, Boston, Massachusetts, USA, April 24-28, 1994, Conference Companion*, pages 415–416. ACM, 1994. 27

[81] Anne Kreuter, Kai Sassenberg, and Roman Klinger. Items from psychometric tests as training data for personality profiling models of Twitter users. In Jeremy Barnes, Orphée De Clercq, Valentin Barriere, Shabnam Tafreshi, Sawsan Alqahtani, João Sedoc, Roman Klinger, and Alexandra Balahur, editors, *Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis*, pages 315–323, Dublin, Ireland, May 2022. Association for Computational Linguistics. 116

[82] Sachin Kumar, Chan Young Park, and Yulia Tsvetkov. Gen-z: Generative zero-shot text classification with contextualized label descriptions. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. 108

[83] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. 57, 63

[84] William Labov. The social motivation of a sound change. *Word*, 19(3):273–309, 1963. 28

[85] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. In Kevin Knight, Ani Nenkova, and Owen Rambow, editors, *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California, June 2016. Association for Computational Linguistics. 18

[86] Brian Larson. Gender as a variable in natural-language processing: Ethical considerations. In Dirk Hovy, Shannon Spruit, Margaret Mitchell, Emily M. Bender, Michael Strube, and Hanna Wallach, editors, *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 1–11, Valencia, Spain, April 2017. Association for

Computational Linguistics. 12

[87] Hyunji Lee, Sohee Yang, Hanseok Oh, and Minjoon Seo. Generative multi-hop retrieval. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1417–1436, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. 73

[88] Hyunji Lee, JaeYoung Kim, Hoyeon Chang, Hanseok Oh, Sohee Yang, Vladimir Karpukhin, Yi Lu, and Minjoon Seo. Nonparametric decoding for generative retrieval. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12642–12661, Toronto, Canada, July 2023. Association for Computational Linguistics. 55

[89] Jinhyuk Lee, Seongjun Yun, Hyunjae Kim, Miyoung Ko, and Jaewoo Kang. Ranking paragraphs for improving answer recall in open-domain question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 565–569, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. 80

[90] Sunkyung Lee, Minjin Choi, and Jongwuk Lee. GLEN: Generative retrieval via lexical index learning. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7693–7704, Singapore, December 2023. Association for Computational Linguistics. 73

[91] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. Predicting positive and negative links in online social networks. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, page 641–650, New York, NY, USA, 2010. Association for Computing Machinery. ISBN 9781605587998. 51

[92] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. 12

[93] Mike Lewis and Angela Fan. Generative question answering: Learning to answer the whole question. In *7th International Conference on Learning Representations, ICLR 2019,*

New Orleans, LA, USA, May 6-9, 2019. OpenReview.net, 2019. 36, 98, 108

[94] 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 Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics. 98, 108

[95] Guohao Li, Xin Wang, and Wenwu Zhu. Boosting visual question answering with context-aware knowledge aggregation. In Chang Wen Chen, Rita Cucchiara, Xian-Sheng Hua, Guo-Jun Qi, Elisa Ricci, Zhengyou Zhang, and Roger Zimmermann, editors, *MM '20: The 28th ACM International Conference on Multimedia, Virtual Event / Seattle, WA, USA, October 12-16, 2020*, pages 1227–1235. ACM, 2020. 80, 86, 90, 91

[96] Jiwei Li, Alan Ritter, and Eduard Hovy. Weakly supervised user profile extraction from Twitter. In Kristina Toutanova and Hua Wu, editors, *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 165–174, Baltimore, Maryland, June 2014. Association for Computational Linguistics. xiii, 12, 16, 17, 21, 23, 24

[97] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 19730–19742. PMLR, 2023. 82, 88, 89, 90

[98] Sha Li, Heng Ji, and Jiawei Han. Document-level event argument extraction by conditional generation. 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 894–908, Online, June 2021. Association for Computational Linguistics. 19, 36, 98, 108

[99] Xiaoxi Li, Zhicheng Dou, Yujia Zhou, and Fangchao Liu. Corpuslm: Towards a unified language model on corpus for knowledge-intensive tasks. In Grace Hui Yang, Hongning Wang, Sam Han, Claudia Hauff, Guido Zuccon, and Yi Zhang, editors, *Proceedings of*

the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024, pages 26–37. ACM, 2024. 74

