DESIGNING TRANSPARENT AND FACTUAL TEXT GENERATION SYSTEMS GROUNDED IN LANGUAGE STRUCTURE

Vidhisha Balachandran

CMU-LTI-24-001

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
www.lti.cs.cmu.edu

Thesis Committee
Yulia Tsvetkov (chair), Carnegie Mellon University
William Cohen, Carnegie Mellon University & Google Research
Emma Strubell, Carnegie Mellon University
Hannaneh Hajishirzi, University of Washington & Allen Institute for AI

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
In Language and Information Technologies

Abstract

Large language models have brought about a shift towards constructing large, general purpose computational models of language, moving away from task-specific architectures. These models, trained on massive unstructured data, are opaque and challenging to control by design. Consequently, such data-driven models tend to overfit to spurious artifacts, perform poorly on underrepresented data, and fail in unpredictable ways. Thus, a paradigm shift towards developing trustworthy systems to ensure fairness, accountability, and robustness in their outcomes is essential. In this thesis, I argue that leveraging language structures to design trustworthy systems can facilitate this shift.

This thesis presents methods and solutions that leverage language structure to improve the trustworthiness, transparency, and reliability of large-scale, data-driven language generation models, across various stages of the model pipeline. The thesis is divided into three parts. The first part introduces semantically grounded evaluation measures and analysis to assess the factual reliability of trained language generation models. The second part presents model designs that incorporate inter-sentence structures to promote inductive biases and transparency. Finally, the third part presents techniques that use syntactic structures to generate synthetic, general, high-quality datasets for training robust and factual systems. The thesis highlights the challenges in developing trustworthy language generation models and proposes solutions that utilize language structure to improve their interpretability and factual reliability *by design*.

Contents

Chapter 1

Introduction

Human language has a rich, hierarchical structure that facilitates effective communication between individuals and contributes to the diversity of human culture and expression. In linguistics, this structure has been a critical element in developing general theories for studying and representing languages (Harris, 1963; Gruber, 1965; Fillmore et al., 1976; Talmy, 1983), as well as gaining insights into how different languages reflect various cultures and modes of thinking (Biber et al., 1998; Haspelmath et al., 2005; Lupyan and Dale, 2010).

For decades, Computational Linguistics (CL) and Natural Language Processing (NLP) research has leveraged various levels of language structure like morphology, syntax, semantics, discourse or pragmatics, to computationally process human language. Early years of CL research focused on constructing rule-based grammars to represent sentences using structures like constituency parses or dependency trees (Bach, 1967; Schank et al., 1970; Berwick, 1980). These models enabled the creation of many NLP applications such as translation and information extraction (Bar-Hillel, 1960; Mauldin, 1984; Sondheimer and Weischedel, 1980; Kaji, 1988). As statistical and neural approaches have become more prevalent in NLP, language structure has primarily provided inductive biases for designing models focused on specific tasks and domains (Brown et al., 1990; Pereira and Schabes, 1992; Sarawagi and Cohen, 2004; Hajishirzi et al., 2012; Dyer et al., 2015; Yang et al., 2016; Seo et al., 2016; Strubell et al., 2018).

Recently, with the creation of transformer architectures (Vaswani et al., 2017) and self-supervised training paradigms (Devlin et al., 2019; Liu et al., 2019a), there has been a shift in NLP towards building large, general-purpose computational models of language. These models replace task-specific architectures and inductive biases with a single transformer architecture that is pre-trained on large volumes of unstructured internet data, allowing for direct adaptation to any target task without significant modifications. The advent of large language models (LLMs; Brown et al., 2020; Chowdhery et al., 2022; Zhang et al., 2022a) capable of generating text with human-like fluency, coherence, and realism (Zellers et al., 2020; Ippolito et al., 2020) has significantly improved the performance of several language based applications (chatbots, summarizers, information extractors), impacting fields such as education (Schulten, 2023; Gleason, 2022), healthcare (Patel and Lam, 2023), law (ChatGPT and Perlman, 2022) and science (Stokel-Walker, 2023).

However, these advancements also come with certain challenges. Popular language models are currently trained on large unstructured datasets without grounding in any language structure, knowledge or inductive biases, which makes them difficult to understand (Lipton, 2018; Vellido, 2020; Belinkov et al., 2020) and control (Ziegler et al., 2019; Dathathri et al., 2020) by design. Their design does not incorporate inductive biases regarding the task requirements or intended behavior. Consequently, the resulting models tend to overfit to spurious artifacts (Lipton, 2018; Doshi-Velez and Kim, 2017), exhibit lower accuracy for underrepresented data (Hooker et al., 2019; Chang et al., 2019; Hooker, 2021) and fail unpredictably with minor distribution shifts (Belinkov and Bisk, 2017; Recht et al., 2019). Especially when considering large-scale text generation models, they have been demonstrated to produce biased text (Sheng et al., 2021; Dhamala et al., 2021), containing hallucinations and factual errors (Xiao and Wang, 2021; Maynez et al., 2020). As a result, they are ripe for misuse by inadvertently propagating misinformation, bias and discrimination, especially against disadvantaged users (Barocas and Selbst, 2016), leading to reduced trust in model based applications. To address these issues, a shift towards designing trustworthy systems that are fair, accountable, and robust in their outcomes is imperative.

Such unintended behavior in large language models cannot be attributed to a single source in the model development process. A typical machine learning pipeline involves several stages, including data processing, model design, training, inference, and evaluation, and each stage involves design decisions that can contribute to these issues. In order to scale these pipelines to handle large volumes of data and train models with billions of parameters, inductive biases and specialized designs at each stage are often sacrificed, resulting in models that lack transparency and reliability. For example, unfiltered internet data used for training have been shown to contain toxic text and stereotypes, leading to biased model outputs (Han and Tsvetkov, 2021), and black-box neural models with minimal insight into the model decision making process has hindered their deployment in high-stakes applications (Lipton, 2018). It is necessary to develop solutions that introduce transparency and control at each stage of the model pipeline, though it is unclear whether current data-driven approaches are sufficient for achieving this. I believe that leveraging language structures to design for the required model behavior through inductive biases plays a vital role towards developing trustworthy models with transparency and control.

Structure inherently provides a framework or organization to understand, analyze, and manipulate complex concepts easily and effectively. Such benefits can be realized in NLP models by leveraging language structure as a principled, hierarchical framework to understand model behavior and ensure desired outcomes. Incorporating language structure can allow for the interpretation and control of model behavior yielding benefits of interpretability, robustness, fairness and factuality, without compromising performance and generalizability. Grounding models in language structure has already shown initial potential towards this goal through designing interpretable model components aligned with linguistic theories (Silva et al., 2019; Bastings, 2020; Zhang et al., 2021), using hierarchical concepts for human understandable interpretations (Jin et al., 2019; Kim et al., 2018), analysing data to uncover spurious artifacts (Udomcharoenchaikit et al., 2022; Eisenstein, 2022), controlling factuality of outputs at structured unit level (Balachandran et al., 2022; Mao et al., 2022; Chen et al., 2022) and more. Aggregating these independent strands of research, I identify that within the context of a typical data-driven model pipeline, the use of linguistic structure can offer the following benefits: i) breaking down complex concepts like "bias" or "factuality" into fine-grained linguistic concepts for improved understanding of system requirements ii) using linguistic structure to analyze and construct datasets, enabling the correction of imbalances at scale iii) incorporating inductive biases in model design to align with task definitions, and (iv) utilizing linguistic theories for fine-grained evaluation to measure intended system functionality.

Driven by this belief, this thesis aims to bring linguistic theories and language structure into the world of large-scale NLP, highlighting how language structure can aid in incorporating inductive biases about expected model behavior to produce transparent and dependable NLP systems. Focusing on transparency and factual reliability, I present methods and frameworks, grounded in language structure, to directly address the limitations in existing ways of developing and training text generation systems. At each stage of a NLP model development, I identify critical challenges and present solutions which leverage language structure to improve the interpretability and factual reliability of models.

Developing transparent and factual text generation models requires addressing multiple technical challenges. First, different NLP applications require different types of interpretability to understand their output. For example, while interpretability in text summarization involves understanding the choice of content selection, in dialog modeling it is understanding grounding in dialog history, it is tracing content to source knowledge. One solution fits all approaches often fall short of providing such application specific requirements. Thus, I propose application-specific solutions grounded in various forms of language structure to address the unique requirements of different NLP applications.

Second, the subjective nature of these problems makes it challenging and costly to develop solutions that rely on large annotated data. Collecting human annotations on explanations or factual inconsistencies in text is time-consuming and expensive, and the resulting data often exhibits low agreement among annotators(Pagnoni et al., 2021). Further, such annotations are task-specific and therefore not useful to transfer to different settings (Devaraj et al., 2022). Therefore, the solutions I present here are designed to be data-efficient focusing on unsupervised or distantly-supervised techniques for incorporating interpretability and factual consistency in NLP models.

1.1 Thesis Statement

The central goal of this thesis is to demonstrate that grounding large-scale, data-driven language generation models in language structure can facilitate the development of transparent and reliable models. I present methods and solutions that use different language structures in novel ways to introduce application-specific inductive biases and support the design and evaluation of trustworthy language generation systems. I identify challenges in designing such systems and propose solutions which incorporate various structures, such as sentence-level syntax and semantics, as well as longer document structures and pragmatics, in diverse tasks and applications. The thesis is divided into three parts. The first part introduces semantically grounded analysis and evaluation measures to assess the factual reliability of trained language generation models. The second part presents model designs that incorporate inter-sentence structures to promote inductive biases and transparency. Finally, the third part presents techniques that use syntactic structures to generate synthetic, high-quality datasets for training robust and factual systems. This thesis contributes i) new research directions critical for the design of trustworthy language generation systems; ii) new models that effectively combine benefits from large data and language structure to quantitatively and qualitatively improve their transparency and reliability; and iii) new evaluation protocols and resources for the responsible development of such systems.

1.2 Thesis Overview:

Each stage of a typical machine learning pipeline can contribute to unfair and unreliable model outcomes. It is therefore important to identify issues leading to this behavior and consider responsible ML development principles at each stage (Toreini et al., 2020). At the data level, it is important to analyze and construct datasets that contain only relevant artifacts for the model to learn. At the model level, it is critical to design architectures that facilitate an understanding of the model process and allow verification of desired behavior. While evaluating models, it is important to include diverse measures that can provide fine-grained understanding of limitations of the model in addition to an aggregate measure of reliability.

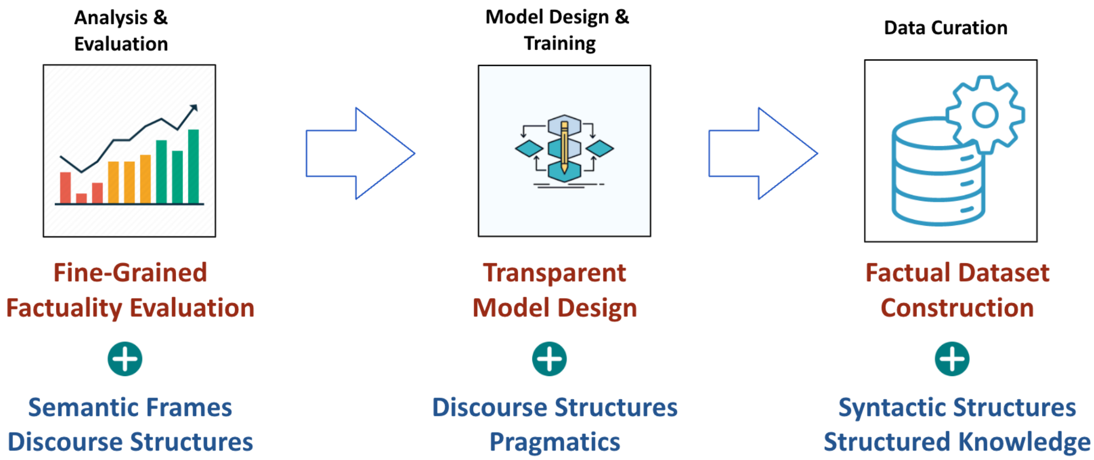

Figure 1.1: Overview of Thesis Outline. Each part focuses on a stage in model development and proposes solutions which leverage structure for improving transparency and factual reliability of language generation models.

Following this, the thesis is organized around different stages of the model pipeline—data analysis, construction, model design and evaluation—and presents research problems and novel solutions in each stage. In the first part of the thesis, I begin with model evaluation of popular trained models, and use the semantic structure of sentences for the analysis and evaluation of the factual reliability of trained language generation models. Using semantic components as fine-grained fact units, I present chapters that i) define evaluation protocols for measuring the factual accuracy of model generated text and ii) describe two fine-grained benchmarks to quantitatively measure the factual accuracy across error types, domains and tasks. This helps set up a common understanding and vocabulary of challenges with reliable generation and provides a platform for presenting and evaluating solutions aimed to address them. Building on the findings, in the second part of the thesis I leverage language structure to incorporate inductive biases and transparency in model design and enable tracing of incorrect model outputs to the source of the issue. To demonstrate the generalizability of the idea, I present prototypes in two settings i) abstractive summarization using discourse structure and ii) dialog modeling using conversation structure. In the final part of the thesis, I go back to the data to develop methods to generate synthetic datasets with intended artifacts. I demonstrate how the syntactic structure of sentences can facilitate the construction of datasets that are controlled for specific features, thereby enabling the training of robust and factual systems. I present two techniques for synthetic data construction i) transforming specific units of sentence for generating adversarial data and ii) constructing entire sentences from structured knowledge for generating pretraining data. These chapters are summarized in detail below:

Part 1 - Language Structure for Trained Model Analysis and Evaluation

I begin by studying current NLP models and presenting fine-grained evaluation techniques, designed using semantic frame, rhetoric discourse structure and pragmatic theories, for measuring the factual reliability of model generated text. Developing computational methods to detect factual errors in any model generated text is a crucial part of a model's evaluation pipeline. Additionally, this can contribute to understanding model limitations and improving them. Initial approaches to develop such methods focused primarily on building binary classifiers to identify factual errors in text. But since they present aggregate binary accuracy their usefulness towards understanding the kinds of factual errors and what impact they have on applications is limited. Pre-trained language generations models are highly fluent and grammatical, and do not make simple syntax based errors. Instead, they often generate syntactically accurate sentences conveying inaccurate meaning. To study them, I leverage higher level semantic and discourse structure to design a fine-grained taxonomy of errors. I use semantic frame, rhetoric discourse structure and pragmatic theories to define factual errors in text and organize them into fine-grained error categories for systematic evaluation. In the two chapters in this part, I introduce the fine-grained error categories and evaluation measures for text summarization (Chapter 2) and extend the taxonomy for studying more complex hallucinations in open-ended generation (Chapter 3).

As a first step towards studying and detecting factual errors in text, we require a general definition of factual errors in text. In Chapter 2 (Relevant Paper: (Pagnoni et al., 2021)), I present a taxonomy of factual errors grounded in grounded in frame semantics (Fillmore et al., 1976; Palmer et al., 2005) and linguistic discourse theory (Brown and Yule, 1983) to enable analysis and evaluation of such fine-grained errors. I collect a human-annotated benchmark to facilitate further research in this direction. Through this collected benchmark, I present a study of factual errors in summarization across different error types, as well as evaluate how well existing factual error detection systems are able to detect such error types. To summarize, this chapter outlines fine-grained evaluation and analysis of factual errors in the task of *Document Summarization* grounded in *Semantic Frames and Rhetoric Discourse Structure*.

A more complex setting for studying the factuality in model generated text is open-ended generation where model generates text or answers questions based on internal knowledge without access to a specific document for grounding. In Chapter 3 (Relevant Paper: (Mishra et al., 2024)), I present a taxonomy for fine-grained evaluation of open-ended generation systems to study the accuracy of their generated content and identify limitations in existing evaluation measures. I extend the previous error taxonomy to include more complex pragmatic errors like invented concepts and unverifiable errors which are more prevalant in this setting. Based on the new taxonomy, I collect a benchmark of human annotated judgements for model generated text across many diverse information seeking queries and propose a detection model to identify such fine-grained errors in generated text. To summarize, this chapter outlines fine-grained evaluation measures, detection models and analysis of factual errors in *Open-Ended Generation* grounded in *Semantic Frames, Pragmatics and Discourse Structure*.

Part 2 - Language Structure for Transparent Model Design

To mitigate the generation of non-factual text, it is essential to verify the correct functioning of the model by analysing model interpretations and attributing model predictions to their source. In the next part, I focus on model design and introduce model components to ensure such transparency by design. Existing work on interpretability focuses mostly on generic post-hoc explanations which have been shown to be counter-intuitive and unfaithful (Feng et al., 2018; Jain and Wallace, 2019), and often not useful for specific domain applications (Ehsan et al., 2021). In this part, I present model architectures that incorporate broader discourse and pragmatic knowledge of the end task to design transparent model components. The models are therefore equipped to provide explanations, grounded in domain knowledge, for model decisions which. Using two diverse, challenging tasks—abstractive summarization and negotiation dialog modeling, I identify language structures that are critical for each task—discourse structure and conversation dependency structure—and design model components that use them as inductive biases. These solutions, though application-oriented, serve as prototypes for designing similar solutions for other applications.

In Chapter 4 (Relevant Paper: (Balachandran et al., 2021)), I present a study on using document structure to introduce content selection interpretability in text summarization. Selecting the right content to summarize is a critical model decision for effective summarization and requires understanding the narrative structure of input (Barzilay and McKeown, 2005). To model this, I introduce a framework that introduces structured document representations into summarization models by: (1) inducing latent sentence graphs to learn discourse structure and (2) incorporating external linguistic structure (e.g., coreference links). We show that incorporating such linguistic structure as inductive biases helps improve the summarization quality as well as provide a means to interpret content selection decisions made by the model. To summarize, this chapter outlines an interpretable model for *Text Summarization* leveraging *Discourse Structure* for introducing transparency.

In Chapter 5 (Relevant Paper: (Joshi et al., 2021)), I consider the task of negotiation dialog and present a study on using dependency structures to introduce dialog and negotiation strategy interpretability in dialog modeling. Learning how to negotiate effectively involves deep pragmatic understanding and planning the dialogue strategically (Thompson, 2001). I present an end-to-end negotiation dialogue system that leverages Graph Attention Networks to model complex negotiation strategies while providing interpretability for the model via intermediate structures. This enables the model to learn complex utterance and negotiation structure, tailored for the particular negotiation task, and leverage it for predicting the next sequence and for interpreting the learnt dependencies for verification. To summarize, this chapter outlines an interpretable model for *Dialog Modeling* leveraging pragmatic structures like *Negotiation Strategies and Utterance Graphs* for introducing transparency.

Part 3 - Language Structure for Controlled Dataset Construction

Finally, in the last part I focus on data pipelines to mitigate unintended artifacts in datasets. By leveraging syntactic structures in text, I construct synthetic datasets that allow for control over the features that the model is expected to learn. With the large scale data used to train language models, it becomes impossible to analyze and verify the artifacts and content being provided as input to models (Liang et al., 2022). Consequently, models pick up and amplify unintended biases and toxic content from such training data. A popular emerging direction of research to alleviate this problem is the generation of synthetic data, which can be controlled to have certain specific features and properties (Marzoev et al., 2020; Wu et al., 2022a; Krishna et al., 2021). Most approaches use logical or template based methods to construct such data, limiting the usefulness and generalizability of data. Instead, I propose to leverage the syntactic structure of language along with the flexibility of language models to construct natural and realistic synthetic data, increasing their generalizability while still maintaining control in design. In the two chapters in this part, I present two data generation techniques, where I use syntactic units to carefully edit sentences (Chapter 6) and use sentence syntax structure to construct entire sentences (Chapter 7) producing realistic data which closely resembles human-language structure, but with the ability to control the features present in the dataset.

Curating training data for factual error correction models is challenging as collecting human-annotations is subjective and expensive (Pagnoni et al., 2021) and constructing heuristic synthetic data is not generalizable (Cao et al., 2020). In Chapter 6 (Relevant Paper: (Balachandran et al., 2022)), I present a new approach for generating diverse synthetic data for this task. By decomposing existing reference summaries into their smaller syntactic units, I use infilling language models to replacing each factually correct syntactic unit in a reference summary, producing a set of plausible, likely, and fluent, incorrect synthetic summaries. In this way, I combine the benefits of language structure and capabilities of large language models for fine-grained transformation of summaries. I experimentally show how this diverse training data improves factual error correction across a wide range of settings. To summarize, this chapter outlines a training data generation method for the task of *Factual Error Correction* leveraging the *Syntactic Structure* of language to introduce control and quality in the generation process.

In Chapter 7 (Relevant Paper: Feng et al. (2023a)), I propose a fact-aware pretraining approach trained on synthetically generated data to improve representing fact units for error detection. Using external knowledge bases as a repository of world facts, I aim to leverage the syntactic structure of sentences to construct entire natural language like sentences from structured knowledge triples. I propose a fact-aware pretraining approach, describing multiple factuality based objectives to pretrain language models using the synthetic corpus and improve fact representation for tasks like error detection and fact checking across news and scientific domains. To summarize, this chapter outlines a pre-training data generation method for the setting of *Factual Error Detection* leveraging the *Syntactic Structure* of language to introduce control and quality in the generation process.

Part I

Language Structure for Fine-Grained Model Evaluation

Chapter 2

Definition and Analysis of Factual Errors in Summarization with Semantic Frame Structure

Model Evaluation is an extremely important component of the model development pipeline. Thorough evaluation enables understanding capabilities and limitations of models, charting of progress towards certain goals as a community and guiding effective efforts towards addressing limitations of models. Today, model evaluation is predominantly done via benchmarks which provide high level aggregate measures on various tasks, which is useful for model comparison but does not provide useful feedback on model limitations or to guide model development work. In this context, fine-grained model analysis and evaluation plays an important role to fill this gap. But such fine-grained evaluations are often done via manual inspections or qualitative analysis which are not scalable. In this section of the thesis, I propose to leverage language structure as an inductive bias to develop scalable fine-grained evaluation protocols and I demonstrate this by evaluating factual consistency in language generation as a use case.

Modern text generation models generate highly fluent but often factually unreliable outputs. This chapter considers the task of automatic text summarization and describes a typology of error types drawn from semantic frame theory and rhetoric discourse structure theory to support studying and characterizing the range of factual errors in summarization models. Based on the typology, the chapter presents a benchmark of human annotated factuality judgements of model generated summaries and presents a study identifying the proportion of different categories of factual errors in various summarization models and benchmark factuality metrics, showing their correlation with human judgement as well as their specific strengths and weaknesses. Research described in this chapter was conducted in collaboration with Artidoro Pagnoni and Yulia Tsvetkov and was presented in a NAACL 2021 publication (Pagnoni et al., 2021).

2.1 Introduction

Factuality is defined as a measure of "whether eventualities are characterized as corresponding to facts, possibilities, or situations that do not hold in the world" (Sauri, 2008; Saurí and Pustejovsky, 2012). In summarization, this "world" is the article, which is taken as ground-truth, and the output summary must be faithful to the article's facts. Despite advancements in neural abstractive summarization (Narayan et al., 2018; Liu and Lapata, 2019; Lewis et al., 2019), ∼30% of summaries have factual inconsistencies (Cao et al., 2018a). With summarization being an integral component of information consumption, this highlights a need for ensuring summarization systems are factually consistent and developing methods for evaluating them.

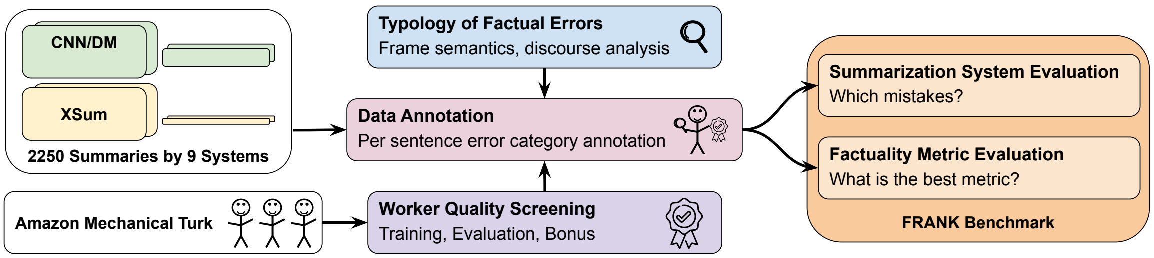

Figure 2.1: We propose a linguistically grounded typology of factual errors. We select crowd workers to annotate summaries from two datasets according to this typology achieving near perfect agreement with experts. We collect FRANK, the resulting dataset, to benchmark factuality metrics and state-of-art summarization systems.