[100] Yongqi Li, Nan Yang, Liang Wang, Furu Wei, and Wenjie Li. Generative retrieval for conversational question answering. *Inf. Process. Manag.*, 60(5):103475, 2023. 73

[101] Yongqi Li, Nan Yang, Liang Wang, Furu Wei, and Wenjie Li. Multiview identifiers enhanced generative retrieval. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6636–6648, Toronto, Canada, July 2023. Association for Computational Linguistics. 73

[102] Yongqi Li, Nan Yang, Liang Wang, Furu Wei, and Wenjie Li. Learning to rank in generative retrieval. In Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan, editors, *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 8716–8723. AAAI Press, 2024. 55, 56, 61, 73

[103] Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*, 2023. 71

[104] Bin Liang, Yonghao Fu, Lin Gui, Min Yang, Jiachen Du, Yulan He, and Ruifeng Xu. Target-adaptive graph for cross-target stance detection. In *Proceedings of the Web Conference 2021*, WWW '21, page 3453–3464, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450383127. 51, 96

[105] Bin Liang, Zixiao Chen, Lin Gui, Yulan He, Min Yang, and Ruifeng Xu. Zero-shot stance detection via contrastive learning. In *Proceedings of the ACM Web Conference 2022*, WWW '22, page 2738–2747, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450390965. 51, 96, 107

[106] Bin Liang, Qinglin Zhu, Xiang Li, Min Yang, Lin Gui, Yulan He, and Ruifeng Xu. JointCL: A joint contrastive learning framework for zero-shot stance detection. 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 81–91, Dublin, Ireland, May 2022. Association for Computational Linguistics. 51, 96, 107

[107] Shuailong Liang, Olivia Nicol, and Yue Zhang. Who blames whom in a crisis? detecting

blame ties from news articles using neural networks. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 655–662. AAAI Press, 2019. 43

[108] Sheng-Chieh Lin, Chankyu Lee, Mohammad Shoeybi, Jimmy Lin, Bryan Catanzaro, and Wei Ping. Mm-embed: Universal multimodal retrieval with multimodal llms. *CoRR*, abs/2411.02571, 2024. 97, 102

[109] Yuanze Lin, Yujia Xie, Dongdong Chen, Yichong Xu, Chenguang Zhu, and Lu Yuan. REVIVE: regional visual representation matters in knowledge-based visual question answering. In *NeurIPS*, 2022. 80, 81, 82, 85, 86, 90

[110] Yuxiao Lin, Yuxian Meng, Xiaofei Sun, Qinghong Han, Kun Kuang, Jiwei Li, and Fei Wu. BertGCN: Transductive text classification by combining GNN and BERT. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1456–1462, Online, August 2021. Association for Computational Linguistics. 51, 96, 103

[111] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *CoRR*, abs/2310.03744, 2023. 88, 90

[112] Rui Liu, Zheng Lin, Yutong Tan, and Weiping Wang. Enhancing zero-shot and few-shot stance detection with commonsense knowledge graph. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3152–3157, Online, August 2021. Association for Computational Linguistics. 51, 96, 103, 107

[113] Tie-Yan Liu. Learning to rank for information retrieval. *Found. Trends Inf. Retr.*, 3(3): 225–331, 2009. 80

[114] Wendy Liu and Derek Ruths. What's in a name? using first names as features for gender inference in twitter. In *2013 AAAI Spring Symposium Series*, 2013. 12, 27

[115] Wendy Liu, Faiyaz Zamal, and Derek Ruths. Using social media to infer gender composition of commuter populations. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 6, pages 26–29, 2012. 12, 27

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

BERT pretraining approach. *CoRR*, abs/1907.11692, 2019. 18, 35, 42

[117] Yujian Liu, Xinliang Frederick Zhang, David Wegsman, Nicholas Beauchamp, and Lu Wang. POLITICS: Pretraining with same-story article comparison for ideology prediction and stance detection. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, editors, *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1354–1374, Seattle, United States, July 2022. Association for Computational Linguistics. 43