Common evaluation metrics for summarization based on n-gram overlap – BLEU, ROUGE, and METEOR (Papineni et al., 2002; Lin, 2004; Lavie and Agarwal, 2007) – are insufficient to measure the factual correctness of summaries and fail to correlate with the human judgements of factuality (Falke et al., 2019; Kryscinski et al., 2019). More recent metrics proposed to improve the evaluation of summarization factuality (Kryscinski et al., 2020; Durmus et al., 2020; Wang et al., 2020a; Maynez et al., 2020) cannot be compared due to the lack of common benchmarks. More critically, while these approaches differ in the way they model factuality, they all consider factuality as a binary concept, labeling summaries of any length as factual or non-factual. They do not provide any fine-grained understanding of the factual errors made by different systems that could serve as an actionable feedback on a system's limitations.

The binary factuality of a text can be difficult to determine. Falke et al. (2019) show relatively low crowd–expert agreement, indicating the presence of subjectivity in the annotation process. Moreover, not all factual errors are equally important and the number of errors can have a significant impact on the perceived factuality of a text. This suggests that non-factuality should be modeled as a multi-dimensional construct and not a label.

In this chapter, we propose a linguistically motivated typology of factual errors for fine-grained analysis of factuality in summarization systems (§2.2). Our typology is theoretically grounded in language structure: frame semantics (Fillmore et al., 1976; Palmer et al., 2005) and linguistic discourse theory (Brown and Yule, 1983) and provides several benefits. First, we find that decomposing the concept of factuality in (relatively) well-defined and grounded categories makes the final binary decision more objective leading to significantly high agreement between crowd and expert annotators ($\kappa = 0.86$). Second, this approach provides some measure of the degree of non-factuality both in terms of the quantity and the category of factual violations that appear in the text. This typology also provides us with the means to categorize the types of errors made by summarization systems, helping us gain deeper insights than simply categorizing content as factual or hallucinated.

We define an annotation protocol of factuality based on our typology and collect a benchmark of human annotated judgements – FRANK – over a diverse set of model generated summaries on the CNN/DM (Hermann et al., 2015) and XSum (Narayan et al., 2018) datasets (§2.3). Through this dataset, we aim to both assess the factuality of summarization systems and benchmark recently proposed factuality metrics. In §2.4 we discuss various state-of-art models and show a detailed analysis of the factual errors they make. Finally, in §2.5 we evaluate multiple summarization metrics against our benchmark and show their strengths and weaknesses in detecting specific types of factual errors. Figure 2.1 shows an overview of this work. [1]

[1] Code, data, and online leaderboard will be available at https://github.com/artidoro/frank

	Category	Description	Example
PredE	Relation Error	The predicate in the summary statement is inconsistent with the source article.	*The Ebola vaccine **was rejected** by the FDA in 2019.*
EntE	Entity Error	The primary arguments (or their attributes) of the predicate are wrong.	*The **COVID-19 vaccine** was approved by the FDA in 2019.*
CircE	Circumstance Error	The additional information (like location or time) specifying the circumstance around a predicate is wrong.	*The first vaccine for Ebola was approved by the FDA in **2014**.*
CorefE	Coreference Error	A pronoun/reference with wrong or non-existing antecedent.	*The first vaccine for Ebola was approved in 2019. **They** say a vaccine for COVID-19 is unlikely to be ready this year.*
LinkE	Discourse Link Error	Error in how multiple statements are linked together in the discourse (for example temporal ordering/causal link).	*To produce the vaccine, scientists have to show successful human trials, **then** sequence the DNA of the virus.*
OutE	Out of Article Error	The statement contains information not present in the source article.	***China** has already started clinical trials of the COVID-19 vaccine.*
GramE	Grammatical Error	The grammar of the sentence is so wrong that it becomes meaningless.	*The Ebola vaccine **accepted have already started.***

Table 2.1: Typology of factual errors. Original text for the examples: *The first vaccine for Ebola was approved by the FDA in 2019 in the US, five years after the initial outbreak in 2014. To produce the vaccine, scientists had to sequence the DNA of Ebola, then identify possible vaccines, and finally show successful clinical trials. Scientists say a vaccine for COVID-19 is unlikely to be ready this year, although clinical trials have already started.*

2.2 Frame-Centric Factual Error Benchmark - FRANK

Previous studies of factuality in summarization only distinguish factual and hallucinated content (Kryscinski et al., 2019; Maynez et al., 2020) and provide limited insights on the fine-grained types of factual errors. In the simplest case, factual errors appear within a single proposition. However, as summaries include several sentences, discourse markers describe relations across propositions. These cross-sentence links, such as causality or temporal ordering, can introduce inconsistencies with the article. Furthermore, information in the summary should be verifiable given the article. This understanding outlines different levels of linguistic structure where factual mistakes can arise in summaries: at the semantic frame level, at the discourse level, or because the content cannot be verified. Below we define a typology of factual errors further detailing these three levels. This typology is theoretically grounded in frame semantics (Fillmore et al., 1976; Baker et al., 1998; Palmer et al., 2005) and linguistic discourse analysis (Brown and Yule, 1983). Examples for each category are shown in Table 2.1.

2.2.1 Semantic Frame Errors

A *semantic frame* is a schematic representation of an event, relation, or state, which consists of a predicate and a list of participants, called frame elements Baker et al. (1998). A semantic frame has both core and non-core frame elements (FE). Core frame elements are essential to the meaning of the frame, while non-core (e.g. location, time) provide additional descriptive information. Our first three categories capture factual errors in each of these components (frame, core and non-core FE) respectively.

Predicate Error (PredE): Category *PredE* encompasses errors where the predicate in a summary statement is inconsistent with the source text. More generally, this represents cases where the frame from a summary statement does not align with what is expressed in the source text.

Entity Error (EntE): Category *EntE* captures errors where the primary arguments (like entities) of the predicate are wrong or have the wrong attributes, although the relation was expressed in the original text. More generally, these account for cases where the core frame elements in a frame are wrong. This also captures directionality errors where the elements are interchanged (similar to agent-patient swap).

Circumstance Error (CircE): In additional to the core arguments, predicates can be further specified using additional information or attributes that describe the circumstance in which the arguments and predicates interact (e.g. location, time, manner, direction, modality). Category *CircE* captures errors where one or more such attributes (non-core frame elements within a frame) are wrong.

2.2.2 Discourse Errors

The communicative intent of an author is also expressed through relations that hold between parts of the text. Factual errors in summarized text can often extend beyond a single semantic frame introducing erroneous links between discourse segments. Below we outline such categories of errors which are grounded in discourse analysis and rhetorical structure theory (RST) (Brown and Yule, 1983; Mann and Thompson, 1988). RST is an elaborate system for annotating coherence relations in discourse. Some examples of such relations include: "Elaboration", "Background", "Motivation", and "Volitional Cause". Here we depart from semantic frame terminology as its rooting in a single frame does not allow us to represent such errors.

Coreference Error (CorefE): Category *CorefE* accounts for errors where pronouns and other types of references to previously mentioned entities either are incorrect or have no clear antecedents, making them ambiguous.

Discourse Link Error (LinkE): Category *LinkE* encompasses errors involving a discourse link between different statements. These include errors of incorrect temporal ordering or incorrect discourse links (e.g. RST relations, discourse connectors) between statements.

2.2.3 Content Verifiability Errors

Facts should be be verifiable given the article: the information should be contained in the article, and they should be well formed. Often statements in a summary cannot be verified against the source text due to difficulty in aligning them to the source. Below we outline two categories of errors for such cases.

Out of Article Error (OutE): Since summaries of a document should only contain information that can be deduced from the original text, we include a category for such errors *OutE* (prior work refers to this as extrinsic hallucinations (Maynez et al., 2020)).

Grammatical Error (GramE): We use *GramE* to categorize statements that are not well formed. When grammatical mistakes make the meaning of a statement incomprehensible or ambiguous, it cannot be verified against the source and is thus considered trivially wrong. Minor grammatical errors are acceptable.

Finally, for completeness in our annotation exercise, we add two additional categories **Others (OthE)** for factually errors that do not correspond to any of the above categories and **Not an Error (NE)** for statements that do not contain any errors.

2.3 Dataset Creation

Beyond theoretical grounding, we empirically verify our typology through large scale human annotations of five abstractive summarization models on the CNN/DM dataset and four on the XSum dataset. Through our dataset, we aim to have a broad coverage of different types of errors made by neural summarization systems, with human judgements on their fine-grained factuality errors.

Annotation Data For the annotation, we include model summaries from CNN/DM and XSum datasets as they present different characteristics. CNN/DM summaries are longer, with three sentences on average, while XSum has only single sentence summaries. Having longer summaries is crucial to identify discourse level errors. On the other hand, XSum summaries are more abstractive and include more factual errors on average (Maynez et al., 2020). For a diverse set of model summaries, we collect publicly available model outputs from different summarization models with differing factuality capabilities.

On the CNN/DM (Hermann et al., 2015) dataset we use five different models. We use the preprocessed model outputs provided by Fabbri et al. (2020).
S2S: an LSTM based Sequence-to-Sequence with attention model (Rush et al., 2015)
PGN: an LSTM based Pointer-Generator Network with Copy Mechanism (See et al., 2017)
BUS: Bottom-Up Summarization (Gehrmann et al., 2018) - a Pointer-Generator model with a data-efficient content selector to over-determine phrases in a source document that should be part of the summary.
BERTSum: summarization with pretrained encoders (Liu and Lapata, 2019)
BART: pretrained transformer based encoder-decoder model (Lewis et al., 2019)

On the XSum dataset (Narayan et al., 2018) we use four different models. All model outputs for this dataset are taken from (Maynez et al., 2020).
PGN: pointer-generator network from above (See et al., 2017)
TConvS2S: Topic-Aware Convolution Sequence-to-Sequence (Narayan et al., 2018)
TranS2S: A randomly initialized Transformer (Vaswani et al., 2017) encoder-decoder model fine-tuned on the XSum dataset
BERTS2S: Transformer encoder-decoder model with parameter sharing (Rothe et al., 2020) where both encoder and decoder are initialized with the BERT-Base checkpoints (Devlin et al., 2019) and fine-tuned on XSum

Annotation Collection Using the above model generated summaries, we collect human annotations from three independent annotators for 250 articles from each dataset (with a total of 1250 model outputs on CNN/DM and 1000 on XSum). We annotate each sentence of a summary to break the judgement of factuality into smaller units. We present sentences in the context of the entire summary to identify discourse errors spanning multiple sentences. Annotations are a two step process: for each sentence in the summary, the annotator first selects whether the sentence is factual, and if marked not factual, identifies the category of each error based on our typology. [2] A sentence can be annotated with more than one category of errors to account for multiple errors within a sentence.

We conduct the annotation task on the Amazon Mechanical Turk (MTurk) platform. To achieve high quality crowd-sourced annotations, we build an intuitive interface[3]. Screenshots of the interface are shown in Appendix 1.1.1. We outline specific highlights of our annotation protocol below:

1. **Clear Instructions:** We explain the annotation scheme without assuming linguistic knowledge and give several examples for each category. We also provide a practical step-by-step to determine the category of the errors.
2. **Training:** Every first-time user has to go through a tutorial which exercises the comprehension of the annotation scheme. The tutorial presents an article and several hand-crafted summaries of the article that need to be annotated. It is designed to be very similar to the actual annotation task and to contain at least one occurrence of each category of error. Feedback is provided when a user selects the wrong category of error. This tutorial is not used to evaluate users, only to help them understand the different categories in a practical setting.
3. **Qualification test:** To participate in the annotation, users have to obtain a minimum score of 85% on a qualification test. The test comprehends an article and several summaries to be annotated. It contains at least one instance of each category of error. We use this test to verify that users can effectively recognize error categories. This ensures that users are able to perform the task correctly, but does not enforce that high standards of work quality are maintained throughout the annotation task.
4. **Continuous evaluation:** We continuously evaluate a user by verifying that they read the text. For every article that is annotated, we ask to identify one of three entities that was not present in the article. We also monitor the annotations on artificially altered sentences that are randomly inserted at the end of summaries. Wrong sentences contain one of the following errors: negation of declarative sentences (PredE), pronoun swap (CorefE), sample sentence from another article (OutE), word scrambling (GramE). We immediately block users that fail the entity test or perform poorly on these sentences (less than 50% of correct answers on altered sentences) to ensure high quality annotations.
5. **Fair pay and Bonuses:** All workers are paid 50% more than the average American minimum wage but we offer bonuses for scores of 60% or above on the continuous evaluation, and for completion a sequences of 10 annotations. We observe that bonuses increase the percentage of users with high continuous evaluation scores (<10% blocked users with bonuses versus 30% without bonuses).

Inter-Annotator Agreement: We report inter-annotator agreement in terms of Fleiss Kappa κ (Fleiss, 1971). Following Durmus et al. (2020), we report the percentage p of annotators that agree with the majority class. Each datapoint in our dataset corresponds to a sentence in a summary. We compute agreement on all 4942 annotated sentences. On the annotation of whether a sentence is factual or not we obtain $\kappa = 0.58$, with $p = 91\%$ of annotators agreeing with the majority class. As a comparison, Durmus et al. (2020) reports $p = 76.7\%$ average agreement. When all three annotators agree that a sentence is not factual, we obtain $\kappa = 0.39$ with $p = 73.9\%$ of annotators agreeing with the majority class on the eight category annotation (seven categories of errors and "other") which indicate a moderate agreement.

[2] We experimented with Likert scale evaluation of full summaries in a pilot study. Such an annotation would not provide precise information about where in the summary an error appears and also resulted in lower agreement. Hence, we opted for sentence level judgements.

[3] We make the interface available for future human annotations that follow our typology

Agreement with Domain Expert: We measure agreement between the majority class of the three annotators and one expert annotator on 201 datapoints (10 summaries from CNN/DM and 10 summaries from XSum). We find a Cohen Kappa of $\kappa = 0.86$ indicating nearly perfect agreement. Previous work found agreement of $\kappa = 0.65$ between three crowd annotators and expert annotations of factuality (Falke et al., 2019). Even with more than nine workers, they report agreement with expert annotations of at most $\kappa = 0.74$. This improvement validates the robustness of our annotation interface and protocol which achieves higher agreement with fewer workers.

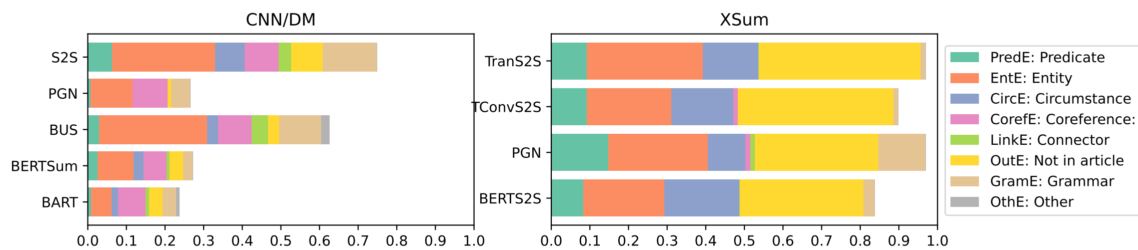

Figure 2.2: Proportion of summaries with factual errors based on collected annotations, with breakdown of the categories of errors within. Full specification of categories of errors in Table 2.1.

2.4 Summarization Model Analysis

We evaluate the performance of different summarization models in terms of factuality. Figure 2.2 visualizes the percentage of summaries with factual errors for each category model and dataset, with a breakdown of proportion of different error types within each. A summary is considered incorrect if it contains at least one sentence with a factual error. A sentence contains a factual error if the majority of annotators indicate the presence of an error (here we do not consider annotations where all three annotators disagree on the category).

How factual are generated summaries across different datasets? From our annotations, we observe that 60% of the summaries that were annotated contain at least one factual error. From Figure 2.2, we see that the XSum dataset has more factually incorrect model summaries (92%) than CNN/DM (43%). It poses more significant challenges in terms of factuality as all models produce > 80% summaries with factual errors, with the best model (BertS2S) producing 83% wrong summaries. On the CNN/DM dataset, while state-of-the-art pretrained models like BERTSum and BART have better factuality numbers, the percentage of factually incorrect summaries is still high (23% for BERTSum and 27% for BART). The proportion of errors across different categories vary widely between the two datasets. For the CNN/DM dataset, the most frequent classes of errors are Entity Error (EntE) and Coreference Error (CorefE). For the XSum dataset they are Out of Article Error (OutE) and Entity Error (EntE). Note that there are no discourse errors (CorefE, LinkE) in the XSum dataset because the data only contains single sentence summaries. Additionally, we observe that OthE makes up a very small percentage ($\sim 1\%$) of errors overall showing that our typology is *nearly complete* with most errors being mapped to one of our existing categories.

How factual are generated summaries across different models? From Figure 2.2, we observe that LSTM based models like S2S and BUS generate many incorrect summaries. Interestingly, PGN on CNN/DM has fewer summaries with factual errors (26%) compared to S2S (74%) and BUS (62%) potentially due to the extractive nature of CNN/DM and the copy based objective in PGN. PGN has been previously shown to produce highly extractive summaries on CNN/DM copying large portions of text (often entire sentences) (Gehrmann et al., 2018; Balachandran et al., 2021). On the more abstractive dataset XSum, PGN produces $> 96\%$ factually incorrect summaries. We also observe that large-scale pretrained models improve factuality on both datasets, as also noted by Durmus et al. (2020), with more significant gains on CNN/DM. On CNN/DM, BERTSum and BART display half the error rate of BUS. In contrast, on XSum, BertS2S improves over non-pretrained models by $\sim 10\%$ only, showing that XSum poses a significant challenge for factuality even in pretrained models.

Different models also exhibit different distributions in the error categories. LSTM based models have higher proportion of Grammatical Errors (GramE) while transformer and CNN based models have a lower proportion. For pretrained transformer models, we observe that the improved error-rate on the CNN/DM dataset can be attributed to improvements at the frame level (PredE, EntE, CircE) while the discourse level errors still remain a challenge. Errors CorefE, LinkE account for a higher proportion of errors in BERTSum and BART compared to the other models.

2.5 Factuality Metric Evaluation

We propose the FRANK dataset resulting from the human annotation study as a common benchmark to assess different factuality metrics. We provide an evaluation protocol of factuality metrics, which controls for dataset biases, and a fine grained analysis of the strengths of each metric.

2.5.1 Benchmark

The FRANK benchmark provides a diverse dataset for evaluating various metrics on their ability to capture factual errors. Notably, our benchmark has *factual error diversity*, as it covers all types of errors described in the typology in §2.2, and *data diversity* as it combines 2250 summaries from different systems and datasets. Our annotations go beyond binary labels of factuality on a summary by providing fine-grained category annotations for every sentence. This allows us to determine how well each metric can capture each type of error. Furthermore, through averaging of sentence level judgements, we can also obtain a factuality scores (0 to 1 range) for a summary. To measure the degree that automated metrics capture a certain characteristic, we compute their correlation with human judgements and report Pearson correlation and Spearman rank correlation along with their p-values.

We evaluate different classes of metrics against the FRANK benchmark. We select four general summarization metrics. ROUGE, BLEU, and Meteor are n-gram based metrics and computed with respect to the reference summary. We also select the following model based metrics focused on factuality:

BERTScore (Zhang et al., 2020b): We report BERTScore Precision, Recall, and F1 between the model output and the article being summarized. Our experiments show that recall and F1 do not correlate as well with the human judgement of factuality for BERTScore.

OpenIE : We use a simple baseline based on OpenIE (Banko et al., 2007) and Sentence-BERT (Reimers and Gurevych, 2019). We use OpenIE (Banko et al., 2007) to extract subject-relation-object triplets from the article, reference summary, and model generated summary. We consider binary relations only and thus use the

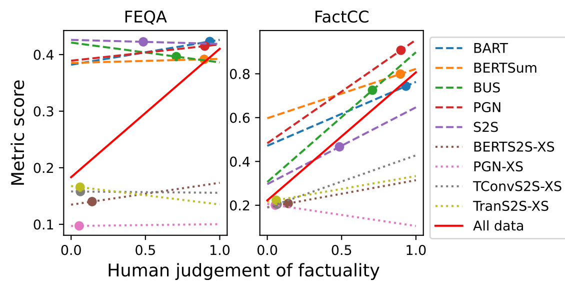

Figure 2.3: Correlation between metrics and human judgement on subsets of data. The x and y axis represent the human judgement the metric scores respectively. The red line is a linear regression fitted on full data. Each dotted line is a linear regression fitted on a model-dataset subset. Each colored point has coordinates equal to average factuality judgement, and metric score for its corresponding partition.

first two arguments of the relation.[4] After replacing corefering entity mentions with the main mention of the cluster[5], we use BERT base Sentence-BERT (Reimers and Gurevych, 2019) to obtain embeddings of each element of the subject-relation-object triplets extracted by OpenIE. Two relation triplets are considered to be equivalent if their embeddings have cosine similarity higher than a threshold for all three elements of the triplet (we use 0.6 as threshold after a grid search between 0.5 and 0.9 on data from our pilot study).

FEQA Durmus et al. (2020): FEQA is a question generation and answering (QGA) factuality metric. We relied on the original implementation of the authors for this metric as well as their pre-trained model weights. We used the full summary to generate questions and we answer them both using the summary and article text.

QAGS Wang et al. (2020a): QAGS is another QGA metric. The authors kindly provided outputs on the FRANK benchmark generating 10 questions for each summary.

DAE Goyal and Durrett (2020): DAE is an entailment classification metric that operates on dependencies. The authors kindly provided outputs on the FRANK benchmark. We note that the model was trained with a max length of 128 after concatenating both article and summary. The CNN/DM articles can be significantly longer, thus the results reported for this metric involve truncating parts of the article.

FactCC Kryscinski et al. (2020): FactCC is an entailment classification metric. We use the sentences of the model generated summary as input claims to the entailment classifier FactCC. For each sentence we obatain a binary factuality label. We take the average of these labels as the factuality score for the summary.

2.5.2 Controlling for Dataset Biases

Since our benchmark contains diverse summaries from different datasets and models, dataset biases can hamper accurate reporting. In Figure 2.3, we visually show correlations between two factuality metrics (FEQA and FactCC) and human judgement on the entire data and on partitions of the data. For both metrics, we notice that the slope (an unscaled measure of correlation) of the line fitted through the entire data (red line) is significantly larger. In FEQA, the dotted lines (fitted on subsets of the data of each model and dataset) are almost horizontal. This likely indicates the presence of a confounding variable associated with the properties

[4]We use the model and implementation from Stanovsky et al. (2018) for OpenIE extraction.
[5]https://github.com/huggingface/neuralcoref

Metrics	All data				CNN/DM				XSum			
	Pearson		Spearman		Pearson		Spearman		Pearson		Spearman	
	ρ	p-val	r	p-val	ρ	p-val	r	p-val	ρ	p-val	r	p-val
BLEU	0.10	0.00	0.07	0.00	0.08	0.01	0.08	0.01	0.14	0.00	0.20	0.00
METEOR	0.14	0.00	0.11	0.00	0.12	0.00	0.10	0.00	0.15	0.00	0.10	0.00
Rouge-1	0.14	0.00	0.10	0.00	0.12	0.00	0.10	0.00	0.15	0.00	0.09	0.01
Rouge-2	0.12	0.00	0.08	0.00	0.08	0.00	0.07	0.01	0.17	0.00	0.14	0.00
Rouge-L	0.13	0.00	0.09	0.00	0.11	0.00	0.09	0.00	0.16	0.00	0.10	0.00
OpenIE	0.11	0.00	0.02	0.36	0.16	0.00	0.15	0.00	0.00	0.93	-0.45	0.00
BERTS P	**0.27**	0.00	0.24	0.00	0.35	0.00	0.29	0.00	**0.18**	0.00	0.09	0.00
BERTS R	0.14	0.00	0.13	0.00	0.21	0.00	0.17	0.00	0.07	0.03	0.03	0.38
BERTS F1	0.24	0.00	0.21	0.00	0.32	0.00	0.26	0.00	0.15	0.00	0.06	0.05
FEQA	0.00	0.83	0.01	0.60	-0.01	0.76	-0.01	0.72	0.02	0.45	0.07	0.04
QAGS	0.06	0.00	0.08	0.00	0.13	0.00	0.09	0.00	-0.02	0.48	0.01	0.65
DAE	0.16	0.00	0.14	0.00	0.25	0.00	0.24	0.00	0.04	0.16	**0.28**	0.00
FactCC	0.20	0.00	**0.30**	0.00	**0.36**	0.00	**0.33**	0.00	0.07	0.02	0.25	0.00

Table 2.2: Partial Pearson correlation and Spearman rank correlation coefficients and p-values between human judgements and metrics scores. Comparisons should be made along with the pairwise Williams test found in Table 2.3.

of each system and dataset. This can lead to false measures of high correlation if not accounted for. To address this, we suggest to control for confounding variables using partial correlations.