[118] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 103

[119] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 13–23, 2019. 87

[120] Kai Lu, Yi Zhang, Lanbo Zhang, and Shuxin Wang. Exploiting user and business attributes for personalized business recommendation. In Ricardo Baeza-Yates, Mounia Lalmas, Alistair Moffat, and Berthier A. Ribeiro-Neto, editors, *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015*, pages 891–894. ACM, 2015. 11

[121] Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. Unified structure generation for universal information extraction. 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 5755–5772, Dublin, Ireland, May 2022. Association for Computational Linguistics. 36

[122] Man Luo, Yankai Zeng, Pratyay Banerjee, and Chitta Baral. Weakly-supervised visual-retriever-reader for knowledge-based question answering. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6417–6431, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. 80, 81, 82, 86, 91

[123] Ji Ma, Ivan Korotkov, Yinfei Yang, Keith Hall, and Ryan McDonald. Zero-shot neural

passage retrieval via domain-targeted synthetic question generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1075–1088, 2021. 74

[124] Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. Reader-guided passage reranking for open-domain question answering. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 344–350, Online, August 2021. Association for Computational Linguistics. 80

[125] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. OK-VQA: A visual question answering benchmark requiring external knowledge. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 3195–3204. Computer Vision Foundation / IEEE, 2019. 81, 85

[126] Kenneth Marino, Xinlei Chen, Devi Parikh, Abhinav Gupta, and Marcus Rohrbach. KRISP: integrating implicit and symbolic knowledge for open-domain knowledge-based VQA. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 14111–14121. Computer Vision Foundation / IEEE, 2021. 86, 87, 91

[127] Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online, July 2020. Association for Computational Linguistics. 3

[128] Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. Cross-task generalization via natural language crowdsourcing instructions. 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 3470–3487, Dublin, Ireland, May 2022. Association for Computational Linguistics. 3

[129] Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D. Manning, and Chelsea Finn. Memory-based model editing at scale. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 15817–15831. PMLR, 2022. 3

[130] Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. SemEval-2016 task 6: Detecting stance in tweets. In Steven Bethard, Marine Carpuat, Daniel Cer, David Jurgens, Preslav Nakov, and Torsten Zesch, editors, *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41, San Diego, California, June 2016. Association for Computational Linguistics. 33, 51, 96, 107

[131] Mitra Mohtarami, Ramy Baly, James Glass, Preslav Nakov, Lluís Màrquez, and Alessandro Moschitti. Automatic stance detection using end-to-end memory networks. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 767–776, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. 51, 96

[132] Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. Crosslingual generalization through multitask finetuning. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, Toronto, Canada, July 2023. Association for Computational Linguistics. 3, 35, 37, 42, 43

[133] Dong Nguyen and Jacob Eisenstein. A kernel independence test for geographical language variation. *Computational Linguistics*, 43(3):567–592, September 2017. 28

[134] Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. From doc2query to doctttttquery. 2019. 56, 60, 74

[135] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. 3, 56

[136] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classifica-

tion using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 79–86. Association for Computational Linguistics, July 2002. 33

[137] Richard Yuanzhe Pang, Weizhe Yuan, Kyunghyun Cho, He He, Sainbayar Sukhbaatar, and Jason Weston. Iterative reasoning preference optimization. *CoRR*, abs/2404.19733, 2024. 56, 61

[138] Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cícero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. Structured prediction as translation between augmented natural languages. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 36

[139] Kunwoo Park, Zhufeng Pan, and Jungseock Joo. Who blames or endorses whom? entity-to-entity directed sentiment extraction in news text. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4091–4102, Online, August 2021. Association for Computational Linguistics. 33, 34, 35, 36, 39, 41, 43, 51

[140] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035, 2019. 43

[141] Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. SemEval-2014 task 4: Aspect based sentiment analysis. In Preslav Nakov and Torsten Zesch, editors, *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland, August 2014. Association for Computational Linguistics. 33

[142] Ronak Pradeep, Kai Hui, Jai Gupta, Adam Lelkes, Honglei Zhuang, Jimmy Lin, Donald Metzler, and Vinh Tran. How does generative retrieval scale to millions of passages?