Partial correlation measures the degree of association between two random variables, with the effect of a set of controlling random variables removed. Although we are unaware of the exact confounding variable, we use the categorical variable C of which system and dataset the summary was generated from.

Let M_k represent the output of metric k on the summaries. To compute partial correlation between M_k and human judgements H which we treat as random variables, we solve the two regression problems $M_k|C = c \sim w_{M_k}c$ and $H|C = c \sim w_H c$ and get the residuals:

$$\Delta M_k = M_k - \hat{w}_{M_k}C$$

$$\Delta H = M_k - \hat{w}_H C$$

And then calculate the correlation between these residuals $\rho(\Delta M_k, \Delta H)$ instead of the original random variables. Since partial correlations are proper correlations between random variables, we can apply statistical significance tests without any modification. In this case, both the system and the dataset are taken to be confounding variables.

2.5.3 Results

In Table 2.2, we report the partial Pearson correlation and Spearman rank correlation coefficients with human judgements for each metric, along with their p-values indicating statistical significance.

How do different metrics correlate with human judgements? From Table 2.2 we observe that all metrics exhibit low correlations with human judgements of factuality. The best metrics overall are FactCC with 0.20 Pearson and 0.30 Spearman correlation and BERTScore P with 0.27 Pearson and 0.35 Spearman correlation. Interestingly, we observe that general summarization metrics BLEU, Rouge, and METEOR, and the OpenIE

	B	MET	R-1	R-L	BS-P	OpIE	FEQA	QAGS	DAE	FCC
BLEU	-	0.82	0.77	0.85	0.12	0.25	0.03	-0.02	0.05	0.06
METEOR	0.82	-	0.87	0.85	0.17	0.27	0.02	-0.02	0.09	0.07
Rouge-1	0.77	0.87	-	0.89	0.22	0.21	0.01	-0.03	0.09	0.07
Rouge-L	0.85	0.85	0.89	-	0.18	0.21	0.01	-0.04	0.08	0.07
BERTS P	0.12	0.17	0.22	0.18	-	0.20	0.01	0.06	0.18	0.27
OpenIE	0.25	0.27	0.21	0.21	0.20	-	-0.01	0.09	0.10	0.15
FEQA	0.03	0.02	0.01	0.01	0.01	-0.01	-	-0.01	0.03	0.04
QAGS	-0.02	-0.02	-0.03	-0.04	0.06	0.09	-0.01	-	0.07	0.10
DAE	0.05	0.09	0.09	0.08	0.18	0.10	0.03	0.07	-	0.10
FactCC	0.06	0.07	0.07	0.07	0.27	0.15	0.04	0.10	0.10	-

Table 2.3: Pearson correlation between metrics. If value is in green, the metrics are not the same significant to the 0.05 threshold with the Hotelling Williams test.

baseline have statistically significant correlations with factuality, close to FactCC ($\rho = 0.14$ for Rouge-1 and METEOR versus $\rho = 0.20$ for FactCC). The entailment metrics (FactCC and DAE) and contextual embedding method (BERTScore) have the highest correlations and are statistically significant. The two QGA metrics have lower overall correlation. FEQA's correlation is not statistically significant. QAGS has low, but significant correlation of $\rho = 0.06$.

How well do different metrics capture errors in different datasets? In Figure 2.4, we observe that entailment metrics have significantly higher partial Pearson correlation on the CNN/DM dataset than XSum where their correlation is reduced by a factor of four. QAGS and the OpenIE baseline have similar behavior. This suggests that these metrics capture the error types from CNN/DM better that those from XSum. Specifically, XSum has uniquely high Out of Article (OutE) errors which they might not capture well. This also highlights the importance of data diversity in building and benchmarking factuality metrics to avoid overfitting to certain types of errors. The correlation numbers in Table 2.2 should be read in combination with the pairwise Hotelling-Williams test Graham (2015) results in Table 2.3. The highlighted numbers indicate pairs of models for which the difference in correlation is statistically significant. We use partial correlations to run the test and compute metric-metric correlations.

How well do different metrics capture errors from pretrained and non-pretrained models? On the CNN/DM dataset we observe that entailment metrics and QAGS perform significantly better on non-pretrained models. This indicates that the artificial factual errors on which entailment metrics are trained on are closest to the mistakes that non-pretrained models make. This also suggests that the errors made by pretrained models might be more difficult to capture by these metrics. These trends are less clear on the XSum dataset which we again attribute to high Out of Article (OutE) errors in the pretrained and non-pretrained models (Figure 2.2)

2.5.4 Error Analysis

Figure 2.4 shows partial Pearson correlation on six subsets of the data. To understand capabilities of metrics across the broad categories of errors (semantic frame errors, discourse errors, and content verifiability errors) we perform an ablation study. For each category, we compute the variation in partial correlation with errors from that category omitted. In Figure 2.5, we visualize the influence of a given type of error using the variation

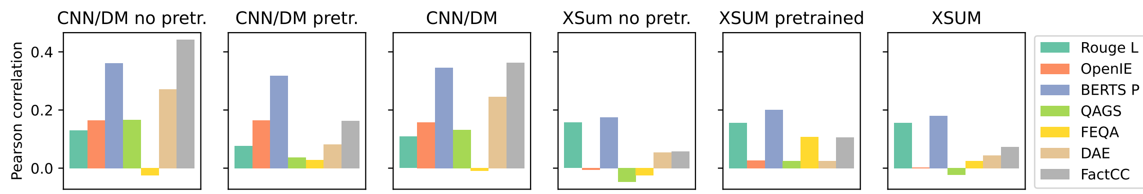

Figure 2.4: Partial Pearson correlation on different partitions of the data. Entailment metrics have highest correlation on pretrained models in the CNN/DM dataset. Their performance degrades significantly on XSum.

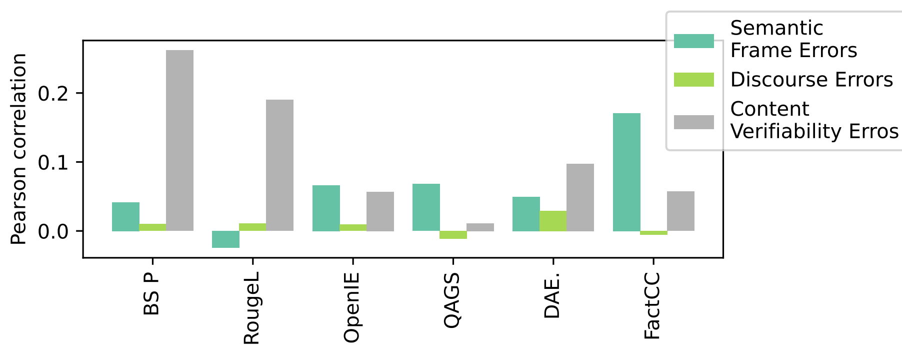

Figure 2.5: Variation in partial Pearson correlation when omitting error types. Higher variation indicates greater influence of an error type in the overall correlation.

for each metric and category. A higher positive bar indicates that the error type was a significant contributer to the overall correlation (or metric highly correlates with error) causing the correlation without it to drop.

General Summarization metrics Unsurprisingly, we observe that Rouge L is best correlated with content verifiability errors (which contains Out of Article Errors) as n-gram matches detect them. Rouge L has negative correlation with semantic frame errors and low correlation with discourse level errors indicating that n-gram matching fails to capture them. We observe that OpenIE is more correlated with semantic frame errors. The metric matches entities and verifies the predicate that relates them and hence is able to capture semantic frame errors. We observe that BERTScore's high correlation is primarily due to its ability to capture content verifiability errors while it has low correlation on semantic frame errors.

QGA metrics Both QGA metrics have negative correlation with discourse errors suggesting that QGA metrics are not able to capture coreference errors or discourse link errors potentially due to the entity oriented questions in their training data. FEQA additionally is also negatively correlated with semantic frame errors and has low positive correlation with content verifiability errors. In contrast QAGS is best correlated with semantic frame errors.

Entailment metrics Both entailment metrics correlate well with semantic frame and content verifiability errors. DAE has the highest correlation of all metrics with discourse errors suggesting that entailment at the dependency level can help model discourse errors (CorefE and LinkE). FactCC is nearly uncorrelated in this category, indicating that artificially generated factual errors need to go beyond simple pronoun swaps to train models to capture discourse errors. FactCC is best at capturing semantic frame among all metrics evaluated.

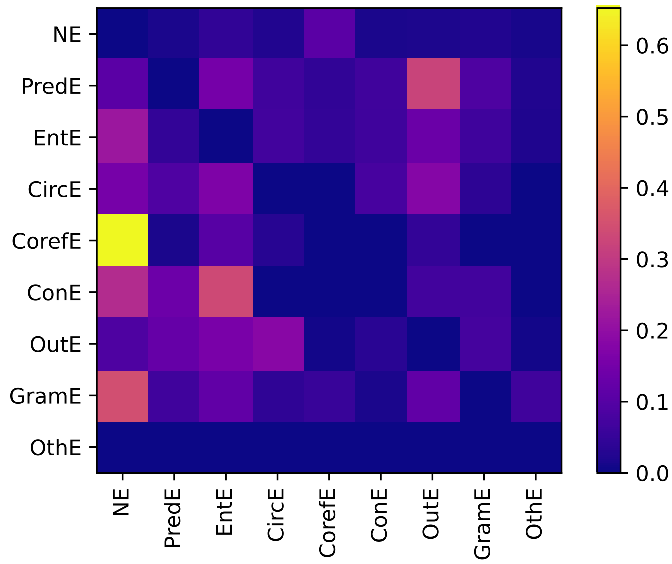

Figure 2.6: Confusion matrix of different types of errors. Entry at row i, column j corresponds to the frequency of annotations that have Fi as the majority class and for which disagreeing annotator selected Fj.

2.5.5 Mutual Exclusiveness of Typology

To understand if our annotations are mutually exclusive, we study cases where two annotators agree on the error category (majority class) and one disagrees (minority class). In Figure 2.6, we report the confusion between majority and minority classes. For each category as majority, we report the distribution of other categories as minority.

We observe that all categories with the exception of **OutE** are frequently confused with **NE** which stands for no factual error. This primarily due to the noise in the annotations collected by crowd workers. However, for category **CorefE** (coreference errors) the confusion is significantly higher with 69.7%. We have noticed the same trend in practice tutorials: crowd annotators easily overlook situations where the correct pronoun is used (in terms of number and gender) but no antecedent appears in the summary. Intuitively after reading the article, unless paying particular attention, it is easy to subconsciously associate referring expressions with entities in the article without noticing their absence in the summary. The error persists despite stating the scenario explicitly in the instructions. This indicates an issue with annotators rather than annotation scheme.

The other trend that we observe is that categories **PredE** (wrong relation) and **CircE** (wrong modifier) are often confused with **OutE** (outside information). In our definition of **OutE**, outside information corresponds to the presence of entities not mentioned in the article or relations that cannot be verified based on the article. The confusion with **PredE** indicates that annotators can have different judgements on whether a relation is verifiable based on the article. Similarly, but to a lesser degree, wrong circumstantial information might be considered unverifiable given the article.

Finally, there were relatively few discourse context errors **LinkE**, so the analysis is less statistically significant. Discourse context errors correspond to using a wrong connectors between different facts, for example different logical links. These were confused with **PredE** and **EntE** (wrong relation). The distinction between the two errors lies in the confusion between what an entity and a fact are, since **PredE** occurs at the frame level while **LinkE** at the discourse level. Note, that there was no confusion in the other direction (**PredE** being confused with **LinkE**).

2.6 Related Work

Kryscinski et al. (2019) and Fabbri et al. (2020) find that standard n-gram based metrics have low correlation with human judgements of factuality. Motivated by this, several automated metrics falling in two paradigms were proposed to improve the evaluation of factuality.

Entailment Classification: Goodrich et al. (2019); Kryscinski et al. (2020); Maynez et al. (2020); Goyal and Durrett (2020) model factuality as entailment classification breaking down the summary into smaller units, such as sentences, which are verified against the original article. However, modeling factuality as a classification task requires supervision on factual and hallucinated data. FactCC (Kryscinski et al., 2020) is trained on the CNN/DM dataset augmented with four types of artificial mistakes as supervision.

Question Generation and Answering (QGA): FEQA (Durmus et al., 2020) and QAGS (Wang et al., 2020a) are two metrics which reduce factuality evaluation to question generation and answering. These methods use a question generation model to obtain questions from the output summary and a question answering model to answer them, separately using the article and the output summary.

Prior Efforts on Factuality Annotations of Summaries: Fabbri et al. (2020) and Maynez et al. (2020) have collected annotations on the CNN/DM and XSum dataset respectively. In this work we cover both datasets to ensure greater data diversity. Other efforts (Kryscinski et al., 2020; Wang et al., 2020a; Durmus et al., 2020) were smaller in scale Durmus et al. (2020) and Kryscinski et al. (2020) annotated 200 and 503 sentences while Wang et al. (2020a) annotated 470 summaries (we collect judgements on 2250 summaries). Crucially, all previous efforts portray factuality as a binary label without variations in degree or type of factual errors.

2.7 Conclusions and Future Work

In this work we provide a linguistically grounded typology of factual errors which we use to collect FRANK, a dataset of human annotations of 2250 summaries covering both CNN/DM and XSum datasets. We use FRANK to assess the factuality of summarization systems and benchmark recently proposed factuality metrics highlighting the types of errors they can capture. With the FRANK benchmark we have started moving away from a summary-level binary understanding of factuality.

Our work leveraged human annotators for extensive annotations which is both expensive and time-consuming, and hence not scalable. An interesting avenue to explore in future is to leverage effective human-model collaboration strategies to conduct more broad and scalable evaluations (Laban et al., 2023). While such collaboration strategies reduce cost and can help effectively scale, incorporating them for evaluation needs to be done carefully to ensure fair and unbiased results. Additionally, this work uses the fine-grained typology to evaluate summarization models and metrics. But such work can able so enable the design and development of automatic metrics for fine-grained error detection, which can enable such fine-grained analysis to be conducted for new summarization models.

Ethical Considerations

This work involved using human annotators for annotating data. We have collected crowd annotations using the Amazon Mechanical Turk platform. Workers were paid 50% more than the average American minimum wage and offered additional bonuses as an incentive to maintain high quality work. Further, all data used for annotation were public news documents and summaries. No information about the workers will be released and worker IDs will be anonymized.

While this work presents an evaluation protocol for fine-grained analysis of summarization models and metrics, it only presents the analysis for specific models and metrics which were state-of-the-art at the time of the publication. Care should be taken to not generalize the results to artifacts which were not evaluated here and rather the evaluation should be redone with the specified protocol to understand limitations of any model/metric of interest. Finally, while evaluation protocols and analysis are guidelines to understand models and their limitations, they are not gold-standard certifications of a model. Thorough evaluations and testing of models need to done across diverse parameters before using them in different applications.

Chapter 3

Evaluation of Factual Consistency in Open-Ended Generation with Semantic Frame Structure

Our findings from Chapter 1 indicate that factual errors in model generated text persist even with additional model pretraining. Particularly entity and hallucination errors are significantly high in cases where the model is expected to generate abstractive text. This issue is further exacerbated in the case of open-ended generation, where models generate text based on pretrained knowledge without being grounded in any specific document. As LMs are increasingly being used with information seeking intent and being incorporated into information access systems, developing evaluation measures to study the factual accuracy of such open-ended language generation systems is important. Conducting such an evaluation is challenging. As information is vast, evaluation measures need to be general and comprehensive to be truly useful. Currently, LMs are evaluated for their memorization capabilities using question answering and slot-filling benchmarks (Mallen et al., 2023; Petroni et al., 2019). Findings from such evaluations are limited and do not transfer to open-ended generation, where information is produced in long passages, motivating the need fine-grained measures to target and evaluate each unit of information.

This chapter addresses the problem of hallucinations in open-ended generation and extends the previous typology to describe more complex factual errors in this setting. First, it describes a taxonomy of hallucination categories drawn from semantic frame theory and pragmatics to support evaluation of hallucinations. Based on the typology, the chapter presents a benchmark of human annotated factuality judgements of model generated text and proposes a model for fine-grained hallucination detection and editing. Research described in this chapter was conducted in collaboration with Abhika Mishra, Akari Asai, Yizhong Wang, Graham Neubig, Yulia Tsvetkov and Hannaneh Hajishirzi and is under review at ACL 2024 (Mishra et al., 2024).

3.1 Introduction

Large language models (LMs; Brown et al. 2020) can generate highly fluent and plausible text. However, these models are prone to produce *hallucinations*—factually incorrect generations—which often impedes their deployment in real-world applications (Mallen et al., 2023). Prior work on hallucinations in natural language generation often assumes the presence of a specific reference text and focuses on studying faithfulness to the references (Ji et al., 2023). On the contrary, escalating apprehensions have been articulated with respect to LM generations that are not grounded in any specific source text, but rather in world knowledge and facts (Zhang et al., 2023; Huang et al., 2022).

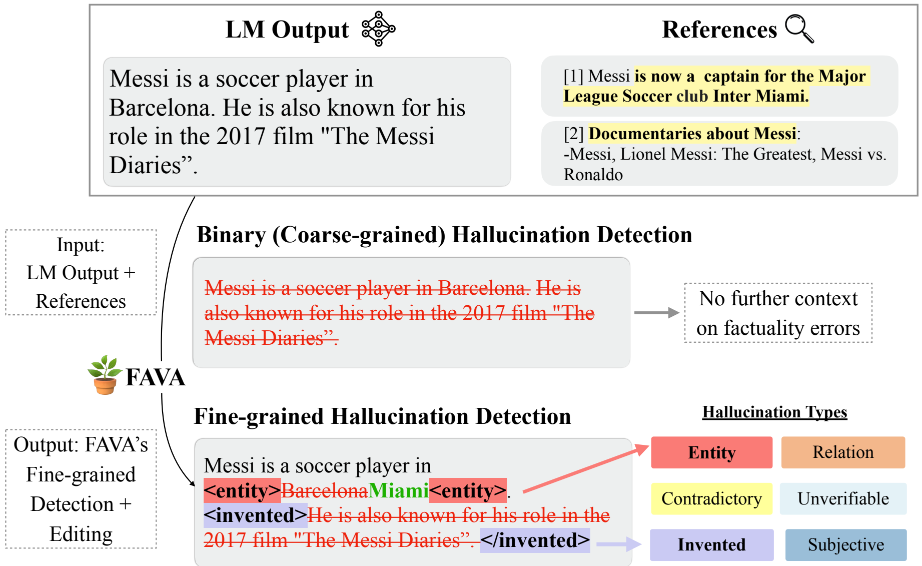

Figure 3.1: Overview of taxonomy for LM hallucinations and our model, FAVA, a fine-grained detection system.

Several recent work studies automatic hallucination detection (Min et al., 2023) or editing outputs (Gao et al., 2022) to address such LM hallucinations. Similar to work in summarization, these systems typically categorize hallucinations into simplistic binary distinctions like *factual* or *not factual* (Figure 3.1 top). We argue that hallucinations in open-ended generation also manifest in diverse forms, each requiring varying degrees of careful assessments to verify factuality. Entity-level contradictions are usually evident and can be easily rectified with a single reference. Conversely, errors involving fabricated entities (e.g., The Messi Diaries in Figure 3.1) demand thorough verification across multiple sources. This underscores the need for a more fine-grained approach to detect hallucination for model development and human verification.

In this work, we propose **automatic fine-grained hallucination detection** (Figure 3.1 bottom) for open-ended generation, a new task requiring a system to provide precise identification of hallucination sequences, discern different types of hallucinations based on a taxonomy, and suggest refinements. We focus on hallucinations in information-seeking scenarios, where grounding to world knowledge matters. Our taxonomy (Figure 3.2) hierarchically classifies hallucinations in LM generations into six categories, based on careful pilot studies with NLP experts.

We construct a new fine-grained hallucination benchmark, FAVABENCH, by carefully annotating approximately 1,000 responses of three widely used LMs (Llama2-Chat 7B, 70B and ChatGPT[1]) to diverse knowledge-intensive queries. Each response is annotated at the span level to identify hallucinations, including erroneous subspace, types, and potential refinements. Our analysis reveals all models include at least one hallucination in the majority of their responses (e.g., 70.2% in Llama2 7B and 59.8% in ChatGPT). In addition to the widely-studied entity-level errors, other error types like unverifiable sentences make up over 60% of LM-generated hallucinations, highlighting the urgent need to study and detect such diverse types.

We introduce FAVA, a new retrieval-augmented LM that can identify and mark hallucinations at the span level using a unified syntax (Figure 3.1, left bottom). Due to the annotation costs, we design an LM-based synthetic data generation process, and train FAVA on the 35k resulting instances. We compare FAVA with

[1]We use `gpt-3.5-turbo-0301` throughout this work.

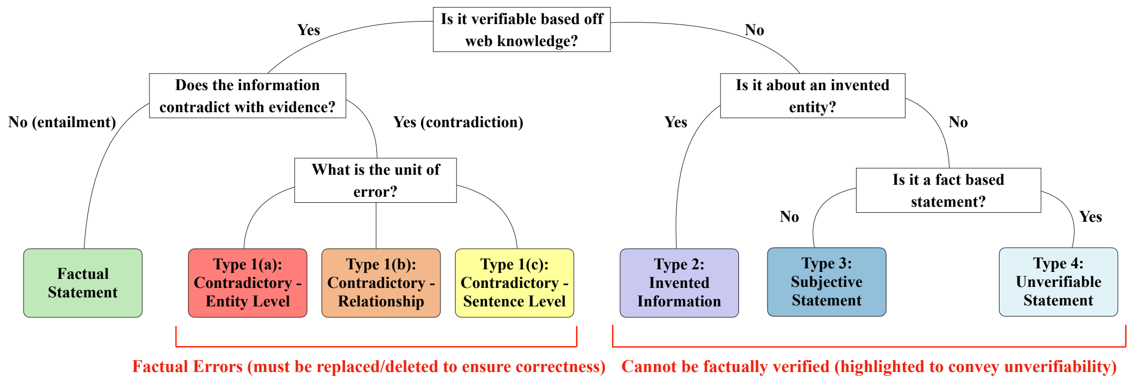

Figure 3.2: An overview of our fine-grained hallucination taxonomy. We identify 6 fine-grained types representing diverse hallucinations in LM-generated text.

state-of-the-art LMs on fine-grained hallucination detection in LM outputs. FAVA significantly outperforms ChatGPT with and without external knowledge by 23.7% on fine-grained hallucination detection.[2]

3.2 Related Work

Hallucinations in natural language generation. Taxonomies of hallucinations in certain downstream tasks such as summarization (Pagnoni et al., 2021), text simplifications (Devaraj et al., 2022), or knowledge-grounded dialogues (Dziri et al., 2022) have been studied. Such prior taxonomies often assume the existence of specific source text and focus on faithfulness to the source (Ji et al., 2023), which is qualitatively different from LMs' factual hallucinations grounded in world knowledge. In addition, prior taxonomies include many task-specific categories that cannot be directly adapted to describe hallucinations in LM generations.

Detecting and editing hallucinations in LMs. Several recent works study methods to predict factuality by giving binary labels of a statement being factual or not (Manakul et al., 2023; Min et al., 2023), or focusing on entity-level factual errors editing (Gao et al., 2022; Chen et al., 2023a). Recent surveys have attempted to categorize hallucinations in LM generations, but such categorizations often stretch the definition of hallucinations to broad error types in LM-generated text, which cannot be traced to specific spans in text (Huang et al., 2023; Zhang et al., 2023; Rawte et al., 2023). In this work, we propose a fine-grained taxonomy for fact-conflicting hallucinations in long-form text generation to improve factual consistency. While prior work often develops factuality verification systems on top of proprietary LMs (Chern et al., 2023), we introduce a carefully designed training pipeline for a smaller yet more competitive fine-grained detection system. Unlike prior qualitative analysis on hallucinations in LM responses on certain knowledge-intensive queries (e.g., open-domain QA, biography generation; Liu et al., 2023b; Min et al., 2023), we annotate responses to diverse queries in multiple domains and reveal hallucination distributions vary across domains.

Fact verification. While research on hallucination detection focuses on identifying errors in model-generated text, a related area of research focuses on identifying factual inaccuracies in human written claims (Bekoulis et al., 2021). Thorne et al. (2018) introduce a large scale dataset based on Wikipedia documents for the training

[2]Our code, data, and demo are available at https://fine-grained-hallucination.github.io/

Type	Example	ChatGPT	Llama2-7b	Llama2-70b
Entity	Lionel Andrés Messi was born on June ~~12~~ **24**, 1987.	42.7%	48.6%	55.7%
Relation	Lionel Messi ~~**acquired**~~ **was acquired by** Paris Saint-Germain.	4.7%	3.4%	2.8%
Contradictory	**Messi has yet to be captain for the Argentina football team.**	18.9%	15.7%	19.9%
Invented	**Messi is known for his famous airplane kick technique.**	14.2%	17.4%	9.5%
Subjective	Lionel Messi is **the best soccer player in the world**.	9.9%	8.6%	5.8%
Unverifiable	**In his free time, Messi enjoys singing songs for his family.**	9.6%	6.3%	6.3%

Table 3.1: Distribution of different errors across ChatGPT, Llama2-Chat-7B and Llama2-Chat-70B outputs and examples of hallucinations for each type from our human-annotated data

and evaluation of fact verification systems. Other datasets in information-dense domains like fact-checking against scientific articles (Wadden et al., 2020) and news documents (Wang, 2017) have been proposed. Several systems have been developed for fact-checking in these settings (Thorne and Vlachos, 2018; Schuster et al., 2021; Nakov et al., 2021), but have not been tested for LM-generated text. While our work focuses on hallucination detection in LM generations, FAVA can be adapted to fact-check human written claims as well.

3.3 Semantic Frame based Hallucination Taxonomy

3.3.1 Focus and Definitions

In this work, we focus on open-ended text generation given information-seeking queries which often require factual knowledge. We define hallucinations as *factually incorrect or unverifiable statements given external world knowledge*. We operationalize world knowledge as documents from the web that are most relevant to the given statement/query according to a search algorithm.[3]

3.3.2 Hallucination Taxonomy

Based on our definition of hallucinations, we build a hierarchical taxonomy grounding in language structure to categorize them into fine-grained error types. To develop a hallucination taxonomy for open-ended LM generations, we draw inspiration from error categories in prior task-specific taxonomies from Chapter 2 and (Devaraj et al., 2022) and introduce new categories to describe more complex errors surfacing in LM generations. We conducted a pilot annotation with 9 NLP experts to discuss and refine our taxonomy to ensure good coverage across diverse error types.

Figure 3.2 shows our taxonomies to classify LM hallucinations and Table 3.1 shows examples of each type of hallucination. We group hallucinations into two major categories: statements that contradict world knowledge at a semantic frame level (Type **1**) and unverifiable statements based on knowledge and context (pragmatics) (Types **2**, **3**, and **4**).

Semantic Frame Contradictions: These categories of errors are an extension of Semantic Frame errors from Chapter 2 and account for contradictions at a frame level. These are cases where a component of a particular semantic frame (Subjects, Relations, Arguments, etc) directly contradict the evidence that exists in world knowledge. They can be fine-grained (a core or non-core component being wrong) or coarse (the entire frame is contradictory).

[3]Errors in common sense, numerical, or logical reasoning are out of the scope of this work.

(1a) **Entity**: Contradictory entity errors are a sub-category within contradictory statement errors where an entity in a statement is incorrect and changing that single entity can make the entire sentence factually correct.
(1b) **Relation**: Contradictory relation errors are another sub-category within contradictory statements where a semantic relationship described in a statement is incorrect. While (1a) often involves errors in nouns, (1b) involves verbs, prepositions, or adjectives, and can be fixed by correcting the incorrect relation.
(1c) **Contradictory**: Contradictory statement errors refer to statements that entirely contradict relevant evidence from the web. These are cases where the full sentence is refuted by the information provided in a given reference.

Coarse Pragmatic Contradictions: These categories of errors are introduced to specifically address more complex errors that manifest in open-ended generation. These errors are broad and may span different levels of structure (entities, semantic frames, sentences, discourse units) and are a consequence of generated text being inaccurate based on multiple pragmatic elements like context, external knowledge or subject background.
(2) **Invented**: Invented errors refer to statements with concepts that do not exist based on context and world knowledge. These are cases when the LM generates a non-existent or entirely fabricated entity that doesn't occur in any relevant evidence. This does not include fictional characters in creative work such as books or movies.
(3) **Subjective**: Subjective errors refer to an expression or proposition that lacks universal validity and is often influenced by personal beliefs, feelings, opinions, or biases and hence cannot be judged as factually correct. Specifically, this represents cases where the LM generates a statement without any factual proposition grounded in relevant evidence or conversation structure.
(4) **Unverifiable**: Unverifiable errors refer to statements that contain factual propositions but cannot be grounded in world evidence. These are cases where the LM generates statements with facts, but none of the retrieved evidence from the web or background knowledge can directly support or contradict the fact (e.g., personal or private matters).

While Entity or Relation are often phrase-level and can be fixed by minimal editing erroneous phrases, other error types can be an entire sentence or part of a sentence and should be removed from a response to make it factual.

3.3.3 Tasks and Metrics

We introduce a new task of identifying fine-grained factual errors in LM outputs. Given an input query x and a corresponding LM output y, our tasks require systems to identify all of the factual errors. Each error e consists of (e^{text}, e^{type}), indicating the factually incorrect text spans and their error types among our taxonomies, respectively. We evaluate systems' abilities concerning identifying fine-grained error types in the model-generated text e^{type}.

Task: Fine-grained hallucination detection. In the first fine-grained error detection task, the system is expected to identify fine-grained errors in an LM output. Due to the subjectivity of span-level annotations, for automatic evaluation, we evaluate systems' abilities to detect whether an error type t exists in a sentence $s_i \in y$. Given an output y consisting of L sentences, we assume the availability of gold error type annotations $e_i^{*t} \in \{\texttt{TRUE}, \texttt{FALSE}\}$, which is a binary label of an error type t existing in the ith sentence (TRUE) or not

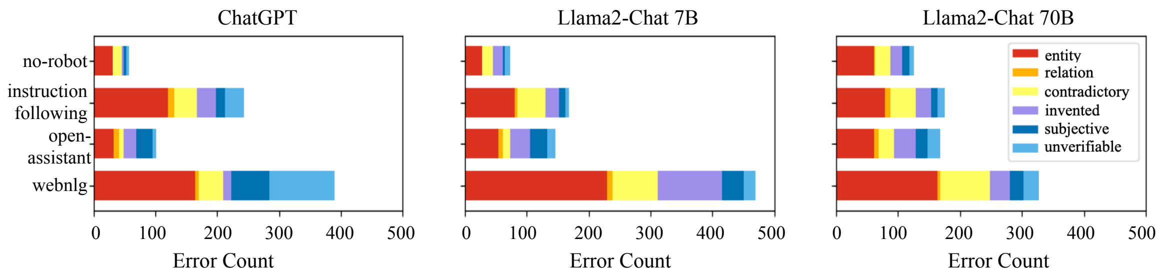

Figure 3.3: Distribution of hallucination types in ChatGPT, Llama2-Chat-7B and Llama2-Chat-70B outputs across four datasets of diverse information-seeking queries based on our human annotated benchmark.

(FALSE). For each type, a system predicts e_i^t and we evaluate precision and recall as follows:

$$\text{Prec}^t = \frac{\sum_{i \in L} \mathbb{1}[e_i^t == e_i^{*t}]}{\sum_{i \in L} \mathbb{1}[e_i^t == \texttt{TRUE}]} \tag{3.1}$$

$$\text{Recall}^t = \frac{\sum_{i \in L} \mathbb{1}[e_i^t == e_i^{*t}]}{\sum_{i \in L} \mathbb{1}[e_i^{*t} == \texttt{TRUE}]} \tag{3.2}$$

Precision indicates how many of the model's predictions of an error type t existing in the ith sentence is correct, while recall indicates how many of the error sentence TRUE is identified by the model. For final score, we compute the F1 scores averaged over six error types as follows:

$$\frac{1}{|\mathcal{E}|} \sum_{t \in \mathcal{E}} \frac{2\text{Prec}^t \text{Recall}^t}{\text{Prec}^t + \text{Recall}^t} \tag{3.3}$$

Fine-grained error detection can be simplified into a binary classification task, in which a system predicts if a sentence s_i includes any factual errors or not. This is similar to FActScore (Min et al., 2023), where we predict and evaluate factuality at the sentence level without extracting atomic facts.

3.4 Benchmark

We create FAVABENCH to facilitate model development and evaluation in fine-grained hallucination detection and understand how prevalent those different types of hallucinations are. FAVABENCH consists of multi-way fine-grained annotations on three LM responses to queries in multiple domains.

Source prompts. The source prompts include a collection of 200 information-seeking queries, spanning four different data sources. See examples in Table 3.2.

- Knowledge-intensive queries sampled from the Open Assistant dataset Köpf et al. (2023), by prompting GPT-4 to judge whether each query from the dataset requires world knowledge.
- A sample of Open QA prompts from the No Robots dataset Rajani et al. (2023).
- Instruction-following prompts requiring more reasoning and knowledge curated by the authors.
- Synthetically-created prompts 50 prompts that require more fine-grained knowledge, by converting data-to-text WebNLG dataset Gardent et al. (2017) using a template.

Dataset	Example
WebNLG	Explain A.C. Cesena, including information about ground, league.
Instruction-following	Explain the differences between New York cheesecakes and Basque cheesecakes in detail.
Open-Assistant	Can you tell me how tall the burj khalifa is?
No Robots	When was Samsung founded?

Table 3.2: Examples of source prompts.

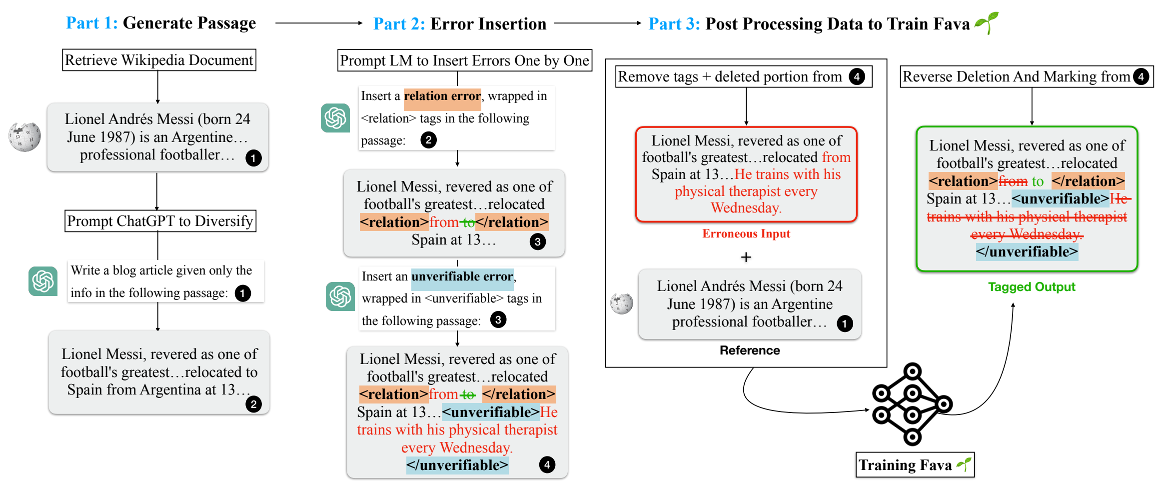

Figure 3.4: Overview of high-quality synthetic data generation process in FAVA. FAVA leverages powerful instruction-tuned models to carefully insert errors into factually accurate statements and produces diverse error types based on our proposed taxonomy.

Annotation details and quality. We obtain responses to the collected source prompts using ChatGPT (`gpt-3.5-turbo-0301`; Ouyang et al. 2022), Llama2-Chat 7B and Llama2-Chat 70B (Touvron et al., 2023) in a zero-shot manner and collect 600 responses to our diverse information-seeking prompts. We recruited 20 students (ten undergraduate and ten NLP graduate students) to annotate the factual accuracy of the responses based on our proposed taxonomy. Each instance is annotated by two annotators who completed 45-minute in-person and virtual training sessions.[4] Our annotation interface and details are in Appendix 1.2.1. To validate our annotation quality, we calculated inter-annotator agreement using Cohen kappa scores and found that our annotators had high agreement in detection across passages, with 75.1% agreement in detection at the sentence level and 60.3% agreement in exact error type detection at the sentence level across passages. Appendix Table 1.2 shows examples of our annotation results.

Analysis on annotated data. Figure 3.3 presents a detailed breakdown of distributions across fine-grained categories in the three domains. 59.8%, 70.2% and 64.9% of the responses of ChatGPT, Llama2-Chat 7B and Llama2-Chat 70B include at least one hallucination, respectively. Entity is the most widely recognized error type, making up 48.3% of the detected errors. Yet, there are diverse types of errors prevalent like invented statements or contradictory statements, which make up 14.1% and 18.1% of the detected errors, respectively. The error distributions vary across different source prompts and LMs. Invented are more common in Llama2-Chat models. Fewer errors in No Robots subsets than other subsets may be because their seed prompts are less knowledge-intensive, or ask about popular factual knowledge (e.g., *How long was the Revolutionary War?*), which is often memorized by LMs (Mallen et al., 2023).

[4]Annotating each response takes approximately 10 minutes, and we pay USD 3.5 for each annotation.

3.5 Model: FAVA

To automatically detect such diverse fine-grained hallucinations, we introduce a new retrieval-augmented LM, FAVA (**FA**ct **V**ericaton with **A**ugmentation). FAVA is trained on high-quality synthetic training data, and at inference, it identifies and fixes fine-grained hallucinations, incorporating retrieved knowledge.

3.5.1 Overview

FAVA consists of two components: a retriever $\mathcal{M}_{ret}$ and an editing LM $\mathcal{M}_{edit}$. $\mathcal{M}_{ret}$ takes the original output LM y and optionally input prompt x (if applicable), and retrieves top relevant documents: $\mathbf{C} = \mathcal{M}_{ret}(x, y)$. Subsequently, the editing model detects and if possible edits factual errors in y given the retrieved context: $\hat{y} = \mathcal{M}_{edit}(x, y, \mathbf{C})$. $\hat{y}$ is an augmented output y interleaved by the error edits with hallucination types as shown in Figure 3.1.

While $\mathcal{M}_{edit}$ can be any LM, in our preliminary experiments, we find that making a state-of-the-art proprietary LM such as ChatGPT to perform fine-grained editing via prompting only is challenging (ref Table 3.5). Reliance on black-box proprietary API models also hurt reproducible. Therefore, we generate high-quality synthetic training data with minimal human efforts (§3.5.2) and fine-tune a powerful expert LM consisting of 7 billion parameters (§3.5.3).

3.5.2 Training Data Creation

To train our $\mathcal{M}_{edit}$ model, we require a large number of erroneous LM outputs y paired with edited versions y^* with fine-grained error types and edits. Inspired by prior work that leverages LMs to generate synthetic training data (Balachandran et al., 2022; Wang et al., 2023c; Asai et al., 2023), we introduce a new data creation method grounded on our fine-grained hallucination taxonomies. We prompt GPT-4 (Achiam et al., 2023) and ChatGPT to diversify and noise a Wikipedia passage by inserting different types of hallucinations.

In particular, our data creation pipeline consists of three steps: **seed passage generations**, **error insertions**, and **postprocessing**. The first two processes create an input sequence with perturbed diverse error annotations, and postprocessing remaps those special tokens to generate the erroneous output y and output with fine-grained detection results y^*. We use the original Wikipedia passage as well as passages retrieved by an off-the-shelf system as $\mathbf{C}$. Figure 3.4 shows the overview of the pipeline.

Seed passage generation. We randomly sample a Wikipedia article c and use it as our gold reference passage. To produce model-generated seed passages, we diversify the sampled article by paraphrasing and transferring the text to another genre via an LM. Specifically, we randomly sample one of the pre-specified genre types (e.g. blog article, question answer pair, tweet, etc.) for each passage and prompt ChatGPT to rephrase the text t in the style of the new genre, specifying to include information only found in the Wikipedia article c to maintain factuality. To further diversity in the dataset, we also add examples of question answering (QA) data. In particular, we prompt ChatGPT with two demonstrations from the NaturalQuestions (Kwiatkowski et al., 2019) dataset to generate 5,000 queries and correct answers using the information in the sampled Wikipedia articles. See the full list of the genres, instructions, and final ChatGPT-generated text in Appendix Table 1.3.

Error insertion. In our pilot studies, we found that asking ChatGPT to insert multiple error types at the same time easily makes the model misunderstand or get confused with different error types (e.g., swapping or incorrectly identifying an error type). We also found while ChatGPT is capable of generating more simple types of errors such as Entity, it struggles with generating plausible and difficult perturbations for more nuanced types. Therefore, we use ChatGPT and GPT-4 interchangeably for six different types.[5] In particular, given an instruction and few-shot demonstrations of an error type e^{type}, GPT-4 or ChatGPT inserts new errors while retaining previously inserted errors and original text. The model was specifically instructed to mark phrases or sentences for deletion along with their error type and insert phrases and sentences along with insertion tags, allowing us to verify the error types and control their distribution. After this process, we have an erroneous text y, which inserts multiple errors into the original factual text t.

Post processing. We then post-process data, by swapping and removing the error tags and edits to form a clean erroneous text y as the input and use the original correct passage y^* as the output with edits, for training $\mathcal{M}_{edit}$. At this stage, we also filter out the examples violating our editing rules (e.g., inserting or removing errors without marking them up with tags and error types). As a result, we have a training instance (c, y, y^*), consisting of the gold context (original Wikipedia paragraph) c, erroneous LM output y, and output with error tags and correct editing y^*. We also retrieve additional four relevant paragraphs from Wikipedia using Contriever (Izacard et al., 2022), and randomly mix the order of the references, forming the final references $\mathbf{C}$. As this process does not involve human annotations, we can easily scale up training instances under an API cost budget.

Statistics of data. We generated a total of 35,074 training instances, 30,0074 of which are based on Wikipedia passages, and 5,000 are based on QA pairs described above. All of the error types are almost equally distributed (roughly 15% each). We present the error distributions in Table 3.3. We conduct a detailed human evaluation in §6.5. On average, 3.1 errors are inserted for the Wikipedia passage-based subsets, and 1.4 errors are inserted for the QA pairs.

	Percentage
Entity	21.2%
Relation	19.9%
Contradictory	15.3%
Invented	14.6%
Subjective	14.1%
Unverifiable	14.9%

Table 3.3: Statistics of generated training data

3.5.3 Training and Inference

We train a smaller LM on the generated large-scale training data. In particular, given a training instance $(\mathbf{C}, y, y^*)$, our model $\mathcal{M}_{edit}$ takes $(\mathbf{C}, y)$ as input and learns to predict the edited outputs with tags to represent error type y^* using standard language modeling objective. Our base model is Llama2-Chat 7b trained using 4xA40 GPUs. Our training code is based off Open-Instruct (Wang et al., 2023b)[6]. Table 3.4 shows the training hyperparameters.

[5] We use ChatGPT for the Type **1** and Type **3** while using GPT-4 for other types and insert factual errors one by one into our diversified text.

[6] https://github.com/allenai/open-instruct/blob/main/scripts/finetune_with_accelerate.sh

Precision	Epochs	Weight Decay	Warmup Ratio	Learning Rate	Max. Seq. Length	Batch Size
BFloat16	2	0	0.03	2e-5	2048	128

Table 3.4: Training hyperparameters.

	Generator: ChatGPT								Generator: Llama2-Chat 70B							
Editor	ent	rel	con	inv	subj	unv	OA	Bi	ent	rel	con	inv	subj	unv	OA	Bi
ChatGPT	19.5	**28.6**	40.0	11.8	7.7	0.0	18.8	50.1	24.7	15.6	26.7	11.0	17.6	12.8	24.1	68.4
Rt+ChatGPT	28.1	19.2	25.5	5.4	37.7	15.5	24.4	64.8	33.7	24.2	24.0	22.2	17.8	4.7	27.8	72.8
GPT4	38.6	16.6	17.9	22.2	50.0	17.2	34.2	60.8	55.5	**60.0**	21.2	15.4	2.0	25.0	42.5	74.2
FAVA (ours)	**54.5**	25.0	**66.7**	**16.7**	**70.5**	**35.3**	**48.1**	**79.6**	**57.3**	34.5	**27.7**	**52.2**	**31.25**	**43.4**	**47.2**	**80.3**

Table 3.5: Fine-grained detection F1. `OA` and `Bi` indicates overall and binary predictions.

At inference time, we retrieve the top five documents from Wikipedia,[7] using Contriever-MSMARCO (Izacard et al., 2022), and insert them together with an LM output that may include factual errors. The model identifies factual errors, marks phrases or sentences for deletion, and suggests edits for improving the factuality of the text.

3.6 Experiments

We evaluate FAVA and state-of-the-art LMs with and without retrieval on the fine-grained hallucination detection (§3.6.1). We further conduct human evaluations on our best models (§3.6.2).

3.6.1 Experiments for Hallucination Detection

Evaluation data. We use our new benchmark, consisting of 364 annotated passages,[8] as our test data. We measure the models' detection performance based on sentence-level per-category classification task as formulated in §3.3.3.

Baselines. For fine-grained hallucination detection tasks, we test three baselines. **ChatGPT** prompts ChatGPT (`gpt-3.5-turbo-0301`) with a carefully designed prompt describing all six categories with two demonstrations. **Rt-ChatGPT** uses the same prompt and demonstrations but also includes the top five retrieved documents by Contriever at test time to augment the original prompt.[9]

3.6.2 Human Evaluations

Our automatic evaluations may not fully capture the models' abilities to detect annotated data and editing errors due to the potential subjectivity of annotations and may be affected by the performance of factuality evaluation metrics. We evaluate randomly sampled 50 outputs from FAVA as well as the baseline with the highest automatic evaluation score, namely Rt-ChatGPT. We ask human annotators to verify how many of the detection are indeed correct based on the provided retrieved documents. This is similar to our automatic precision evaluation, but instead of coarsely evaluating detection performance at the sentence level, we evaluate

[7]We use English Wikipedia data from January 2023 and generate embeddings using the Contriever encoder.

[8]In total we collected 471 annotated passages including our experimental batches, and we excluded earlier batches for evaluations.

[9]While we also tested strong white box LMs including Llama2-Chat 13B, we found their predictions often show confusion among different categories or struggle to output predictions in the expected format.

Model	avg. $\|\mathbf{E}\|$	Detect (%)
Rt+ChatGPT	1.9	23.9
FAVA (ours)	2.4	**55.7**

Table 3.6: Human evaluation results. We show the average number of detected errors and the correctness (%) of the fine-grained types.

the model performance at the individual detection level. Due to the cost of evaluating the factuality of all verification-worthy statements (Min et al., 2023), we do not evaluate recall.

3.7 Results and Analysis

3.7.1 FAVA Evaluation Results

Detection results. Table 3.5 shows the fine-grained detection accuracy of FAVA and baselines. We provide the full precision and recall scores in Appendix Tables 1.5 and 1.6, respectively. We also provide binary prediction performance.

Table 3.5 shows that while ChatGPT-based baselines show relatively good binary predictions of existence of factual errors, their fine-grained error type detection performance is significantly lower. FAVA shows its strong capabilities of conducting fine-grained error detection, resulting in significant performance improvements from ChatGPT or Ret-ChatGPT on both fine-grained error detection and binary error detection. Specifically, FAVA shows high accuracy on error types such as Contradictory, Subjective, Entity. On the other hand, its performance on Invented and Unverifiable are still limited. Those two error types often require intensive search over many web documents beyond top few passages, while FAVA by default only considers top five documents. Our inter-annotator agreements also show that there are more disagreements in those two categories, which might also deflate the performance.

Human evaluation results. Our human evaluation results are shown in Table 3.6. As in automatic evaluations, our method shows significantly better performance than Rt+ChatGPT on both editing and detection and recognizes more errors than retrieval-augmented ChatGPT. These results further demonstrate the strong capabilities of FAVA detecting factual errors in LM outputs.

3.7.2 Ablation and Analysis

Effects of scale. We evaluate our model trained on varying numbers of model-generated training data. In particular, we train three FAVA variants trained on 10k, 20k, and 30k training instances, and evaluate their performance on fine-grained error detection tasks. Figure 3.5 shows the results of each variant and demonstrates that FAVA variants with a larger number of training instances perform significantly better at detecting fine-grained errors.

Human evaluation of generated data. We conduct human evaluations on 50 generated data to assess the automatic data creation quality. We evaluate a generated instance from two aspects: (1) *validity*—whether the model edits do not violate our annotation scheme; (2) *quality*—the inserted errors are feasible and realistic and notably different from provided few-shot samples. We ask human annotators to score each category for each passage either 0, 1, or 2 (higher is better; see the detailed criteria in our Appendix).

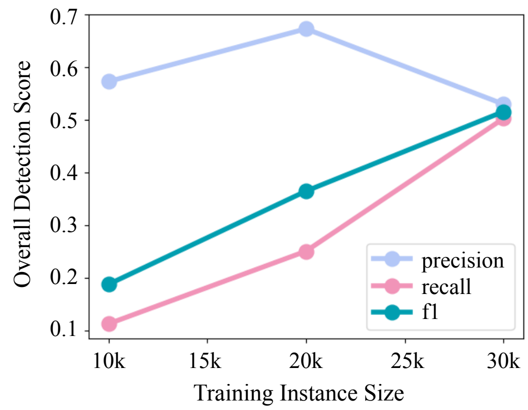

Figure 3.5: Detection scores across different training instance values.