In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1305–1321, Singapore, December 2023. Association for Computational Linguistics. 68

[143] Yujie Qian, Jie Tang, Zhilin Yang, Binxuan Huang, Wei Wei, and Kathleen M Carley. A probabilistic framework for location inference from social media. *arXiv preprint arXiv:1702.07281*, 2017. 12, 27

[144] Yujie Qian, Enrico Santus, Zhijing Jin, Jiang Guo, and Regina Barzilay. GraphIE: A graph-based framework for information extraction. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 751–761, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. xiii, 12, 17, 21, 22, 23, 24

[145] 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. 37

[146] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021. 82, 85

[147] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 61

[148] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2020. 12, 19, 20, 36, 64

[149] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learn-

ing with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1), jun 2022. ISSN 1532-4435. 98, 108

[150] Delip Rao, David Yarowsky, Abhishek Shreevats, and Manaswi Gupta. Classifying latent user attributes in twitter. In José Carlos Cortizo, Francisco M. Carrero, Iván Cantador, José Antonio Troyano Jiménez, and Paolo Rosso, editors, *Proceedings of the 2nd International Workshop on Search and Mining User-Generated Contents, SMUC@CIKM 2010, Toronto, ON, Canada, October 30, 2010*, pages 37–44. ACM, 2010. 12, 27

[151] Delip Rao, Michael Paul, Clay Fink, David Yarowsky, Timothy Oates, and Glen Coppersmith. Hierarchical bayesian models for latent attribute detection in social media. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 5, pages 598–601, 2011. 12, 27

[152] Hannah Rashkin, Sameer Singh, and Yejin Choi. Connotation frames: A data-driven investigation. In Katrin Erk and Noah A. Smith, editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 311–321, Berlin, Germany, August 2016. Association for Computational Linguistics. 52

[153] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In Rajesh Gupta, Yan Liu, Jiliang Tang, and B. Aditya Prakash, editors, *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 3505–3506. ACM, 2020. 64

[154] Vikas Raunak, Arul Menezes, and Marcin Junczys-Dowmunt. The curious case of hallucinations in neural machine translation. 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 1172–1183, Online, June 2021. Association for Computational Linguistics. 3

[155] Adam Roberts, Hyung Won Chung, Anselm Levskaya, Gaurav Mishra, James Bradbury, Daniel Andor, Sharan Narang, Brian Lester, Colin Gaffney, Afroz Mohiuddin, Curtis Hawthorne, Aitor Lewkowycz, Alex Salcianu, Marc van Zee, Jacob Austin, Sebastian Goodman, Livio Baldini Soares, Haitang Hu, Sasha Tsvyashchenko, Aakanksha Chowdhery, Jasmijn Bastings, Jannis Bulian, Xavier Garcia, Jianmo Ni, Andrew Chen, Kathleen

Kenealy, Jonathan H. Clark, Stephan Lee, Dan Garrette, James Lee-Thorp, Colin Raffel, Noam Shazeer, Marvin Ritter, Maarten Bosma, Alexandre Passos, Jeremy Maitin-Shepard, Noah Fiedel, Mark Omernick, Brennan Saeta, Ryan Sepassi, Alexander Spiridonov, Joshua Newlan, and Andrea Gesmundo. Scaling up models and data with `t5x` and `seqio`. *arXiv preprint arXiv:2203.17189*, 2022. 85

[156] Stephen E. Robertson and Steve Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In W. Bruce Croft and C. J. van Rijsbergen, editors, *Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Dublin, Ireland, 3-6 July 1994 (Special Issue of the SIGIR Forum)*, pages 232–241. ACM/Springer, 1994. 71

[157] Sara Rosenthal and Kathleen McKeown. Age prediction in blogs: A study of style, content, and online behavior in pre- and post-social media generations. In Dekang Lin, Yuji Matsumoto, and Rada Mihalcea, editors, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 763–772, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. 12, 27

[158] Shigeyuki Sakaki, Yasuhide Miura, Xiaojun Ma, Keigo Hattori, and Tomoko Ohkuma. Twitter user gender inference using combined analysis of text and image processing. In Anja Belz, Darren Cosker, Frank Keller, William Smith, Kalina Bontcheva, Sien Moens, and Alan Smeaton, editors, *Proceedings of the Third Workshop on Vision and Language*, pages 54–61, Dublin, Ireland, August 2014. Dublin City University and the Association for Computational Linguistics. 12, 27