Our analysis revealed that the data generated by our system with iterative insertion obtained an average score of 1.66 for validity assessment and 1.36 for quality assessment. Meanwhile, data generated by one-shot prompting scored an average of 1.1 for validity and 0.9 for quality. Our human evaluation reveals that our LM-generated edits are indeed of higher quality than the one-shot data creation, and provide realistic factual errors.

3.8 Conclusions and Future Work

This work introduces a new task of automatic hallucination detection, built upon our newly introduced taxonomy, which hierarchically classifies hallucinations in LMs into six categories. We collect the first human-annotated fine-grained hallucination benchmark, consisting of span-level hallucination types, hallucination text, and edits for two LM outputs across multiple domains. As the first step, we introduce a new retrieval-augmented LM trained on synthetically generated fine-grained error detection data following our taxonomies. Empirical results show that FAVA significantly outperforms strong baselines by a large margin on fine-grained detection tasks, while still large room for improvements for automated fine-grained error detection.

This work focuses on evaluating LLM generations for open-ended generation settings and targetted towards information seeking contexts. While we have presented the strengths and weaknesses of models in these settings, an open-question of interest for the community is how the factual accuracy in different styles varies. In future work, we plan to expand the evaluations to include more diverse settings like retrieval-augmented generation (Lewis et al., 2020), summarization (Laban et al., 2023; Chang et al., 2023) or text editing (Dwivedi-Yu et al., 2022). Additionally, exploring the application of such a taxonomy for specialized domains like finance, legal or healthcare would also be interesting, potentially requiring adapting the typology to the domain of interest.

Ethical Considerations

This chapter also involved using human annotators for annotating data. Here, we have collected annotations using voluntary undergraduate students and graduate NLP students. Annotators were fairly compensated above average American minimum wage. No personal information about the annotators were collected and the data will anonymized in the public release.

While care was taken to develop a general and robust evaluation framework and detection model, it is not perfect and should not be considered as a gold-standard factuality evaluation for broad deployment. Instead, based on our taxonomy and recommendations, tailored evaluations for the application and domain of interest should be conducted before any model is released or deployed for user consumption. Further, our work has only been extensively tested for general information-seeking contexts and therefore our findings might not apply to other applications like finance, legal or healthcare. When using our framework for such domain specific applications, we recommend adapting and testing the taxonomy and model before deployment.

Part II

Language Structure for Transparent Model Design

Chapter 4

Interpretable Text Summarization by Leveraging Document Structure

In previous chapters we discussed evaluation measures for quantifying factual inconsistencies in model generated text. A critical piece towards mitigating such inconsistencies and supporting more reliable language generation is increasing the transparency of our language generation models to enable tracing and attribution of outputs. Though a predominant set of research focuses on post-hoc interpretation and analysis of models, there have been concerns of their evaluation and general applicability (Madsen et al., 2022; Han et al., 2023), and their potential to mislead users (Dinu et al., 2020). Here, we explore an alternative view towards interpretability by developing *interpretable-by-design* models based on task-based linguistic knowledge. Part 2 comprising of two chapters proposes approaches to designing transparent language generation models, grounded in language structure, for two diverse and challenging applications.

This chapter considers the task of automatic text summarization and describes novel approaches to incorporate the source document structure as inductive biases for content selection. The designed module can support transparency in the content selection decisions, potentially helping in tracing hallucinations to content not present in the source document. Research described in this chapter was conducted in collaboration with Artidoro Pagnoni, Jay Yoon Lee, Dheeraj Rajagopal, Jaime Carbonell and Yulia Tsvetkov and was presented in an EACL 2021 publication (Balachandran et al., 2021).

4.1 Introduction

Text summarization aims at identifying important information in long source documents and expressing it in human readable summaries. Two prominent methods of generating summaries are *extractive* (Dorr et al., 2003; Nallapati et al., 2017), where important sentences in the source article are selected to form a summary, and *abstractive* (Rush et al., 2015; See et al., 2017), where the model restructures and rephrases essential content into a paraphrased summary.

State of the art approaches to abstractive summarization employ neural encoder-decoder methods that encode the source document as a sequence of tokens producing latent document representations and decode the summary conditioned on the representations. Recent studies suggest that these models suffer from several key challenges. First, since standard training datasets are derived from news articles, model outputs are strongly affected by the layout bias of the articles, with models relying on the leading sentences of source documents (Kryscinski et al., 2019; Kedzie et al., 2018). Second, although they aim to generate paraphrased summaries, abstractive summarization systems often copy long sequences from the source, causing their outputs to resemble extractive summaries (Lin and Ng, 2019; Gehrmann et al., 2018). Finally, current methods do not lend themselves easily to interpretation via intermediate structures (Lin and Ng, 2019), which could be useful for identifying major bottlenecks in summarization models.

To address these challenges, we introduce STRUCTSUM: a framework that incorporates structured document representations into summarization models. STRUCTSUM complements a standard encoder-decoder architecture with two novel components: (1) a *latent-structure attention* module that adapts structured representations (Kim et al., 2017; Liu and Lapata, 2017) for the summarization task, and (2) an *explicit-structure attention* module that incorporates an external linguistic structure (e.g., coreference links). The two complementary components are incorporated and learned jointly with the encoder and decoder, as shown in Figure 4.1.

Encoders with induced latent structures have been shown to benefit several tasks including document classification, natural language inference (Liu and Lapata, 2017; Cheng et al., 2016), and machine translation (Kim et al., 2017). Our latent structure attention module builds upon Liu and Lapata (2017) to model the dependencies between sentences in a document. It uses a variant of Kirchhoff's matrix-tree theorem (Tutte, 1984) to model such dependencies as non-projective tree structures(§4.2.2). The explicit attention module is linguistically-motivated and aims to incorporate inter-sentence links from externally annotated document structures. We incorporate a coreference based dependency graph across sentences, which is then combined with the output of the latent structure attention module to produce a hybrid structure-aware sentence representation (§4.2.3).

We test our framework using the CNN/DM dataset (Hermann et al., 2015) and show in §4.4.1 that it outperforms the base pointer-generator model (See et al., 2017) by up to 1.1 ROUGE-L. We find that the latent and explicit structures are complementary, both contributing to the final performance improvement. Our modules are also orthogonal to the choice of an underlying encoder-decoder architecture, rendering them flexible to be incorporated into other advanced models.

Quantitative and qualitative analyses of summaries generated by STRUCTSUM and baselines (§4.4), reveal that structure-aware summarization mitigates the news corpora layout bias by improving the coverage of source document sentences. Additionally, STRUCTSUM reduces the bias of copying large sequences from the source, inherently making the summaries more abstractive by generating $\sim$15% more novel n-grams than a competitive baseline. We also show examples of the learned interpretable sentence dependency structures, motivating further research for structure-aware modeling.[1]

4.2 Structured Representations in LSTMs: STRUCTSUM

Consider a source document $\boldsymbol{x}$ consisting of n sentences $\{\boldsymbol{s}\}$ where each sentence $\boldsymbol{s}_i$ is composed of a sequence of words. Document summarization aims to map the source document to a target summary $\boldsymbol{y}$ of m words $\{y\}$. A typical neural abstractive summarization system is an attentional sequence-to-sequence model that encodes the input sequence $\boldsymbol{x}$ as a continuous sequence of tokens $\{w\}$ using a standard encoder (Hochreiter and Schmidhuber, 1997; Vaswani et al., 2017). The encoder produces a set of hidden representations $\{\mathbf{h}\}$. A decoder maps the previously generated token y_{t-1} to a hidden state and computes a soft attention probability distribution $p(\mathbf{a}_t \mid \boldsymbol{x}, \boldsymbol{y}_{1:t-1})$ over encoder hidden states. A distribution p over the vocabulary is computed at every time step t and the network is trained using the negative log likelihood loss: $\text{loss}_t = -\log p(y_t)$.

[1] Code and data available at: `https://github.com/vidhishanair/structured_summarizer`

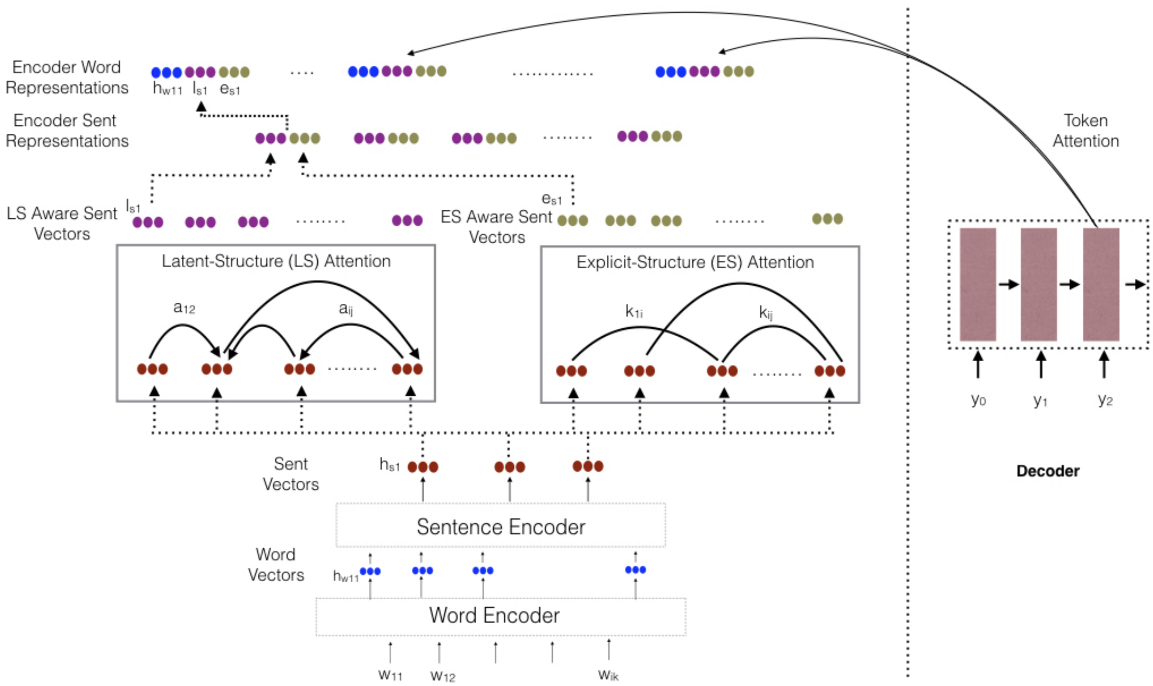

Figure 4.1: STRUCTSUM incorporates Latent Structure (LS) §4.2.2 and Explicit Structure (ES) §4.2.3 Attention to produce structure-aware representations. Here, STRUCTSUM augments the Pointer-Generator model, but the methodology that we proposed is general, and it can be applied to other encoder-decoder summarization systems

STRUCTSUM modifies the above architecture as follows. We aggregate the token representations from the encoder to form sentence representations as in hierarchical encoders (Yang et al., 2016). We then use implicit- and explicit-structure attention modules to augment the sentence representations with sentence dependency information, leveraging both a learned latent structure and an external structure from other NLP modules. The attended vectors are then passed to the decoder, which produces the output abstractive summary. In the rest of this section, we describe our framework architecture, shown in Figure 4.1, in detail.

4.2.1 Sentence Representations

We consider an encoder which takes a sequence of words in a sentence $\boldsymbol{s}_i = \{w\}$ as input and produces contextual hidden representation for each word $\mathbf{h}_{w_{ik}}$, where w_{ik} is the kth word of the ith sentence, $k = 1 : q$ and q is the number of words in the sentence $\boldsymbol{s}_i$. The word hidden representations are max-pooled at the sentence level and passed through a sentence-encoder, which produces new hidden sentence representations for each sentence $\mathbf{h}_{\boldsymbol{s}_i}$. The sentence hidden representations are then passed as inputs to the latent and explicit structure attention modules.

4.2.2 Latent Structure (LS) Attention

We model the latent structure of a source document as a non-projective dependency tree of sentences and force a pairwise attention module to automatically induce this tree. We denote the marginal probability of a dependency edge as $a_{ij} = p(z_{ij} = 1)$ where z_{ij} is the latent variable representing the edge from sentence i to sentence j. We parameterize the unnormalized pairwise scores between sentences with a neural network and use the Kirchoff's matrix tree theorem (Tutte, 1984) to compute the marginal probability of a dependency edge between any two sentences.

Specifically, we decompose the representation of a sentence $\boldsymbol{s}_i$ into a *semantic* vector $\mathbf{g}_{\boldsymbol{s}_i}$ and *structure* vector $\mathbf{d}_{\boldsymbol{s}_i}$ as $\mathbf{h}_{\boldsymbol{s}_i} = [\mathbf{g}_{\boldsymbol{s}_i}; \mathbf{d}_{\boldsymbol{s}_i}]$. Using the structure vectors $\mathbf{d}_{\boldsymbol{s}_i}, \mathbf{d}_{\boldsymbol{s}_j}$, we compute a score f_{ij} between sentence pairs (i, j) (where sentence i is the parent node of sentence j) and a score r_i (where the sentence $\boldsymbol{s}_i$ is the root node):

$$f_{ij} = F_p(\mathbf{d}_{\boldsymbol{s}_i})^T W_a F_c(\mathbf{d}_{\boldsymbol{s}_j}) \text{ and } r_i = F_r(\mathbf{d}_{\boldsymbol{s}_i})$$

where F_p, F_c and F_r are linear-projection functions that build representations for the parent, child and root nodes respectively, and W_a is the weight for bilinear transformation. Here, f_{ij} is the edge weight between nodes (i, j) in a weighted adjacency graph $\mathbf{F}$ and is computed for all pairs of sentences. Using f_{ij} and r_i, we compute normalized attention scores a_{ij} and a_i^r using a variant of Kirchhoff's matrix-tree theorem where a_{ij} is the marginal probability of a dependency edge between sentences (i, j) and a_i^r is the probability of sentence i being the root.

Using these probabilistic attention weights and the semantic vectors $\{\mathbf{g}_{\boldsymbol{s}}\}$, we compute the attended sentence representations as:

$$\begin{aligned}
\mathbf{p}_{\boldsymbol{s}_i} &= \sum_{j=1}^{n} a_{ji}\mathbf{g}_{\boldsymbol{s}_j} + a_i^r \mathbf{g}_{root} \\
\mathbf{c}_{\boldsymbol{s}_i} &= \sum_{j=1}^{n} a_{ij}\mathbf{g}_{\boldsymbol{s}_i} \\
\mathbf{l}_{\boldsymbol{s}_i} &= \tanh(W_r[\mathbf{g}_{\boldsymbol{s}_i}, \mathbf{p}_{\boldsymbol{s}_i}, \mathbf{c}_{\boldsymbol{s}_i}])
\end{aligned}$$

where $\mathbf{p}_{\boldsymbol{s}_i}$ is the context vector gathered from possible parents of sentence i, $\mathbf{c}_{\boldsymbol{s}_i}$ is the context vector gathered from possible children, and $\mathbf{g}_{root}$ is a special embedding for the root node. Here, the updated sentence representation $l_{\boldsymbol{s}_i}$ incorporates the implicit structural information.

4.2.3 Explicit Structure (ES) Attention

Following Durrett et al. (2016), who showed that modeling coreference knowledge through anaphora constraints leads to improved clarity or grammaticality, we incorporate cross-sentence coreference links as the source of explicit structure. First, we use an off-the-shelf coreference parser[2] to identify coreferring mentions. We then build a coreference based sentence graph by adding a link between sentences $(\boldsymbol{s}_i, \boldsymbol{s}_j)$, if they have any coreferring mentions. This graph is converted into a weighted graph by incorporating a weight on the edge between two sentences that is proportional to the number of unique coreferring mentions between them. We normalize these edge weights for every sentence, effectively building a weighted adjacency matrix $\mathbf{K}$ where k_{ij} is given by:

$$\begin{aligned}
k_{ij} &= P(z_{ij} = 1) && (4.1) \\
&= \frac{count(m_i \bigcap m_j) + \epsilon}{\sum_{v=1}^{n} count(m_i \bigcap m_v)} && (4.2)
\end{aligned}$$

[2] https://github.com/huggingface/neuralcoref/

where m_i denotes the set of unique mentions in sentence $\boldsymbol{s}_i$, $(m_i \bigcap m_j)$ denotes the set of co-referring mentions between the two sentences, and z is a latent variable representing a link in the coreference sentence graph. $\epsilon = 5e^{-4}$ is a smoothing hyperparameter.

Given contextual sentence representations $\{\mathbf{h}_{\boldsymbol{s}}\}$ and our explicit coreference-based weighted adjacency matrix $\mathbf{K}$, we learn an explicit structure-aware representation as follows:

$$\mathbf{u}_{\boldsymbol{s}_i} = \tanh(F_u(\mathbf{h}_{\boldsymbol{s}_i}))$$
$$\mathbf{t}_{\boldsymbol{s}_i} = \sum_{j=1}^{p} k_{ij}\mathbf{u}_{\boldsymbol{s}_j}$$
$$\mathbf{e}_{\boldsymbol{s}_i} = \tanh(F_e(\mathbf{t}_{\boldsymbol{s}_i}))$$

where F_u and F_e are linear projections and $\mathbf{e}_{\boldsymbol{s}_i}$ is an updated sentence representation which incorporates explicit structural information.

Finally, to combine the two structural representations, we concatenate the latent and explicit sentence vectors as: $\mathbf{h}_{\boldsymbol{s}_i} = [\mathbf{l}_{\boldsymbol{s}_i}; \mathbf{e}_{\boldsymbol{s}_i}]$ to form encoder sentence representations of the source document. To provide every token representation with the context of the entire document, the token representations are concatenated with their corresponding structure-aware sentence representation: $\mathbf{h}_{w_{ij}} = [\mathbf{h}_{w_{ij}}; \mathbf{h}_{\boldsymbol{s}_i}]$ where $\boldsymbol{s}_i$ is the sentence to which the word w_{ij} belongs. The resulting structure-aware token representations can be used to directly replace previous token representations as input to the decoder.

4.3 Experiments

Dataset: We evaluate our approach on the CNN/Daily Mail corpus[3] (Hermann et al., 2015; Nallapati et al., 2016) and use the same preprocessing steps as in See et al. (2017). The CNN/DM has 287226/13368/11490 train/val/test samples respectively. The reference summaries have an average of 66 tokens ($\sigma = 26$) and 4.9 sentences. Differing from See et al. (2017), we truncate source documents to 700 tokens instead of 400 in training and validation sets to model longer documents with more sentences. All our experiments were trained on Nvidia GTX Titan X GPUs.

Base Model: Although StructSum framework can be incorporated in any encoder-decoder framework with structure-aware representations, for our experiments we chose the pointer-generator model (See et al., 2017) as the base model, due to its simplicity and ubiquitous usage as a neural abstractive summarization model across different domains (Liu et al., 2019b; Krishna et al., 2020). The word and sentence encoders are BiLSTM and the decoder is a BiLSTM with a pointer based copy mechanism. We re-implement the base pointer-generator model and augment it with the StructSum modules described in §4.2 and hence our model can be directly compared to it.

Baselines: In addition to the base model, we compare StructSum with the following baselines:
Tan et al. (2017): This is a graph-based attention model that is closest in spirit to the method we present in this work. A graph attention module is used to learn attention between sentences, but it cannot be easily used to induce interpretable document structures, since its attention scores are not constrained to learn structure. On

[3]https://cs.nyu.edu/~kcho/DMQA/

Model	ROUGE 1	ROUGE 2	ROUGE L
Pointer-Generator (See et al., 2017)	36.44	15.66	33.42
Pointer-Generator + Coverage (See et al., 2017)	39.53	**17.28**	36.38
Graph Attention (Tan et al., 2017)	38.10	13.90	34.00
Pointer-Generator + DiffMask (Gehrmann et al., 2018)	38.45	16.88	35.81
Pointer-Generator (Re-Implementation)	35.55	15.29	32.05
Pointer-Generator + Coverage (Re-Implementation)	39.07	16.97	35.87
Latent-Structure (LS) Attention	39.52	16.94	36.71
Explicit-Structure (ES) Attention	**39.63**	16.98	36.72
LS + ES Attention	39.62	17.00	**36.95**

Table 4.1: Evaluation of summarization models on the CNN/DM dataset.Published abstractive summarization baseline scores are on top. STRUCTSUM results that incorporate latent and explicit document structure into the base models. STRUCTSUM's utility is on par with the base models, while introducing additional benefits of better abstractiveness and intrepretability shown in §4.4.

top of latent and interpretable structured attention between sentences, STRUCTSUM introduces an explicit structure component to inject external document structure, which distinguishes it from Tan et al. (2017). Gehrmann et al. (2018): This work introduces a separate content selector which tags words and phrases to be copied. The DiffMask variant is an end-to-end variant like ours and hence is included in our baselines. We compare STRUCTSUM with the DiffMask experiment.[4]

Hyperparameters: Our encoder uses 256 hidden states for both directions in the one-layer BiLSTM, and 512 for the single-layer decoder. We use the Adagrad optimizer (Duchi et al., 2011) with a learning rate of 0.15 and an initial accumulator value of 0.1. We do not use dropout and use gradient-clipping with a maximum norm of 2. We selected the best model using early stopping based on the ROUGE score on the validation dataset as our criteria. We also used the coverage penalty during inference as shown in Gehrmann et al. (2018). For decoding, we use beam-search with a beam width of 3. We did not observe significant improvements with higher beam widths.

4.4 Results

A standard ROUGE metric does not shed meaningful light into the quality of summaries across important dimensions. As a recall-based metric it is not suitable for assessing the abstractiveness of summarization; it is also agnostic to layout biases and does not facilitate intrepretability of model decisions. We thus adopt automatic metrics tailored to evaluating separately each of these aspects. We compare STRUCTSUM to our base model, the pointer-generator network with coverage (See et al., 2017) and the reference.

4.4.1 Automatic Metrics

We first conduct a standard comparison of generated summaries with reference summaries using ROUGE-1,2 and L (Lin, 2004) F1[5] metric. Table 4.1 shows the results. We first observe that introducing the latent structures

[4] The best results from Gehrmann et al. (2018) outperform DiffMask experiment, but they use inference-time hard masking which can be applied on ours. Our baselines also exclude Reinforcement Learning (RL) based systems as they are not directly comparable, but our approach can be introduced in an encoder-decoder based RL system. Since we do not incorporate any pretraining, we do not compare with recent contextual representation based models (Liu and Lapata, 2019).

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

and explicit structures independently improves our performance on ROUGE-L. It suggests that modeling dependencies between sentences helps the model compose better long sequences compared to baselines. We see small improvements in ROUGE-1 and ROUGE-2, hinting that we retrieve similar content words as the baseline but compose them into better contiguous sequences. As both ES and LS independently get similar performance, the results show that LS attention induces good latent dependencies that make up for pure external coreference knowledge.

Finally, our combined model which uses both Latent and Explicit structure performs the best with an improvement of **1.08 points** in ROUGE-L and **0.6 points** in ROUGE-1 over base pointer-generator model (statistically significant for 11490 samples at p=0.05 using Wilson Confidence Test). It shows that the latent and explicit information are complementary and a model can jointly leverage them to produce better summaries. Additionally, we find that structural inductive bias helps a model to converge faster. The combined LS+ES Attention model converges in 126K iterations in comparison to ~230K iterations required for the pointer-generator network.

While ROUGE is a popular metric used for evaluating summarization models, it is limited to only evaluating n-gram overlap while ignoring semantic correctness. Hence, we compared our method with the baseline Pointer-Generator model using the BERTScore metric (Zhang et al., 2020b). We observe that our model improves BERTScore by 9 points (12.3 for Pointer-Generator v/s 21.7 for StructSum) showing that our model is able to generate semantically correct content.

4.4.2 Abstractiveness

Despite being an abstractive model, the pointer-generator model tends to copy very long sequences of words including whole sentences from the source document (also observed by Gehrmann et al. (2018)). We use two metrics to evaluate the abstractiveness of the model:

Copy Length: Table 4.2 shows a comparison of the average length (Copy Len) of contiguous copied sequences from the source document (greater than length 3). We observe that the pointer-generator baseline on average copies 16.61 continuous tokens from the source which shows the extractive nature of the model. This indicates that pointer networks, aimed at combining advantages from abstractive and extractive methods by allowing to copy content from the input document, tend to skew towards copying, particularly in this dataset. A consequence of this is that the model fails to interrupt copying at desirable sequence length. In contrast, modeling document structure through STRUCTSUM reduces the length of copied sequences to 9.13 words on average reducing the bias of copying sentences entirely. This average is closer to the reference (5.07 words) in comparison, without sacrificing task performance. STRUCTSUM learns to stop when needed, while still generating coherent summaries.