[159] Alireza Salemi, Juan Altmayer Pizzorno, and Hamed Zamani. A symmetric dual encoding dense retrieval framework for knowledge-intensive visual question answering. In Hsin-Hsi Chen, Wei-Jou (Edward) Duh, Hen-Hsen Huang, Makoto P. Kato, Josiane Mothe, and Barbara Poblete, editors, *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 110–120. ACM, 2023. 82, 90, 91

[160] Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli,

Thibault Févry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M. Rush. Multitask prompted training enables zero-shot task generalization. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. 35

[161] Maarten Sap, Gregory Park, Johannes Eichstaedt, Margaret Kern, David Stillwell, Michal Kosinski, Lyle Ungar, and Hansen Andrew Schwartz. Developing age and gender predictive lexica over social media. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1146–1151, Doha, Qatar, October 2014. Association for Computational Linguistics. 12, 27

[162] Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilic, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, and et al. BLOOM: A 176b-parameter open-access multilingual language model. *CoRR*, abs/2211.05100, 2022. 3

[163] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-OKVQA: A benchmark for visual question answering using world knowledge. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part VIII*, volume 13668 of *Lecture Notes in Computer Science*, pages 146–162. Springer, 2022. 81, 85

[164] Abigail See, Peter J. Liu, and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada, July 2017. Association for Computational Linguistics. 19

[165] Zhenwei Shao, Zhou Yu, Meng Wang, and Jun Yu. Prompting large language models with

answer heuristics for knowledge-based visual question answering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 14974–14983. IEEE, 2023. 90, 91

[166] Xuehua Shen, Bin Tan, and ChengXiang Zhai. Implicit user modeling for personalized search. In Otthein Herzog, Hans-Jörg Schek, Norbert Fuhr, Abdur Chowdhury, and Wilfried Teiken, editors, *Proceedings of the 2005 ACM CIKM International Conference on Information and Knowledge Management, Bremen, Germany, October 31 - November 5, 2005*, pages 824–831. ACM, 2005. 11

[167] Kartik Shenoy, Filip Ilievski, Daniel Garijo, Daniel Schwabe, and Pedro Szekely. A study of the quality of wikidata. *Journal of Web Semantics*, 72:100679, 2022. 115

[168] Qingyi Si, Zheng Lin, Ming yu Zheng, Peng Fu, and Weiping Wang. Check it again:progressive visual question answering via visual entailment. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4101–4110, Online, August 2021. Association for Computational Linguistics. 91

[169] Umme Aymun Siddiqua, Abu Nowshed Chy, and Masaki Aono. Stance detection on microblog focusing on syntactic tree representation. In Ying Tan, Yuhui Shi, and Qirong Tang, editors, *Data Mining and Big Data*, pages 478–490, Cham, 2018. Springer International Publishing. ISBN 978-3-319-93803-5. 51

[170] Parinaz Sobhani, Diana Inkpen, and Xiaodan Zhu. A dataset for multi-target stance detection. In Mirella Lapata, Phil Blunsom, and Alexander Koller, editors, *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 551–557, Valencia, Spain, April 2017. Association for Computational Linguistics. 34, 51, 96

[171] Swapna Somasundaran and Janyce Wiebe. Recognizing stances in ideological on-line debates. In Diana Inkpen and Carlo Strapparava, editors, *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 116–124, Los Angeles, CA, June 2010. Association for Computational Linguistics. 33, 51, 96

[172] Krishna Srinivasan, Karthik Raman, Jiecao Chen, Michael Bendersky, and Marc Najork. Wit: Wikipedia-based image text dataset for multimodal multilingual machine learning.

In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 2443–2449, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380379. 82, 97, 102

[173] Jiao Sun and Nanyun Peng. Men are elected, women are married: Events gender bias on Wikipedia. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 350–360, Online, August 2021. Association for Computational Linguistics. 116

[174] Weiwei Sun, Lingyong Yan, Zheng Chen, Shuaiqiang Wang, Haichao Zhu, Pengjie Ren, Zhumin Chen, Dawei Yin, Maarten de Rijke, and Zhaochun Ren. Learning to tokenize for generative retrieval. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 55

[175] Latanya Sweeney. Uniqueness of simple demographics in the u.s. population. *LIDAP-WP4, 2000*, 2000. 116

[176] Latanya Sweeney. Simple demographics often identify people uniquely. *LIDAP-WP4, 2000*, 2000. 116