Novel N-Grams: The proportion of novel n-grams generated has been used in the literature to measure the degree of abstractiveness of summarization models (See et al., 2017). Figure 4.2 compares the percentage of novel n-grams in STRUCTSUM as compared to the baseline model. Our model produces novel trigrams 21.0% of the time and copies whole sentences only 21.7% of the time. In comparison, the pointer-generator network has only 6.1% novel trigrams and copies entire sentences 51.7% of the time. This shows that STRUCTSUM on average generates 14.7% more novel n-grams in comparison to the pointer-generator baseline.

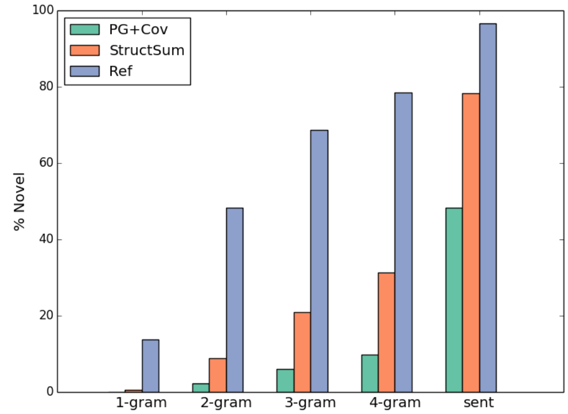

Figure 4.2: Comparison of % Novel n-grams between STRUCTSUM, Pointer-Generator+Coverage and the Reference. Here, "sent" indicates full novel sentences.

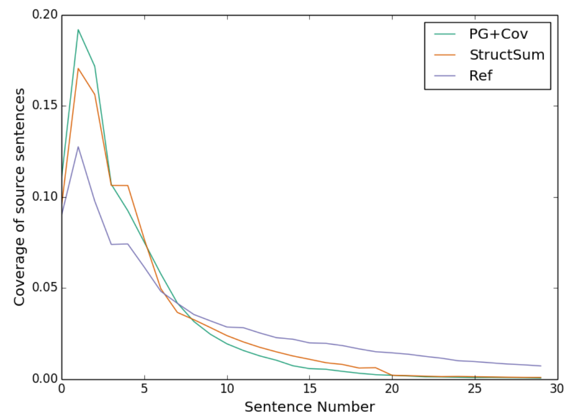

Figure 4.3: Coverage of source sentences in summary. Here the x-axis is the sentence position in the source article and y-axis shows the normalized count of sentences in that position copied to the summary.

4.4.3 Coverage

A direct outcome of copying shorter sequences is being able to cover more content from the source document within given length constraints. We observe that this leads to better summarization performance. We compute coverage by computing the number of source sentences from which contiguous sequences greater than length 3 are copied in the summary. Table 4.2 shows a comparison of the coverage of source sentences in the summary content. While the baseline pointer-generator model only copies from 12.1% of the source sentences, STRUCTSUM copies content from 24.0% of the source sentences. Additionally, the average length of the summaries produced by STRUCTSUM remains mostly unchanged at 66 words on average compared to 61 of the baseline model. This indicates that STRUCTSUM produces summaries that draw from a wider selection of sentences from the original article compared to the baseline models.

4.4.4 Layout Bias

Neural abstractive summarization methods applied to news articles are typically biased towards selecting and generating summaries based on the first few sentences of the articles. This stems from the structure of

	Copy Len	Coverage
PG+Cov	16.61	12.1 %
STRUCTSUM	9.13	24.0 %
Reference	5.07	16.7 %

Table 4.2: Results of analysis of copying and coverage distribution over the source sentences on CNN/DM test set. Copy Len denotes the average length of copied sequences; Coverage – coverage of source sentences.

	Coref	NER	Coref+NER
precision	0.29	0.19	0.33
recall	0.11	0.08	0.09

Table 4.3: Precision and recall of ES and LS shared edges

news articles, which present the salient information of the article in the first few sentences and expand in the subsequent ones. As a result, the LEAD 3 baseline, which selects the top three sentences of an article, is widely used in the literature as a strong baseline to evaluate summarization models applied to the news domain (Narayan et al., 2018). Kryscinski et al. (2019) observed that the current summarization models learn to exploit the layout biases of current datasets and offer limited diversity in their outputs.

To analyze whether STRUCTSUM also holds the same layout biases, we compute a distribution of source sentence indices that are used for copying content (copied sequences of length 3 or more are considered). Figure 4.3 shows the distributions of source sentences covered in the summaries. The coverage of sentences in the reference summaries shows a high proportion of the top 5 sentences of any article being copied to the summary. Additionally, the reference summaries have a smoother tail end distribution with relevant sentences in all positions being copied. It shows that a smooth distribution over all sentences is a desirable feature. We notice that the pointer-generator framework have a stronger bias towards the beginning of the article with a high concentration of copied sentences within the top 5 sentences of the article. In contrast, STRUCTSUM improves coverage slightly having a lower concentration of top 5 sentences and copies more tail end sentences than the baselines. However, although the modeling of structure does help, our model has a reasonable gap compared to the reference distribution. We see this as an area of improvement and a direction for future work.

4.5 Analysis of Induced Document Structures

Similar to Liu and Lapata (2017), we also look at the quality of the intermediate structures learned by the model. We use the Chu-Liu-Edmonds algorithm (Chu and Liu, 1965; Edmonds, 1967) to extract the maximum spanning tree from the attention score matrix as our sentence structure. Table 4.4 shows the frequency of various tree depths. We find that the average tree depth is 2.9 and the average proportion of leaf nodes is 88%, consistent with results from tree induction in document classification (Ferracane et al., 2019). Further, we compare latent trees extracted from STRUCTSUM with undirected graphs based on coreference, on NER, or on both. These are constructed similarly to our explicit coreference based sentence graphs in §4.2.3 by linking sentences with overlapping coreference mentions or named entities. We measure the similarity between the learned latent trees and the explicit graphs through precision and recall over edges. The results are shown in Table 4.3. We observe that our latent graphs have low recall with the linguistic graphs showing that our

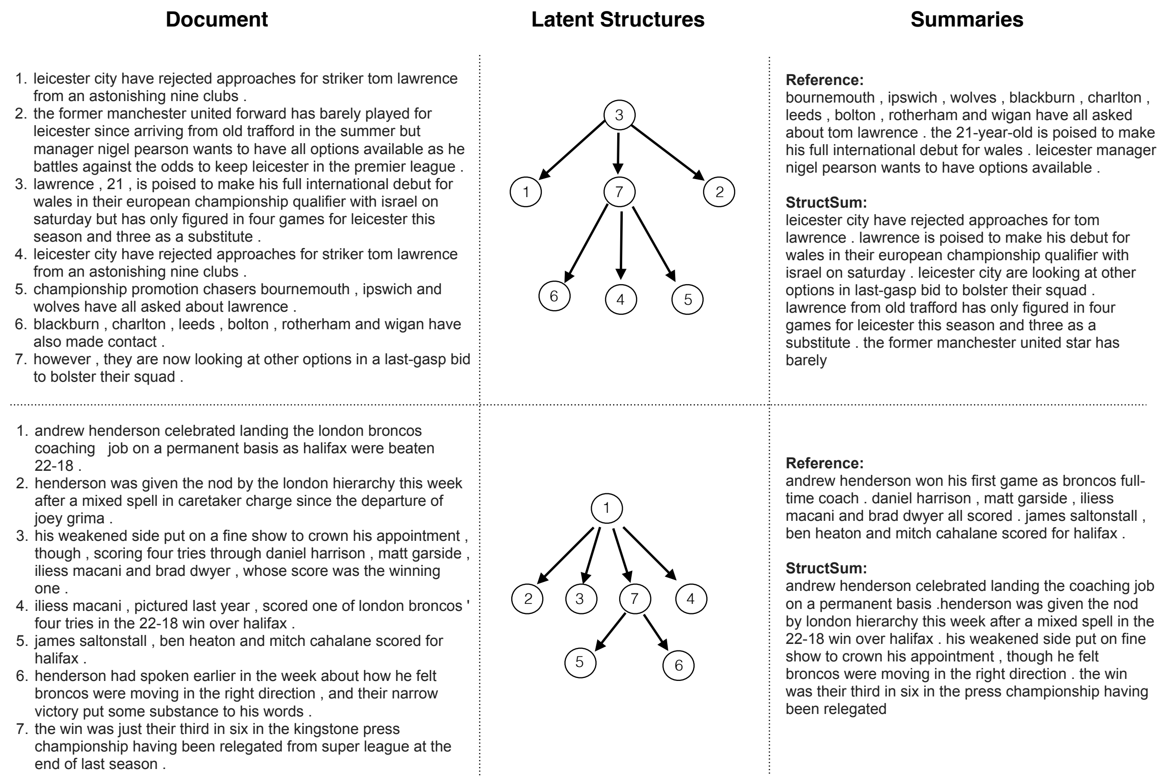

Figure 4.4: Examples of induced structures and generated summaries.

Depth	2	3	4	5+
StructSum	29.3%	53.7%	14.4%	2.6%

Table 4.4: Distribution of latent tree depth.

latent graphs do not capture the coreference or named entity overlaps explicitly, suggesting that the latent and explicit structures capture complementary information.

Figure 4.4 shows qualitative examples of induced structures along with summaries from the STRUCTSUM. The first example shows a tree with sentence 3 chosen as root, which was the key sentence mentioned in the reference. In both examples, the sentences in the lower level of the dependency tree contribute less to the generated summary. Similarly, in the examples source sentences used to generate summaries tend to be closer to the root node. In the first summary, all source content sentences used in the summary are either the root node or within depth 1 of the root node. In the second example, 4 out of 5 source sentences were at depth=1 in the tree. In both examples, generated summaries diverged from the reference by omitting certain sentences used in the reference. These sentences are in the lower section of the tree, providing insights on which sentences were preferred for the summary generation. We also see in example 1 that the latent structures cluster sentences based on the main topic of the document. Sentences 1,2,3 differ from sentences 5,6,7 in the topic discussed and our model clustered the two sets separately.

4.6 Related Work

Data-driven neural summarization falls into *extractive* (Cheng et al., 2016; Zhang et al., 2018b) or *abstractive* (Rush et al., 2015; See et al., 2017; Gehrmann et al., 2018; Chen and Bansal, 2018). Pointer-generator (See et al., 2017) learns to either generate novel in-vocabulary words or copy from the source. It has been the foundation for much work on abstractive summarization (Gehrmann et al., 2018; Hsu et al., 2018; Song et al., 2018). Our model extends it by incorporating latent/explicit structure, but these extensions are applicable to any other encoder-decoder architecture. For example, a follow-up study has already shown benefits of our method in multi-document summarization (Chowdhury et al., 2020).

In the pre-neural era, document structure played a critical role in summarization (Leskovec et al., 2004; Litvak and Last, 2008; Liu et al., 2015; Durrett et al., 2016; Kikuchi et al., 2014). More recently Song et al. (2018) infuse source syntactic structure into the pointer-generator using word-level syntactic features and augmenting them to decoder copy mechanism. In contrast, we model sentence dependencies as latent structures and explicit coreference structures; we do not use heuristics or salient features. Li et al. (2018) propose structural compression and coverage regularizers incorporating structural bias of target summaries while we model the structure of the source document. Fremann and Klementiev (2019) induce latent structures for aspect based summarization, Cohan et al. (2018) focus on summarization of scientific papers, Isonuma et al. (2019) reviews unsupervised summarization, Mithun and Kosseim (2011) use discourse structures to improve coherence in blog summarization and Ren et al. (2018) use sentence relations for multi-document summarization. These are complementary directions to our work. To our knowledge, StructSum is the first to jointly incorporate latent and explicit document structure in a summarization framework.

4.7 Conclusions and Future Work

In this work, we propose the framework StructSum for incorporating latent and explicit document structure in neural abstractive summarization. We introduce a novel explicit-attention module which incorporates external linguistic structures, instantiating it with coreference links. We show that our framework improves the abstractiveness and coverage of generated summaries, and helps mitigate layout biases associated with prior models. We present an extensive evaluation of StructSum- along abstractiveness, coverage, and layout quantitatively.

In this work, we explored inducing and extracting model interpretations for summarization. An important next step is to close the loop towards model development by leveraging findings from model explanations to improve the model design. In Ahia et al. (2023), we explored this for classification by improving models via supervising model explanations. Further research in this direction would be very fruitful to identify and address model limitations. An interesting direction for future work would be to investigate the role of document structures in pretrained language models (Lewis et al., 2019; Liu and Lapata, 2019). Initial studies on such language models have found that pretrained models implicitly learn language structure in their parameters (Tenney et al., 2019). Expanding on this finding, exploring how such learned knowledge could be used to produce model interpretations and understanding in such pretrained models would be extremely beneficial.

Chapter 5

Interpretable Dialog Modeling by Leveraging Conversational Structure

This chapter considers the task of dialog modeling for negotiation and describes novel approaches that incorporate pragmatic strategies in a negotiation dialogue using graph neural networks. The model explicitly incorporates dependencies between sequences of strategies to enable improved and interpretable prediction of next optimal strategies, given the dialogue context. Research described in this chapter was conducted in collaboration with Rishabh Joshi, Shikhar Vashishth, Alan Black and Yulia Tsvetkov and was presented in an ICLR 2021 publication (Joshi et al., 2021).

5.1 Introduction

Negotiation is ubiquitous in human interaction, from e-commerce to the multi-billion dollar sales of companies. Learning how to negotiate effectively involves deep pragmatic understanding and planning the dialogue strategically (Thompson, 2001; Bazerman et al., 2000; Pruitt, 2013).

Modern dialogue systems for collaborative tasks such as restaurant or flight reservations have made considerable progress by modeling the dialogue history and structure explicitly using the semantic content, like slot-value pairs (Larionov et al., 2018; Young, 2006), or implicitly with encoder-decoder architectures (Sordoni et al., 2015; Li et al., 2016b). In such tasks, users communicate explicit intentions, enabling systems to map the utterances into specific intent slots (Li et al., 2020). However, such mapping is less clear in complex non-collaborative tasks like *negotiation* (He et al., 2018) and *persuasion* (Wang et al., 2019), where user intent and most effective strategies are hidden. Hence, along with the generated dialogue, the strategic choice of framing and the sequence of chosen strategies play a vital role, as depicted in Figure 5.1. Indeed, prior work on negotiation dialogues has primarily focused on optimizing dialogue strategies—from high-level task-specific strategies (Lewis et al., 2017), to more specific task execution planning (He et al., 2018), to fine-grained planning of linguistic outputs given strategic choices (Zhou et al.,

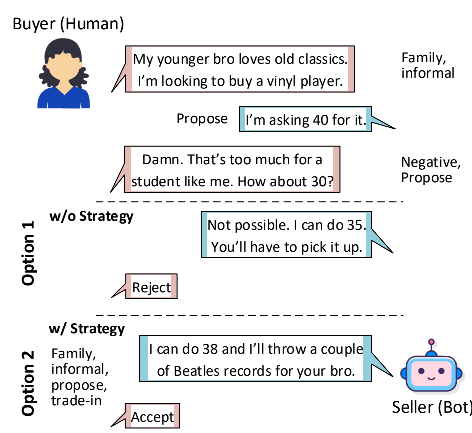

Figure 5.1: Both options are equally plausible and fluent, but a response with effective pragmatic strategies leads to a better deal.

2019). These studies have confirmed that it is crucial to control for pragmatics of the dialogue to build effective negotiation systems.

To model the explicit dialogue structure, prior work incorporated Hidden Markov Models (HMMs) (Zhai and Williams, 2014; Ritter et al., 2010), Finite State Transducers (FSTs) (Zhou et al., 2020) and RNNs (He et al., 2018; Shi et al., 2019). While RNN-based models lack interpretability, HMM- and FST-based approaches may lack expressivity. In this paper, we hypothesize that Graph Neural Networks (GNNs) (Wu et al., 2020) can combine the benefits of interpretability and expressivity because of their effectiveness in encoding graph-structured data through message propagation. While being sufficiently expressive to model graph structures, GNNs also provide a natural means for interpretation via intermediate states (Xie and Lu, 2019; Pope et al., 2019).

We propose DIALOGRAPH, an end-to-end negotiation dialogue system that leverages Graph Attention Networks (GAT) (Veličković et al., 2018) to model complex negotiation strategies while providing interpretability for the model via intermediate structures. DIALOGRAPH incorporates the recently proposed hierarchical graph pooling based approaches (Ranjan et al., 2020) to learn the associations between negotiation strategies, including conceptual and linguistic strategies and dialogue acts, and their relative importance in predicting the best sequence. We focus on buyer–seller negotiations in which two individuals negotiate on the price of an item through a chat interface, and we model the seller's behavior on the CraigslistBargain dataset (He et al., 2018).[1] We demonstrate that DIALOGRAPH outperforms previous state-of-art methods on strategy prediction and downstream dialogue responses. This paper makes several contributions. First, we introduce a novel approach to model negotiation strategies and their dependencies as graph structures, via GNNs. Second, we incorporate these learned graphs into an end-to-end negotiation dialogue system and demonstrate that it consistently improves future-strategy prediction and downstream dialogue generation, leading to better negotiation deals (sale prices). Finally, we demonstrate how to interpret intermediate structures and learned sequences of strategies, opening-up the black-box of end-to-end strategic dialogue systems.[2]

5.2 Conversational Graphs via Structured Transformers - DIALOGRAPH

We introduce DIALOGRAPH, a modular end-to-end dialogue system, that incorporates GATs with hierarchical pooling to learn pragmatic dialogue strategies jointly with the dialogue history. DIALOGRAPH is based on a hierarchical encoder-decoder model and consists of three main components: (1) *Hierarchical Dialogue Encoder*, which learns a representation for each utterance and encodes its local context; (2) *Structure Encoder* for encoding sequences of negotiation strategies and dialogue acts; and (3) *Utterance Decoder*, which finally generates the output utterance. Formally, our dialogue input consists of a sequence of tuples, $\mathcal{D} = [(u_1, da_1, ST_1), (u_2, da_2, ST_2), ..., (u_n, da_n, ST_n)]$ where u_i is the utterance, da_i is the coarse dialogue act and $ST_i = \{st_{i,1}, st_{i,2}, \ldots, st_{i,k}\}$ is the set of k fine-grained negotiation strategies for the utterance u_i.[3] The dialogue context forms the input to (1) and the previous dialogue acts and negotiation strategies form the input to (2). The overall architecture is shown in Figure 5.2. In what follows, we describe DIALOGRAPH in detail.

[1]We focus on the seller's side following Zhou et al. (2019) who devised a set of strategies specific to maximizing the seller's success. Our proposed methodology, however, is general.

[2]Code, data and a demo system is released at `https://github.com/rishabhjoshi/DialoGraph_ICLR21`

[3]For example, in an utterance *Morning! My bro destroyed my old kit and I'm looking for a new pair for $10*, the coarse dialogue act is *Introduction*, and the finer grained negotiation strategies include *Proposing price*, *Being informal* and *Talking about family for building rapport*.

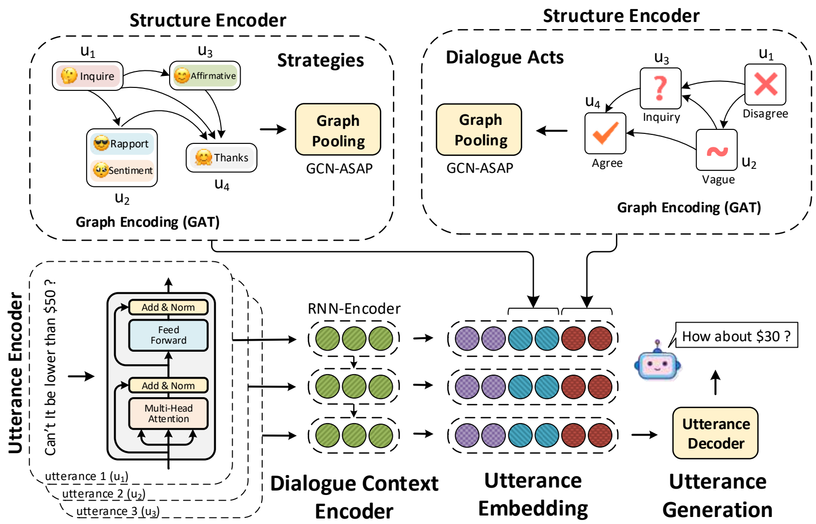

Figure 5.2: Overview of DIALOGRAPH. At time t, utterance u_t is encoded using BERT and then passed to the Dialogue Context Encoder to generate the dialogue representation. This representation is enriched with the encodings of explicit strategy and dialogue act sequences using the Structure Encoders which is then used to condition the Utterance decoder. Please refer to §5.2 for details.

5.2.1 Hierarchical Dialogue Encoder

A dialogue context typically comprises of multiple dialogue utterances which are sequential in nature. We use hierarchical encoders for modeling such sequential dialogue contexts (Jiao et al., 2019). To encode the utterance u_t at time t, we use the pooled representations from BERT (Devlin et al., 2019) to obtain the corresponding utterance embedding $\boldsymbol{e}_t$. We then pass the utterance embeddings through a GRU to obtain the dialogue context encoding till time t, denoted by $\boldsymbol{h}_t^U$.

5.2.2 Structure Encoder

Our Structure Encoder is designed to model the graph representations of the strategies and dialogue acts using GATs and output their structural representations. These structural representations are used to predict the next set of strategies and dialogue acts and enrich the encoded dialogue representation. Below we describe the Structure Encoder for negotiation strategies.

We model the sequence of negotiation strategies, $ST = [ST_1, ST_2, \ldots, ST_t]$ by creating a directed graph, where ST_i is the set of k fine-grained negotiation strategies for the utterance u_i. Formally, we define a graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, X)$ with $|\mathcal{E}|$ edges and $N = |\mathcal{V}|$ nodes where each node $v_i \in \mathcal{V}$ represents a particular negotiation strategy for an utterance and has a d-dimensional feature representation denoted by $\boldsymbol{z}_i$. $\boldsymbol{Z} \in Nd$ denotes the feature matrix of the nodes and $\boldsymbol{A} \in NN$ represents the adjacency matrix, where N is the total number of nodes (strategies) that have occurred in the conversation till that point. Therefore, each node represents a strategy-utterance pair.

We define the set of edges as $\mathcal{E} = \{(a, b)\}; a, b \in \mathcal{V}$ where a and b denote strategies at utterances u_a and u_b, present at turns t_a and t_b, such that $t_b > t_a$. In other words, we make a directed edge from a particular node (strategy in an utterance) to all the consecutive nodes. This ensures a direct connection from all the previous strategies to the more recent ones.[4] In the same way, we form the graph out of the sequence of dialogue acts. These direct edges and learned edge attention weights help us interpret the dependence and influence of strategies on each other.

To get the structural representations from the strategy graphs, we pass them through a hierarchical graph pooling based encoder, which consists of l layers of GAT, each followed by the Adaptive Structure Aware Pooling (ASAP) layer (Ranjan et al., 2020). As part of the ASAP layer, the model first runs GAT over the input graph representations to obtain structurally informed representations of the nodes. Then a cluster assignment step is performed which generates a cluster assignment matrix, S, which tells the model which nodes come in a similar structural context. After that, the clusters are ranked and then the graph is pooled by taking the top few clusters as new nodes and forming edges between them using the existing graph. This way the size of the graph is reduced at every step which leads to a structurally informed graph representation. We take advantage of the cluster formulation to obtain the associations between the negotiation strategies, as identified from the cluster assignment matrix, S. These association scores can later be used to interpret which strategies are associated with each other and tend to co-occur in similar contexts. Moreover, we also use the node attention scores from GAT to interpret the influence of different strategies on the representation of a particular strategy, which essentially gives the dependence information between strategies.

In this way, the structure representation is learned and accumulated in a manner that preserves the structural information (Ying et al., 2018; Lee et al., 2019). After each pooling step, the graph representation is summarized using the concatenation of *mean* and *max* of the node representations. The summaries are then added and passed through fully connected layers to obtain the final structural representation of the strategies $\boldsymbol{h}_t^{ST}$. We employ a similar Structure Encoder to encode the graph obtained from the sequence of dialogue acts, to obtain $\boldsymbol{h}_t^{da}$.

5.2.3 Utterance Decoder

The utterance decoder uses the dialogue context representation and structural representations of dialogue acts and negotiation strategies to produce the dialogue response (next utterance). We enrich the dialogue representation by concatenating the structural representations before passing it to a standard greedy GRU (Cho et al., 2014) decoder. This architecture follows Zhou et al. (2020), who introduced a dynamic negotiation system that incorporates negotiation strategies and dialogue acts via FSTs. We thus follow their utterance decoder architecture to enable direct baseline comparison. For the jth word of utterance u_{t+1}, w_{t+1}^{j}, we condition on the previous word w_{t+1}^{j-1} to calculate the probability distribution over the vocabulary as $\boldsymbol{p}_{t+1}^{w_j} = \mathrm{softmax}(\mathrm{GRU}(\boldsymbol{h}_t, \boldsymbol{w}_{t+1}^{j-1}))$ where $\boldsymbol{h}_t = [\boldsymbol{h}_t^{u}; \boldsymbol{h}_t^{ST}; \boldsymbol{h}_t^{da}]$ and $[;]$ represents the concatenation operator. For encoding the price, we replace all price information in the dataset with placeholders representing the percentage of the offer price. For example, we would replace $\$35$ with $< price - 0.875 >$ if the original selling price is $\$40$. The decoder generates these placeholders which are then replaced with the calculated price before generating the utterance.

5.2.4 Model Training

We use $\boldsymbol{h}_t^{ST}$ to predict the next set of strategies ST_{t+1}, a binary value vector which represents the k-hot representation of negotiation strategies for the next turn. We compute the probability of the jth strategy occurring in u_{t+1} as $\boldsymbol{p}(st_{t+1,j}|\boldsymbol{h}_t^{ST}) = \sigma(\boldsymbol{h}_t^{ST})$. where σ denotes the sigmoid operator. We threshold the

[4] Appendix 1.3.1 shows an example of the graph obtained from a sequence of strategies.

probability by 0.5 to obtain the k-hot representation. We denote the weighted negative log likelihood of strategies $\mathcal{L}_{ST}$ as the loss function of the task of next strategy prediction $\mathcal{L}_{ST} = -\sum_j \delta_j \log(\boldsymbol{p}(st_{t+1,j})) - \sum_k \log(1 - \boldsymbol{p}(st_{t+1,k}))$ where the summation of j are over the strategies present ($st^{'}_{t+1,j} = 1$) and not present ($st^{'}_{t+1,k} = 0$) in the ground truth strategies set, $ST^{'}$. Here δ_j is the positive weight associated with the particular strategy. We add this weight to the positive examples to trade off precision and recall. We set $\delta_j = \text{\# of instances not having strategy j}/\text{\# of instances having strategy j}$.

Similarly, we use $\boldsymbol{h}_t^{da}$ to predict the dialogue act for the next utterance da_{t+1}. Given the target dialogue act $da^{'}_{t+1}$ and the class weights ρ_{da} for the dialogue acts, we denote the class-weighted cross entropy loss over the set of possible dialogue acts, $\mathcal{L}_{DA} = -\rho_{da} \log(\text{softmax}(\boldsymbol{h}_t^{da}))$. We pass $\boldsymbol{h}_t = [\boldsymbol{h}_t^u; \boldsymbol{h}_t^{ST}; \boldsymbol{h}_t^{da}]$ through a linear layer to predict the negotiation success, which is denoted by the sale-to-list ratio $r = (\text{sale price} - \text{buyer target price})/(\text{listed price} - \text{buyer target price})$ (Zhou et al., 2019). We split the ratios into 5 negotiation classes of equal sizes using the training data and use those to predict the success of negotiation. Therefore, given the predicted probabilities for target utterance $u^{'}_{t+1}$ from §5.2.3, target ratio class $y^{'}_r$ and the learnable parameters W_r and b_r, we use the cross entropy loss as the loss for the generation task ($\mathcal{L}_{NLG}$) as well as the negotiation outcome prediction task ($\mathcal{L}_R$), thus $\mathcal{L}_{NLG} = -\sum_{w_j \in u^{'}_{t+1}} \log(\boldsymbol{p}_{t+1}^{w_j})$ and $\mathcal{L}_R = -\sum_{r\in[1,5]} y^{'}_r \log(\text{softmax}(W_r \boldsymbol{h}_t + b_r))$. The $\mathcal{L}_R$ loss optimizes for encoding negotiation strategies to enable accurate prediction of negotiation outcome.

We use hyperparameters α, β and γ to optimize the joint loss $\mathcal{L}_{joint}$, of strategy prediction, dialogue act prediction, utterance generation and outcome prediction together, using the Adam optimizer (Kingma and Ba, 2014), to get $\mathcal{L}_{joint} = \mathcal{L}_{NLG} + \alpha\mathcal{L}_{ST} + \beta\mathcal{L}_{DA} + \gamma\mathcal{L}_R$.

5.3 Experimental Setup

Dataset: We use the CraigslistBargain dataset[5] (He et al., 2018) to evaluate our model. The dataset was created using Amazon Mechanical Turk (AMT) in a negotiation setting where two workers were assigned the roles of buyer and seller respectively and were tasked to negotiate the price of an item on sale.The buyer was additionally given a target price. Both parties were encouraged to reach an agreement while each of the workers tried to get a better deal. We remove all conversations with less than 5 turns. In Table 5.1 we provide the CraiglistBargain dataset statistics along with data sizes after filtering conversations with less than 5 turns. The maximum and average number of turns in any conversation is 47 and 9.2 respectively. Also, the maximum and average number of strategies in an utterance is 13 and 3 respectively.

Data split	Size
Train conversations	5383
Valid conversations	643
Test conversations	656
Filtered train conversations	4828
Filtered valid conversations	561
Filtered test conversations	567
Vocabulary size	13339

Table 5.1: Dataset statistics.

[5] https://github.com/stanfordnlp/cocoa/tree/master/craigslistbargain

We extract from the dataset the coarse dialogue acts as described by He et al. (2018). This includes a list of 10 *utterance dialogue acts*, e.g., *inform*, *agree*, *counter-price*. We augment this list by 4 *outcome dialogue acts*, namely, ⟨*offer*⟩, ⟨*accept*⟩, ⟨*reject*⟩ and ⟨*quit*⟩, which correspond to the actions taken by the users. Negotiation strategies are extracted from the data following Zhou et al. (2019). These include 21 fine-grained strategies grounded in prior economics/behavioral science research on negotiation (Pruitt, 2013; Bazerman et al., 2000,?; Fisher et al., 2011; Lax and Sebenius, 2006; Bazerman et al., 2000), e.g, *negotiate side offers*, *build rapport*, *show dominance*.

Meaning	Dialogue Act	Example	Detector
Greetings	intro	I would love to buy	rule
Ask a question	inquiry	Sure, what's your price	rule
Propose the first price	init-price	I'm on a budget so i could do $5	rule
Proposing a counter price	counter-price	How about $15 and I'll waive the deposit	rule
Unknown	unknown	Hmm, let me think	rule
Agree with the proposal	agree	That works for me	rule
Disagree with a proposal	disagree	Sorry I can't agree to that	rule
Answer a question	inform	This bike is brand new	rule
Using comparatives with existing price	vague-price	That offer is too low	rule
Insist on an offer	insist	Still can I buy it for $ 5. I'm on a tight budget	rule
Offer the price	⟨offer⟩		agent action
Accept the offer	⟨accept⟩		agent action
Reject the offer	⟨reject⟩		agent action
Quit the session	⟨quit⟩		agent action

Table 5.2: The list of dialogue acts that we use to annotate the data.

We provide the details about the dialogue acts that we have used to annotate the utterances in Table 5.2. 10 are taken from He et al. (2018) and 4 are based on the actions taken by the users. The rule based acts are extracted using the code provided by them[6].

We provide the details about the 15 Negotiation Strategies (Zhou et al., 2019) and 21 Negotiation Strategies (Zhou et al., 2020) in Tables 5.3 and 5.4.

High level Negotiation Rules	Sub Strategy	Example	Detector
Focus on interests, not positions	Describe Product	The car has leather seats	classifier
	Rephrase product	45k miles $\longrightarrow$ less than 50k miles	classifier
	Embellish product	a luxury car with attractive leather seats	classifier
	Address concerns	I've just taken it to maintenance	classifier
	Communicate interests	I'd like to sell it asap.	classifier
Invent options for mutual gain	Propose Price	How about 9k?	classifier
	Do not propose first	n/a	rule
	Negotiate side offers	I can deliver it for you	rule
	Hedge	I **could** come down a bit	rule
Build Trust	Communicate Politely	Greetings, gratitude, apology, please	rule
	Build rapport	My kid really liked this bike, but he outgrew it	rule
	Talk informally	Absolutely, ask away!	rule
Insist on your position	Show dominance	The absolute highest I can do is 640	rule
	Negative Sentiment	Sadly, I simply cannot go under 500	rule
	Certainty words	It has **always** had a screen protector	rule

Table 5.3: The details of 15 Negotiation Strategies proposed by Zhou et al. (2019).

[6]https://github.com/stanfordnlp/cocoa/

Negotiation Strategies	Train set frequency
first_person_singular_count	26,121
pos_sentiment	24,862
number_of_diff_dic_pos	18,610
third_person_singular	17,000
hedge_count	12,227
number_of_diff_dic_neg	10,402
personal_concern	9,135
propose	8,449
politeness_greet	6,639
assertive_count	4,437
neg_sentiment	3,680
factive_count	3,429
politeness_gratitude	3,171
first_person_plural_count	2,876
liwc_certainty	2,530
liwc_informal	2,396
third_person_plural	1,721
trade_in	883
politeness_please	372
family	201
friend	149
<start>	5,383

Table 5.4: The details of 21 Negotiation Strategies (<start> added by us) used by Zhou et al. (2020). These are used to operationalize the 15 strategies using a rule based system (`https://github.com/zhouyiheng11/augmenting-non-collabrative-dialog/`). The frequency statistics on the train set (5383 conversations) is given. A detailed description regarding the rules used by prior work to extract these are out of scope of this work, however, we intend to provide the code and extracted strategies, along with the rule based mapping to the 15 strategies upon acceptance of this work.

Training Setup: We use most of the hyperparameters from Zhou et al. (2020). Each training run took at most 3 hours on a single Nvidia GeForce GTX 1080Ti GPU and all the models were saved based on Strategy Macro F1 performance. For experiments for Table 5.5 and 5.6 we saved the best models on best Strategy Macro F1 performance (HED being saved on outcome class prediction). This is because we wanted to prioritize and optimize our final model to capture sequence-structural information owing to our focus on interpretability. While performing ablation studies for Table 5.7, not all models have structure encoders, and hence for a fair comparison we chose a metric independent of the different modules for all the models in ablations. We use the negotiation outcome class prediction (RC-Acc) scores as that optimizes the dialogue for good negotiation outcome, which indirectly helps train the model to capture the sequence of strategies.

Baselines: **DIALOGRAPH** refers to our proposed method. To corroborate the efficacy of DIALOGRAPH, we compare it against our implementation of the present state-of-the-art model for the negotiation task: FST-enhanced hierarchical encoder-decoder model (**FeHED**) (Zhou et al., 2020) which utilizes FSTs for encoding sequences of strategies and dialogue acts.[7] We also conduct and ablation study, and evaluate the variants of DIALOGRAPH with different ways of encoding negotiation strategies, namely, **HED**, **HED+RNN**, and **HED+Transformer**. HED completely ignores the strategy and dialogue act information, whereas HED+RNN and HED+Transformer encode them using RNN and Transformers (Vaswani et al., 2017) respectively. While

[7] We replace the utterance encoder with BERT for fair comparison. This improved slightly the performance of the FeHED model compared to results published in Zhou et al. (2020).

HED+RNN is based on the dialogue manager of He et al. (2018), HED+Transformer has not been proposed earlier for this task. For a fair comparison, we use a pre-trained BERT (Devlin et al., 2019) model as the utterance encoder (§5.2.1) and a common utterance decoder (§5.2.4) in all the models, and only vary the Structure Encoders as described above. The strategies and dialogue acts in RNN and Transformer based encoders are fed as sequence of k-hot vectors.

Evaluation Metrics: For evaluating the performance on the next strategy prediction and the next dialogue act prediction task, we report the F1 and ROC AUC scores for all the models. For these metrics, macro scores tell us how well the model performs on less frequent strategies/dialogue acts and the micro performance tells us how good the model performs overall while taking the label imbalance into account. Strategy prediction is a multi-label prediction problem since each utterance can have multiple strategies. For the downstream tasks of utterance generation, we compare the models using BLEU score (Papineni et al., 2002) and BERTScore (Zhang et al., 2020b). Finally, we also evaluate on another downstream task of predicting the outcome of negotiation, using the ratio class prediction accuracy (RC-Acc) (1 out of 5 negotiation outcome classes, as described in §5.2.4). Predicting sale outcome provides better interpretability over the progression of a sale and potentially control to intervene when negotiation has a bad predicted outcome. Additionally, being able to predict the sale outcome with high accuracy shows that the model encodes the sequence of negotiation strategies well.

5.4 Results

We evaluate (1) strategy and dialogue act prediction (intrinsic evaluation), and (2) dialogue generation and negotiation outcome prediction (downstream evaluation). For all metrics, we perform bootstrapped statistical tests (Berg-Kirkpatrick et al., 2012; Koehn, 2004) and we bold the best results for a metric in all tables (several results are in bold if they have statistically insignificant differences).

Strategy and Dialogue Act Prediction: We compare DIALOGRAPH's effectiveness in encoding the explicit sequence of strategies and dialogue acts with the baselines, using the metrics described in §5.3. Table 5.5 shows that DIALOGRAPH performs on par with the Transformer based encoder in strategy prediction macro scores and outperforms it on other metrics. Moreover, both significantly outperform the FST-based based method, prior state-of-the-art. We hypothesize that lower gains for dialogue acts are due to the limited structural dependencies between them. Conversely, we validate that for negotiation strategies, RNNs are significantly worse than DIALOGRAPH. We also observe that higher macro scores show that DIALOGRAPH and Transformers are able to capture the sequences containing the less frequent strategies/dialogue acts as well. These results supports our hypothesis of the importance to encode the structure in a more expressive model. Moreover, DIALOGRAPH also provides interpretable structures which the other baselines do not. We will discuss these findings in §5.5.

Automatic Evaluation on Downstream tasks: In this section, we analyze the impact of DIALOGRAPH on the downstream task of Negotiation Dialogue based on the automatic evaluation metrics described in §5.3. In Table 5.6, we show that DIALOGRAPH helps improve the generation of dialogue response. Even though DIALOGRAPH attains higher BLEU scores, we note that single-reference BLEU assumes only one possible response while dialogue systems can have multiple possible responses to the same utterance. BERTScore alleviates this problem by scoring semantically similar responses equally high (Zhang et al., 2020b). We also find that both Transformer and DIALOGRAPH have a comparable performance for negotiation outcome prediction, which is significantly better than the previously published baselines (FeHED and HED+RNN). A

	Negotiation Strategies						Dialogue Acts				
	F1			ROC AUC			F1			ROC AUC	
Model	Macro	Micro	Weighted	Macro	Micro	Weighted	Macro	Micro	Weighed	Macro	Weighed
FeHED	17.6	25.6	36.3	55.8	61.7	54.7	20.6	37.4	30.6	76.9	79.2
HED+RNN	23.2	26.7	42.4	65.3	65.3	60.4	33.0	**46.2**	42.8	83.1	84.2
HED+Transformer	**26.3**	32.1	**43.3**	**68.2**	71.8	**61.8**	32.5	44.6	42.0	**85.6**	85.1
DIALOGRAPH	**26.1**	**34.1**	**43.5**	**68.1**	**73.0**	**61.8**	**33.4**	**45.8**	**43.7**	**85.6**	**85.4**

Table 5.5: Performance of the next strategy and dialogue-act prediction of various models. We report the F1 and ROC AUC scores. Significance tests were performed as described in §5.4 and the best results (along with all statistically insignificant values) are bolded.

	Generation				Outcome Prediction
		BERTScore			
Model	BLEU	Precision	Recall	F1	RC-Acc
HED	20.9	21.8	22.3	22.1	35.2
FeHED	23.7	27.1	26.8	27.0	42.3
HED+RNN	22.5	22.9	22.7	22.8	47.9
HED+Transformer	**24.4**	27.4	**28.1**	27.7	**53.7**
DIALOGRAPH	**24.7**	**27.8**	**28.3**	**28.1**	**53.1**

Table 5.6: Downstream evaluation of negotiation dialogue generation and negotiation outcome prediction. The best results (along with all statistically insignificant values to those) are bolded.

Model	BERT Score F1
DIALOGRAPH	**27.4**
w/o Strategy (ST)	26.8
w/o ST, Dialogue Acts (DA)	26.3
w/o ST, DA, BERT	22.7

Table 5.7: DIALOGRAPH ablation analysis. This shows that all the different components provide complementary benefits. We also evaluate without BERT for comparison with previously published works.

higher performance on this metric demonstrates that our model is able to encode the strategy sequence better and consequently predict the negotiation outcome more accurately. Additionally, ablation results in Table 5.7 show that both strategy and dialogue act information helps DIALOGRAPH in improving dialogue response. The difference in BERTScore F1 scores in Tables 5.6 and 5.7 arises due to different metrics chosen for early stopping. More details in Appendix 1.3.2.

Although, both HED+Transformer and DIALOGRAPH are based on attention mechanisms, DIALOGRAPH has the added advantage of having structural attention which helps encode the pragmatic structure of negotiation dialogues which in turn provides an interpretable interface. The components in our graph based encoder such as the GAT and ASAP layer provide strategy influence and cluster association information which is useful to understand and control negotiation systems. This is described in more detail in §5.5. Though transformers have self attention, the architecture is limited and doesn't model the structure/dependence between strategies providing only limited understanding. Further, our results show that DIALOGRAPH maintains or improves performance over strong models like Transformer and has much more transparent interpretability. We later show that DIALOGRAPH performs significantly better than HED+Transformer in human evaluation.

Human Evaluation: Since automatic metrics only give us a partial view of the system, we complement our evaluation with detailed human evaluation. For that, we set up DIALOGRAPH and the baselines on Amazon Mechanical Turk (AMT) and asked workers to role-play the buyer and negotiate with a single bot. After their chat is over, we ask them to fill a survey to rate the dialogue on how persuasive (*My task partner was persuasive.*), coherent (*My task partner's responses were on topic and in accordance with the conversation history.*), natural (*My task partner was human-like.*) and understandable (*My task partner perfectly understood what I was typing.*) the bot was [8]. Prior research in entailment has shown that humans tend to get better

[8] We use the setup of https://github.com/stanfordnlp/cocoa/. Screenshots in Appendix 1.3.4.

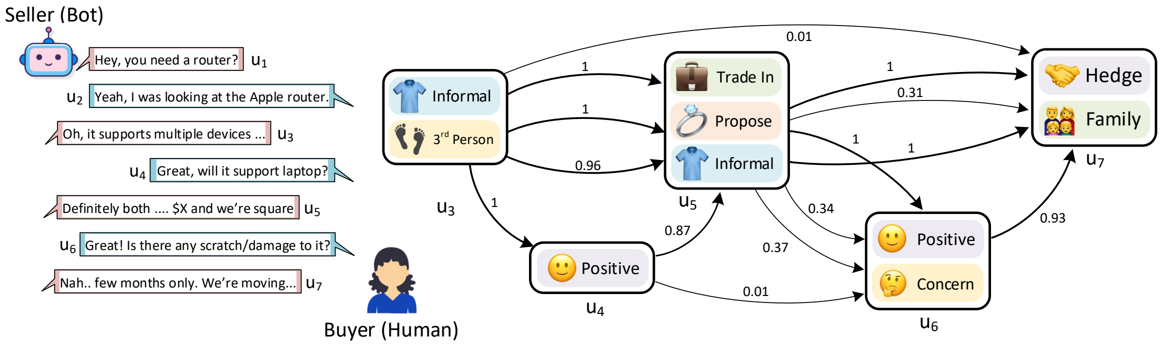

Figure 5.3: Visualization of the learnt latent strategy sequences in DIALOGRAPH where bolder edges represent higher influence. Here we present only a few edges for brevity and visualize min-max normalized attention values as edge weights to analyze the relative ranking of strategies. For example, for *family* at u_7, *informal* of u_5 has the most influence followed by *propose*. We present the full attention map for this example in Figure 5.4.

Model	Persuasive	Coherent	Natural	Understandable	Sale Price Ratio	Avg Turns	Avg words/turn
HED	2.50	2.50	**4.50**	2.50	-2.13	11.00	4.25
FeHED	3.30	3.75	3.70	**3.69**	0.25	14.30	5.76
HED+RNN	2.81	3.27	3.36	3.27	-3.68	13.90	3.61
HED+Transformer	3.50	3.50	3.70	3.40	-0.07	11.40	4.36
DIALOGRAPH	**3.58**	**3.94**	3.75	**3.70**	**0.49**	**15.72**	**5.84**

Table 5.8: Human evaluation ratings on a scale of 1-5 for various models. We also provide the average sale price ratio (§5.2.4). Negative ratio means that average sale price was lower than the buyer's target.

as they chat (Mizukami et al., 2016; Beňuš et al., 2011) and so we restrict one user to chat with just one of the bots. We further prune conversations which were incomplete potentially due to dropped connections. Finally, we manually inspect the conversations extracted from AMT to extract the agreed sale price and remove conversations that were not trying to negotiate at all.

The results of human evaluations of the resulting 90 dialogues (about 20 per model) are presented in Table 5.8. We find that baselines are more likely to accept unfair offers and apply inappropriate strategies. Additionally, DIALOGRAPH bot attained a significantly higher Sale Price Ratio, which is the outcome of negotiation, showing that effectively modeling strategy sequences leads to more effective negotiation systems. Our model also had a higher average total number of turns and words-per-turn (for just the bots) compared to all baselines, signifying engagement. It was also more persuasive and coherent while being more understandable to the user. From qualitative inspection we observe that the HED model generates utterances that are shorter and less coherent. They are natural responses like "*Yes it is*", but generic and contextually irrelevant. We hypothesize that this is due to the HED model not being optimized to encode the sequence of negotiation strategies and dialogue acts. We believe that this is the reason for the high natural score for HED. From manual inspection we see that HED is not able to produce very persuasive responses. We provide an example of a dialogue in Appendix 1.3.3. We see that although HED+Transformer model performs well, DIALOGRAPH achieves a better sale price outcome as it tries to repeatedly offer deals to negotiate the price. We see that the HED is unable to understand the user responses well and tends to repeat itself. Both the FeHED and HED baselines tend to agree with the buyer's proposal more readily whereas HED+Transformers and DIALOGRAPH provide counter offers and trade-ins to persuade the user.

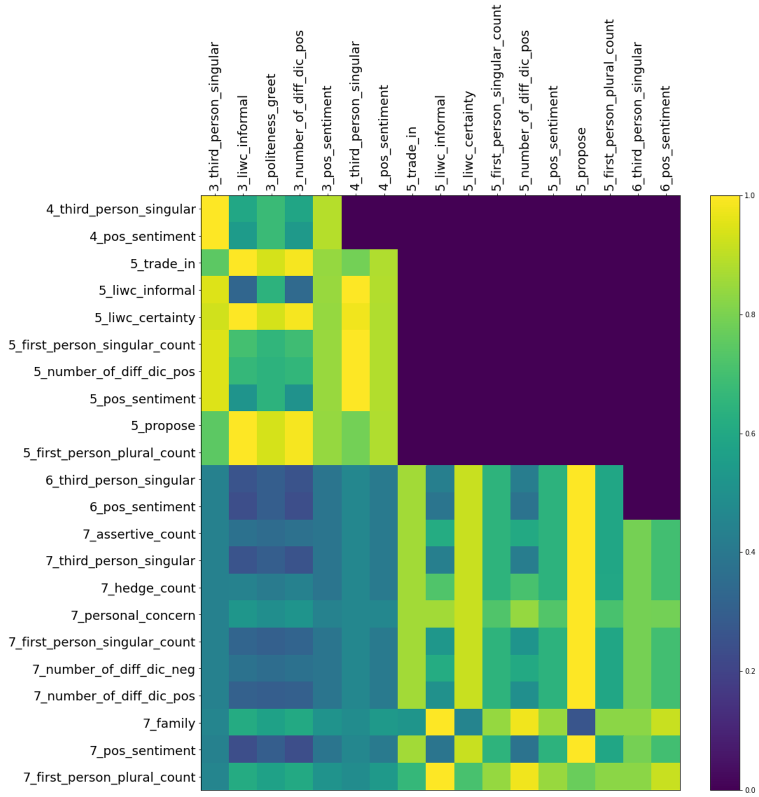

Figure 5.4: Visualization of the attention map learned by DIALOGRAPH for the example depicted in Figure 5.3 in the main paper. We only show it for a few turns for brevity. Here the axis labels represent the turn and the strategy.