[177] Hao Tan and Mohit Bansal. LXMERT: Learning cross-modality encoder representations from transformers. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111, Hong Kong, China, November 2019. Association for Computational Linguistics. 87

[178] Duyu Tang, Bing Qin, and Ting Liu. Learning semantic representations of users and products for document level sentiment classification. In Chengqing Zong and Michael Strube, editors, *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1014–1023, Beijing, China, July 2015. Association for Computational Linguistics. 11

[179] Qiaoyu Tang, Jiawei Chen, Zhuoqun Li, Bowen Yu, Yaojie Lu, Cheng Fu, Haiyang Yu,

Hongyu Lin, Fei Huang, Ben He, Xianpei Han, Le Sun, and Yongbin Li. Self-retrieval: End-to-end information retrieval with one large language model, 2024. 73

[180] Yixuan Tang and Yi Yang. Multihop-rag: Benchmarking retrieval-augmented generation for multi-hop queries. *CoRR*, abs/2401.15391, 2024. 57, 63

[181] Yubao Tang, Ruqing Zhang, Jiafeng Guo, Jiangui Chen, Zuowei Zhu, Shuaiqiang Wang, Dawei Yin, and Xueqi Cheng. Semantic-enhanced differentiable search index inspired by learning strategies. In Ambuj K. Singh, Yizhou Sun, Leman Akoglu, Dimitrios Gunopulos, Xifeng Yan, Ravi Kumar, Fatma Ozcan, and Jieping Ye, editors, *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023, Long Beach, CA, USA, August 6-10, 2023*, pages 4904–4913. ACM, 2023. 73

[182] Yubao Tang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Wei Chen, and Xueqi Cheng. Listwise generative retrieval models via a sequential learning process. *ACM Trans. Inf. Syst.*, 42(5):133:1–133:31, 2024. 73, 74

[183] Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Prakash Gupta, Tal Schuster, William W. Cohen, and Donald Metzler. Transformer memory as a differentiable search index. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. 55, 57, 73

[184] Jaime Teevan, Meredith Ringel Morris, and Steve Bush. Discovering and using groups to improve personalized search. In Ricardo Baeza-Yates, Paolo Boldi, Berthier A. Ribeiro-Neto, and Berkant Barla Cambazoglu, editors, *Proceedings of the Second International Conference on Web Search and Web Data Mining, WSDM 2009, Barcelona, Spain, February 9-11, 2009*, pages 15–24. ACM, 2009. 11

[185] Hannu Toivonen and Michele Boggia, editors. *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation*, Online, April 2021. Association for Computational Linguistics. 3

[186] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas,

Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288, 2023. 3, 35

[187] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. 104

[188] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017. 37, 55, 83

[189] Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. Trl: Transformer reinforcement learning. `https://github.com/huggingface/trl`, 2020. 64

[190] Denny Vrandecic and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, 2014. 12

[191] Somin Wadhwa, Silvio Amir, and Byron Wallace. Revisiting relation extraction in the era of large language models. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15566–15589, Toronto, Canada, July 2023. Association for Computational Linguistics. 36

[192] Haoyu Wang, Muhao Chen, Hongming Zhang, and Dan Roth. Joint constrained learning for event-event relation extraction. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 696–706, Online, November 2020. Association for Computational Linguistics. 35, 39, 51

[193] Jue Wang and Wei Lu. Two are better than one: Joint entity and relation extraction with table-sequence encoders. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1706–1721, Online, November 2020. Association for Computational Linguistics. 35

[194] Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. Gpl: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2345–2360, 2022. 74

[195] Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Improving text embeddings with large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11897–11916, Bangkok, Thailand, August 2024. Association for Computational Linguistics. 71

[196] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 101

[197] Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerry Tesauro, Bowen Zhou, and Jing Jiang. R^3: Reinforced ranker-reader for open-domain question answering. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5981–5988. AAAI Press, 2018. 80

[198] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. 3, 35

[199] Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. Neural text generation with unlikelihood training. In *8th International Conference*

on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net, 2020. 96, 99

[200] Haoyang Wen and Alexander Hauptmann. Zero-shot and few-shot stance detection on varied topics via conditional generation. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1491–1499, Toronto, Canada, July 2023. Association for Computational Linguistics. 51

[201] Haoyang Wen and Heng Ji. Utilizing relative event time to enhance event-event temporal relation extraction. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10431–10437, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. 35

[202] Haoyang Wen, Zhenxin Xiao, Eduard Hovy, and Alexander Hauptmann. Towards open-domain Twitter user profile inference. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3172–3188, Toronto, Canada, July 2023. Association for Computational Linguistics. 36

[203] Janyce Wiebe, Theresa Wilson, and Claire Cardie. Annotating expressions of opinions and emotions in language. *Lang. Resour. Evaluation*, 39(2-3):165–210, 2005. 33

[204] Matthew L Williams, Pete Burnap, and Luke Sloan. Towards an ethical framework for publishing twitter data in social research: Taking into account users' views, online context and algorithmic estimation. *Sociology*, 51(6):1149–1168, 2017. PMID: 29276313. 116

[205] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771, 2019. 43

[206] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In Qun Liu and David Schlangen, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language*

Processing: System Demonstrations, pages 38–45, Online, October 2020. Association for Computational Linguistics. 64

[207] Jialin Wu, Jiasen Lu, Ashish Sabharwal, and Roozbeh Mottaghi. Multi-modal answer validation for knowledge-based VQA. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 2712–2721. AAAI Press, 2022. 86, 91

[208] Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. C-pack: Packaged resources to advance general chinese embedding, 2023. 71

[209] Chang Xu, Cécile Paris, Surya Nepal, and Ross Sparks. Cross-target stance classification with self-attention networks. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 778–783, Melbourne, Australia, July 2018. Association for Computational Linguistics. 51, 96

[210] Hang Yan, Junqi Dai, Tuo Ji, Xipeng Qiu, and Zheng Zhang. A unified generative framework for aspect-based sentiment analysis. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2416–2429, Online, August 2021. Association for Computational Linguistics. 36, 98, 108

[211] Tianchi Yang, Minghui Song, Zihan Zhang, Haizhen Huang, Weiwei Deng, Feng Sun, and Qi Zhang. Auto search indexer for end-to-end document retrieval. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6955–6970, Singapore, December 2023. Association for Computational Linguistics. 73

[212] Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang. An empirical study of GPT-3 for few-shot knowledge-based VQA. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 3081–3089. AAAI Press, 2022. 86, 90, 91

[213] Jing Yao, Zhicheng Dou, and Ji-Rong Wen. Employing personal word embeddings for personalized search. In Jimmy X. Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu, editors, *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 1359–1368. ACM, 2020. 11

[214] Yue Yu, Yuchen Zhuang, Jieyu Zhang, Yu Meng, Alexander J. Ratner, Ranjay Krishna, Jiaming Shen, and Chao Zhang. Large language model as attributed training data generator: A tale of diversity and bias. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 57, 63

[215] Hansi Zeng, Chen Luo, Bowen Jin, Sheikh Muhammad Sarwar, Tianxin Wei, and Hamed Zamani. Scalable and effective generative information retrieval. In Tat-Seng Chua, Chong-Wah Ngo, Ravi Kumar, Hady W. Lauw, and Roy Ka-Wei Lee, editors, *Proceedings of the ACM on Web Conference 2024, WWW 2024, Singapore, May 13-17, 2024*, pages 1441–1452. ACM, 2024. 73

[216] Hailin Zhang, Yujing Wang, Qi Chen, Ruiheng Chang, Ting Zhang, Ziming Miao, Yingyan Hou, Yang Ding, Xupeng Miao, Haonan Wang, Bochen Pang, Yuefeng Zhan, Hao Sun, Weiwei Deng, Qi Zhang, Fan Yang, Xing Xie, Mao Yang, and Bin Cui. Model-enhanced vector index. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 73

[217] Peitian Zhang, Zheng Liu, Yujia Zhou, Zhicheng Dou, Fangchao Liu, and Zhao Cao. Generative retrieval via term set generation. In Grace Hui Yang, Hongning Wang, Sam Han, Claudia Hauff, Guido Zuccon, and Yi Zhang, editors, *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, pages 458–468. ACM, 2024. 55

[218] Rong Zhang, Qifei Zhou, Bo An, Weiping Li, Tong Mo, and Bo Wu. Enhancing neural models with vulnerability via adversarial attack. In Donia Scott, Nuria Bel, and Chengqing Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1133–1146, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. 96

[219] Shaodian Zhang, Lin Qiu, Frank Chen, Weinan Zhang, Yong Yu, and Noémie Elhadad. We make choices we think are going to save us: Debate and stance identification for online breast cancer cam discussions. In *Proceedings of the 26th International Conference on World Wide Web Companion*, WWW '17 Companion, page 1073–1081, Republic and Canton of Geneva, CHE, 2017. International World Wide Web Conferences Steering Committee. ISBN 9781450349147. 96

[220] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona T. Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. OPT: open pre-trained transformer language models. *CoRR*, abs/2205.01068, 2022. 3

[221] Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Jialin Pan, and Lidong Bing. Sentiment analysis in the era of large language models: A reality check. *CoRR*, abs/2305.15005, 2023. 47

[222] Xinliang Frederick Zhang, Nick Beauchamp, and Lu Wang. Generative entity-to-entity stance detection with knowledge graph augmentation. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9950–9969, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. 33, 34, 35, 41, 50, 51

[223] Xinran Zhao, Tong Chen, Sihao Chen, Hongming Zhang, and Tongshuang Wu. Beyond relevance: Evaluate and improve retrievers on perspective awareness. *CoRR*, abs/2405.02714, 2024. 57, 63, 65, 71

[224] Zexuan Zhong, Zhengxuan Wu, Christopher Manning, Christopher Potts, and Danqi Chen. MQuAKE: Assessing knowledge editing in language models via multi-hop questions. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15686–15702, Singapore, December 2023. Association for Computational Linguistics. 3

[225] Yujia Zhou, Jing Yao, Zhicheng Dou, Ledell Wu, Peitian Zhang, and Ji-Rong Wen. Ultron: An ultimate retriever on corpus with a model-based indexer. *CoRR*, abs/2208.09257, 2022. 55, 58, 73

[226] Yujia Zhou, Zhicheng Dou, and Ji-Rong Wen. Enhancing generative retrieval with reinforcement learning from relevance feedback. In Houda Bouamor, Juan Pino, and Kalika

Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12481–12490, Singapore, December 2023. Association for Computational Linguistics. 55, 56, 61, 73, 74

[227] Tongyao Zhu, Qian Liu, Liang Pang, Zhengbao Jiang, Min-Yen Kan, and Min Lin. Beyond memorization: The challenge of random memory access in language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3373–3388, Bangkok, Thailand, August 2024. Association for Computational Linguistics. 3

[228] Yangbo Zhu, Jamie Callan, and Jaime G. Carbonell. The impact of history length on personalized search. In Sung-Hyon Myaeng, Douglas W. Oard, Fabrizio Sebastiani, Tat-Seng Chua, and Mun-Kew Leong, editors, *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, Singapore, July 20-24, 2008*, pages 715–716. ACM, 2008. 11

[229] Zihao Zhu, Jing Yu, Yujing Wang, Yajing Sun, Yue Hu, and Qi Wu. Mucko: Multi-layer cross-modal knowledge reasoning for fact-based visual question answering. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 1097–1103. ijcai.org, 2020. 86

[230] Honglei Zhuang, Zhen Qin, Rolf Jagerman, Kai Hui, Ji Ma, Jing Lu, Jianmo Ni, Xuanhui Wang, and Michael Bendersky. RankT5: Fine-tuning T5 for text ranking with ranking losses. In Hsin-Hsi Chen, Wei-Jou (Edward) Duh, Hen-Hsen Huang, Makoto P. Kato, Josiane Mothe, and Barbara Poblete, editors, *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 2308–2313. ACM, 2023. 84, 88

[231] Shengyao Zhuang, Houxing Ren, Linjun Shou, Jian Pei, Ming Gong, Guido Zuccon, and Daxin Jiang. Bridging the gap between indexing and retrieval for differentiable search index with query generation. *CoRR*, abs/2206.10128, 2022. 56, 60, 61, 67, 73, 74

[232] Noah Ziems, Wenhao Yu, Zhihan Zhang, and Meng Jiang. Large language models are built-in autoregressive search engines. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 2666–2678, Toronto, Canada, July 2023. Association for Computational Linguistics. 73