Negotiation Strategy	Least associative strategies	Highly associative strategies
concern	certainty (0.1759), trade in (0.228)	politeness please (0.7072), politeness gratitude (0.5859)
hedge	trade in (0.4367), pos sentiment (0.4501)	propose (0.5427) friend (0.6218)
propose	factive count (0.3878), family (0.416)	politeness gratitude (0.5048), trade in (0.5223)
negative sentiment	trade in (0.3089), informal (0.3644)	family (0.6363), propose (0.6495)

Table 5.9: Examples of strategies and their least / highly associated strategies based on association scores extracted using the cluster attention scores given by the ASAP layer.

5.5 Interpreting Learned Strategy Graphs

We visualize the intermediate attention scores generated by the GATs while obtaining the strategy node representations. These attention scores tell us what strategies influenced the representation of a particular strategy and can be used to observe the *dependence* between strategies (cf. Xie and Lu, 2019; Norcliffe-Brown et al., 2018). We show an example in Figure 5.3 where for brevity, we present a subset of few turns and only the top few most relevant edges in the figure. For visualization, we re-scale the attention values for all incoming edges of a node (strategy) using min-max normalization. This is done because the range of raw attention values would differ based on the number of edges and this allows us to normalize any difference in scales and visualize the relative ranking of strategies (Yi et al., 2005; Chen and Liu, 2004). We notice that as soon as the first *propose* at u_5 happens, the strategies completely change and become independent of the strategies before the propose point. From Figure 5.3, we see that the edge weight from u_4 to u_6 is 0.01, signifying very low influence. We noticed this trend in other examples as well, wherein, the influence of strategies coming before the first propose turn to strategies coming after that, is very low. A similar phenomenon was also observed by Zhou et al. (2019) who study the conversations by splitting into two parts based on the first propose turn. Another interesting thing we note is that the *trade-in* and *propose* strategies at u_5 seem to be heavily influenced by *informal* from u_3. Similarly, the *informal* of u_5 was influenced by *positive sentiment* from u_4. This indicates that the seller was influenced by previous informal interactions to *propose* and *trade-in* at this turn, and that sellers tend to be more informal if the conversation partner is *positive*. In other examples, we see that at a particular utterance, different strategies depend on separate past strategies and also observe that the attention maps usually demonstrate the strategy switch as soon as the first *propose* happens, which is similar to what has been observed by prior work. These examples demonstrate that DIALOGRAPH can model fine-grain strategies, learn dependence beyond just utterances and give interpretable representations, which previous baselines, including the FSTs, lack. Specifically, each state of the FST is explicitly represented by an action distribution which can only be used to see the sequence of strategies and not observe associations or dependence information which DIALOGRAPH provides.

We utilize these cluster attention scores from the ASAP pooling layer to observe the *association* between various strategies which can help us observe strategies with similar contextual behaviour and structural co-occurrence. We take the average normalized value of the cluster attention scores between two strategies to obtain the association score between them. In Table 5.9, we show some examples of strategies and their obtained association scores. We observe that negative sentiment tends to be most associated to propose. We hypothesize that this is because that people who disagree more tend to get better deals. We observe that people do not tend to associate negative sentiment with trade-in, which is in-fact highly associated with positive sentiment, because people might want to remain positive while offering something. Similarly, people tend to give vague proposals by hedging, for instance, *I could go lower if you can pick it up*, than when suggesting trade-in. Concern also seems to be least associated with certainty, and most with politeness-based strategies. Thus, we observe that our model is able to provide meaningful insights which corroborate prior observations, justifying its ability to learn strategy associations well.

5.6 Related Work

Dialogue Systems: Goal-oriented dialogue systems have a long history in the NLP community. Broadly, goal-oriented dialogue can be categorized into *collaborative* and *non-collaborative* systems. The aim of agents in a collaborative setting is to achieve a common goal, such as travel and flight reservation (Wei et al., 2018) and information-seeking (Reddy et al., 2019). Recent years have seen a rise in non-collaborative goal-oriented dialogue systems such as persuasion (Wang et al., 2019; Dutt et al., 2020, 2021), negotiation (He et al., 2018; Lewis et al., 2017) and strategy games (Asher et al., 2016) due to the challenging yet interesting nature of the task. Prior work has also focused on decision-making games such as Settlers of Catan (Cuayáhuitl et al., 2015) which mainly involve decision-making skills rather than communication. Lewis et al. (2017) developed the DealOrNoDeal dataset in which agents had to reach a deal to split a set of items. Extensive work has been done on capturing the explicit semantic history in dialogue systems (Kumar et al., 2020; Vinyals and Le, 2015; Zhang et al., 2018a). Recent work has shown the advantage of modeling the dialogue history in the form of belief span (Lei et al., 2018) and state graphs (Bowden et al., 2017). He et al. (2018) proposed a bargaining scenario that can leverage semantic and strategic history. Zhou et al. (2020) used unsupervisedly learned FSTs to learn dialogue structure. This approach, however, although effective in explicitly incorporating pragmatic strategies, does not leverage the expressive power of neural networks. Our model, in contrast, combines the interpretablity of graph-based approaches and the expressively of neural networks, improving the performance and interpretability of negotiation agents.

Graph Neural Networks: The effectiveness of GNNs (Bruna et al., 2013; Defferrard et al., 2016; Kipf and Welling, 2017) has been corroborated in several NLP applications (Vashishth et al., 2019), including semantic role labeling (Marcheggiani and Titov, 2017), machine translation (Bastings et al., 2017), relation extraction (Vashishth et al., 2018), and knowledge graph embeddings (Schlichtkrull et al., 2018; Vashishth et al., 2020). Hierarchical graph pooling based structure encoders have been successful in encoding graphical structures (Zhang et al., 2019). We leverage the advances in GNNs and propose to use a graph-based explicit structure encoder to model negotiation strategies. Unlike HMM and FST based encoders, GNN-based encoders can be trained by optimizing the downstream loss and have superior expressive capabilities. Moreover, they provide better interpretability of the model as they can be interpreted based on observed explicit sequences (Tu et al., 2020; Norcliffe-Brown et al., 2018). In dialogue systems, graphs have been used to guide dialogue policy and response selection. However, they have been used to encode external knowledge (Tuan et al., 2019; Zhou et al., 2018) or speaker information (Ghosal et al., 2019), rather than compose dialogue strategies on-the-fly. Other works (Tang et al., 2019; Qin et al., 2020) focused on keyword prediction using RNN-based graphs. Our work is the first to incorporate GATs with hierarchical pooling, learning pragmatic dialogue strategies jointly with the end-to-end dialogue system. Unlike in prior work, our model leverages hybrid end-to-end and modularized architectures (Liang et al., 2020; Parvaneh et al., 2019) and can be plugged as explicit sequence encoder into other models.

5.7 Conclusions and Future Work

We present DIALOGRAPH, a novel modular negotiation dialogue system which models pragmatic negotiation strategies using Graph Attention Networks with hierarchical pooling and learns an explicit strategy graph jointly with the dialogue history. DIALOGRAPH outperforms strong baselines in downstream dialogue generation, while providing the capability to interpret and analyze the intermediate graph structures and the interactions between different strategies contextualized in the dialogue. However, more work is still required in negotiation study to understand negotiation strategies better and improve dialogue system performance.

As future work, we would like to extend our work to discover successful (e.g.: good for the seller) and unsuccessful strategy sequences using our interpretable graph structures. This would play an important role in strategy recommendations for human negotiations. Negotiation-coach systems can effectively use such knowledge to provide interpretable strategy recommendations for human negotiations. Our model which learns interpretable latent strategy sequences for better outcomes could directly impact such negotiation systems. Additionally, exploring the application of graph-based modules with large pretrained language models would enable generation of more coherent negotiation dialogue.

Part III

Language Structure for Controllable Dataset Construction

Chapter 6

Adversarial Data Generation for Error Correction by Leveraging Syntactic Structure

Transparency in models allow tracing and attributing incorrect model outputs to the source of the errors like issues with training data or limitations in model design. Training data at different stages of model development can lead to overfitting and spurious correlations due to several factors such as noisy data, confounding variables, and annotation errors (Wu et al., 2022b; Eisenstein, 2022; Han et al., 2020). Predominant efforts in training data (especially in pre-training) collection and annotation involve ad-hoc scraping (Rana, 2010; Gao et al., 2020; Hamborg et al., 2017), filtering and and blending (Longpre et al., 2023b; Jennings et al., 2023) which require manual trial and error to ensure high quality and desired features. As a consequence, there is a need for more controllable data design and collection strategies which can ensure high quality data leading to desired features in trained models.

In the final part of this thesis, I propose novel techniques for generating synthetic data that facilitate control over specific features during the training process. This chapter focuses on the task of factual error correction, and proposes an approach that uses infilling language models to modify constituent units in a sentence. Through this, I generate adversarial data that can be used for training purposes. The proposed approach enables controlled modification of a sentence to introduce diverse factual errors, producing challenging adversarial data. Research described in this chapter was conducted in collaboration with Hannaneh Hajishirzi, William Cohen and Yulia Tsvetkov and was presented in an EMNLP 2022 publication (Balachandran et al., 2022).

6.1 Introduction

While modern summarization models generate highly fluent summaries that appear realistic (Lewis et al., 2019; Zhang et al., 2020a), these models are prone to generating non-factual and sometimes entirely fabricated content (Cao et al., 2018a; Goodrich et al., 2019; Maynez et al., 2020). With the increasing adoption of language generation tools in user-facing products, such unreliability poses severe risks, including the spread of misinformation, panic and other potentially harmful effects (Ranade et al., 2021; Hutson et al., 2021).

Since it is difficult to control for factuality at training or inference time (Huang et al., 2021; Dreyer et al., 2021), a popular approach to fix the factual inconsistencies is via post-editing generated summaries (Cao et al., 2020; Dong et al., 2020). This allows summarization models to focus on fluency and content-relevance while improving factual consistency. However, there is no suitable data for training post-editing models to directly "translate" an incorrect summary to a correct one. Prior work constructed synthetic training data by introducing simple heuristic errors like replacing entities or numbers in reference summaries (Cao et al., 2020), but it is not clear whether such synthetic errors have sufficient coverage and accurately represent the types and distribution of actual errors made by language models. Further, with increasing language generation capabilities, models make more complex factual errors involving discourse structures and paraphrasing which cannot be easily captured with heuristics (Pagnoni et al., 2021). The goal of our work is to develop post-editing models that generalize over a wider range of factual errors (example in figure 6.1) in generated summaries from diverse summarization model types.

We propose FACTEDIT—a novel approach to post-editing text, to control for content factuality in generated summaries. Rather than manually defining a list of heuristic errors, it incorporates a new algorithm to generate adversarial (non-factual) examples using infilling language models (Donahue et al., 2020). We use lower ranked beam-search candidates from the language model as a source for potentially factually-incorrect summary facts, thereby producing a set of plausible, likely, and fluent, incorrect synthetic summaries for a particular correct reference summary. In this way, we leverage the capabilities of large language models to produce multiple candidates of alternative, erroneous summaries. These examples, along with factually correct references, are then used to train a sequence-to-sequence fact-correction model that aims at generating a factually consistent version of the candidate summary (§6.2).

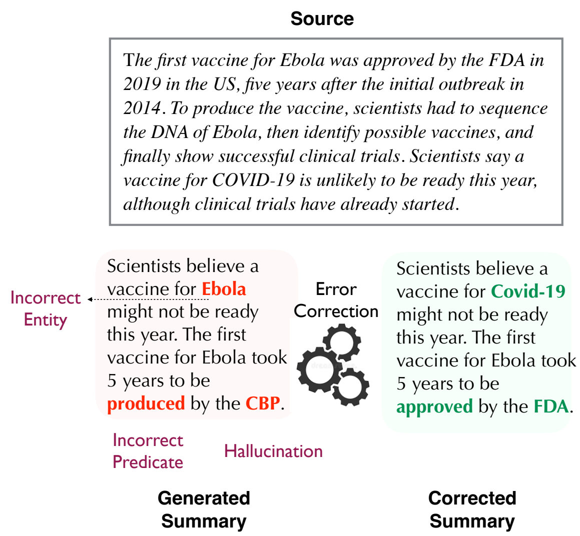

Figure 6.1: Model generated summaries often produce content which is factually inconsistent w.r.t. to the source. FACTEDIT rewrites these summaries by maintaining the abstractiveness but correcting factual errors.

We evaluate FACTEDIT on two datasets - CNN/DailyMail (Hermann et al., 2015) and XSum (Narayan et al., 2018) and across nine summarization models with the FRANK benchmark (Pagnoni et al., 2021) from Chapter 2 for evaluating various categories of factual errors in generated summaries (§6.3). The two summarization datasets represent varied distributions of factual errors in models trained on them and hence constitute a good test bed to evaluate the generalizability of our model. We show that FACTEDIT substantially improves factuality scores across two metrics - Ent-DAE (Goyal and Durrett, 2021) and FactCC (Kryscinski et al., 2020). On the Ent-DAE metric, FACTEDIT improves results by ∼11 points (CNN/DM) and ∼31 points (XSum), and on the FactCC metric we show improvements of ∼6 points (CNN/DM) and ∼24 (XSum) points on average across models (§6.4). Further, our analysis shows that FACTEDIT effectively corrects diverse error categories without the need for special heuristics or annotations (§6.5). An important application of FACTEDIT is to audit summarization systems and facilitate their reliability.[1]

[1]Code and data are available at https://github.com/vidhishanair/FactEdit

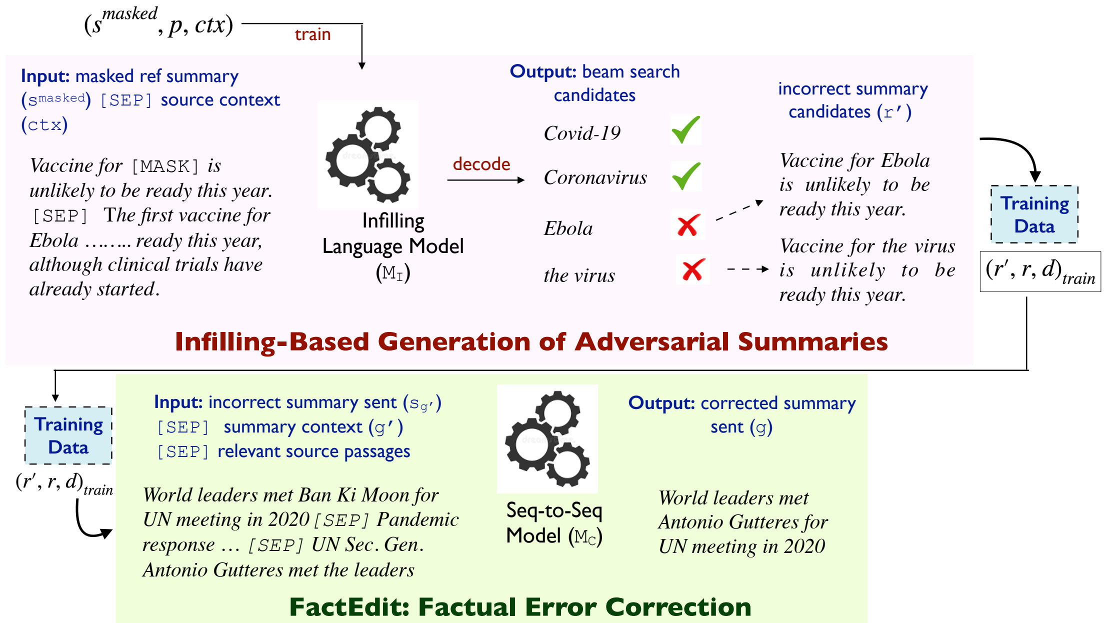

Figure 6.2: Architecture framework for FACTEDIT. Using masked versions of existing reference summaries, we use an infilling language model to produce alternative candidates for the mask position. We construct factually incorrect summaries by replacing the mask with the lower ranked candidates. Finally, we train a sequence-to-sequence model for fact correction using the synthetically constructed data.

6.2 Phrase-Centric Fact Correction - FACTEDIT

Assume a summarization model trained to process a document d and generate a coherent and fluent summary[2] g' which has been shown to often misrepresent facts from the document. FACTEDIT is a fact correction model M_C which takes the generated summary g' and document d, identifies factual errors and generates a rewritten summary g by correcting them (as outlined in figure 6.2).

We present an adversarial data generation approach which combines the power of pre-trained language models and control from fine-grained syntactic units to produce fluent and high-quality factually incorrect summaries. We train an infilling language model $\mathbf{M}_I$ using documents from summarization training data and use the model to introduce diverse factual errors in sentences from them (§6.2.1). Using the trained model, we introduce factual errors in reference summaries of the training data r producing an incorrect summary r' resulting in a synthetic dataset $\{r', r, d\}_{train}$ of erroneous summaries mapped to their corrected versions (pink section in figure 6.2). We train a sequence-to-sequence model $\mathbf{M}_C$ for factual error correction using the generated synthetic data (§6.2.2). Finally, we use the trained correction model to rewrite model generated summaries g' producing a corrected version g (§6.2.3 - green section in figure 6.2).

6.2.1 Infilling Data Generator M_I

Our data generation process leverages infilling language models (Donahue et al., 2020) to produce candidates to fill masked syntactic units (phrases) in a summary sentence. We mask parts of the input and use the infilling model to generate multiple candidates for the masked position. We then use lower order beam candidates as

[2]We denote incorrect input (to fact correction model) summaries using $'$ and corrected output (from fact correction model) without the $'$ throughout this paper. For E.g: g' is incorrect summary, r' is the incorrect reference summary while g is the corrected summary and r' is the corrected reference summary.

potential incorrect candidates to generate an incorrect version of the input. We hypothesize that, given the relevant context of a source document, a strong language model generates relevant and factual sequences at higher probabilities, compared to lower probability sequences. For the infilling model, we hypothesize that the lower ranked candidates are often alternative phrases of similar types (in case of entities) or parts-of-speech which are plausible but often not factually correct. Motivated by prior work (Goyal and Durrett, 2020) using lower ranked beam search candidates as a source for adversarial data, we use the lower ranked candidates to construct erroneous summaries from reference summaries.

Training: Our infilling model $\mathbf{M}_I$ is trained to take a masked sentence s^{masked} and its relevant context ctx as input and generate a correct phrase to fill in the masked span. To train $\mathbf{M}_I$, we construct a dataset using documents d from the training data of existing summarization datasets. For each sentence s in the first-k (k=5) positional sentences of a document d, we identify the subjects, objects and relations {sub, obj, rel} in them using OpenIE (Banko et al., 2007). By iteratively masking each phrase p in {sub,obj,rel}, we create a masked query s^{masked} and its corresponding context ctx by removing the masked sentence from the document, resulting in our training data $\{s^{masked}, p, ctx\}$, where p is the masked span text. We train a sequence-to-sequence model $\mathbf{M}_I$ on this data which takes s^{masked} `[SEP]` ctx as input and learns to generate p as the output. We intentionally use only sentences from the document as masked queries and do not use sentences from the reference summaries, to ensure that the model does not memorize phrases from the references. Thus, when applied to unseen reference sentences during inference, the model will produces richer beam search candidates.

Adversarial Data Generation: We use the trained infilling model to generate the synthetic dataset for fact correction using the document reference pairs $\{d, r\}_{train}$ from the summarization training data. For each sentence in the reference s_r, we use OpenIE to extract {sub, obj, rel} and iteratively mask one phrase at a time to construct masked sentences s^{masked} from the references. We provide this masked reference summary sentence and document d as input to the model and perform beam-search decoding for generation. We then consider lower ranked beam candidates (rank=[5,15])[3] as non-factual alternatives for the corresponding masked phrase. We then use these candidates as the replacements for the mask producing an erroneous summary r'. Running this on the $\{d, r\}_{train}$ training data, we construct a synthetic data $\{r', r, d\}_{train}$ of factually incorrect summaries paired with their correct version where r' and r differ by an incorrect phrase. To train the model to not perform any corrections on factual summaries, we keep original reference summaries for 20% of the data points ($r' = r$).

6.2.2 Fact Correction Model M_C

Using the parallel data $\{r', r, d\}_{train}$ produced by the above infilling method, we train models for factual error correction. In contrast to prior work (See et al., 2017) which used pointer based models to copy phrases from the source document, we use a sequence-to-sequence model – BART (Lewis et al., 2019) – to preserve the abstractive content in the input. The model $\mathbf{M}_C$ is trained with an erroneous reference summary sentence $s_{r'}$ produced by the infilling data generator and the corresponding document d as input and the correct reference summary sentence s_r as output. A straightforward option is to provide $s_{r'}, d$ concatenated as inputs to the model. But we hypothesize that providing the right context can help the model better correct the errors. Below

[3]We chose this range of ranks based on a manual analysis of 500 generated adversarial examples where our method produced factually incorrect replacements over 90% of the time.

we outline input structures that provide better context in the input:

Relevant Supporting Passages: To help the model better connect the relevant facts in the source document to the summary sentence being corrected, we experiment with providing only the most relevant parts of the document as input context instead of the entire document. Using a scoring function (ROUGE), we identify sentences from the document which have high overlap with the generated summary sentence and extract the top-k (k=3 for our work) such sentences. We provide these sentences along with a window of w_k (w_k=2) sentences before and after each as the input context to the model.

Surrounding Summary Context: While simple errors like incorrect entities can be detected and corrected with only the context of the current sentence being corrected, more complex discourse level errors like incorrect pronouns require the context of the rest of the sentences of the summary. To enable this, we additionally give the complete generated summary (other sentences from the summary) as additional context. For single sentence summaries like headline generation, this does not change the original setting, but for longer summaries this setting helps with discourse level errors.

In essence, our model $\mathbf{M}_C$ takes the input as *Incorrect Reference Sentence* ($s_{r'}$) `[SEP]` *Full Reference Summary* (r') `[SEP]` *Relevant Passages* and generates the corrected summary r as output.

6.2.3 Inference

Our trained fact correction model $\mathbf{M}_C$ can be directly applied to any model-generated summaries g', without access to the underlying model. For each sentence in a generated summary, we identify the relevant passages using ROUGE and provide it as an input to the model (in the form *Generated Summary Sentence* ($s_{g'}$) `[SEP]` *Generated Full Summary* (g') `[SEP]` *Relevant Passages*).

6.3 Experiments and Data

6.3.1 Datasets

We use two news summarization datasets CNN-DailyMail (Hermann et al., 2015) and XSum (Narayan et al., 2018). The two datasets have been extensively studied for the factual consistency in their generated summaries across a variety of models (Goodrich et al., 2019; Cao et al., 2018a). Reference summaries from CNN/DM are longer, having on average three sentences, and more extractive in nature. XSum on the other hand has shorter, single sentence summaries and is significantly more abstractive in nature. The summaries in these datasets are qualitatively different, and hence models trained on the two datasets present varied levels of challenges in maintaining factual consistencies.

Prior work have studied summaries generated from different language models and characterized the factual errors in them (Pagnoni et al., 2021). Generated summaries on the CNN/DM dataset are more extractive in nature and hence are more factual ($\sim$70% of summaries are factual) than the more abstractive generated summaries of XSum ($\sim$20% of summaries are factual). The longer summaries in CNN/DM display discourse level inconsistencies while summaries from XSum often hallucinate content which is not supported by the source document. Hence, the two datasets present a varied setting for evaluating the efficacy of our model across different kinds of errors. For our main evaluation, we evaluate the overall capability of FACTEDIT in correcting errors in summaries generated by a BART model.

We further evaluate our model on the FRANK benchmark (Pagnoni et al., 2021) which contains generated summaries obtained using multiple language models for both datasets annotated with human judgements on

Dataset	Method	R1	R2	RL	FactCC	Ent-DAE
CNN/DM	Bart (Lewis et al., 2019)	44.07	21.08	41.01	75.78	74.85
	Cao et al. (2020)	42.72	20.59	39.92	49.98	74.83
	FactEdit	42.17	20.22	39.37	75.49	**75.71**
	FactEdit + FactCC Filter (FF)	42.53	20.48	39.74	**76.03**	75.36
XSum	Bart (Lewis et al., 2019)	34.71	15.04	27.40	21.93	20.03
	Cao et al. (2020)	33.64	14.71	26.49	7.01	20.03
	FactEdit	33.58	14.68	26.71	**23.91**	**20.13**
	FactEdit + FactCC Filter (FF)	33.58	14.68	26.71	**23.91**	**20.13**

Table 6.1: FactEdit performance for correcting BART outputs (best performance in bold). FactEdit ourperforms factuality related baselines on FactCC and DAE scores, while maintaining competitive summarization quality.

their factuality and the category of factual error. As different language models have different distribution of factual error types, this evaluation helps us study the generalizability of FactEdit in correcting errors across them.[4] For the CNN/DM dataset, it contains model outputs from an LSTM Seq-to-Seq model (S2S) (Rush et al., 2015), a Pointer-Generator Network (PGN) model (See et al., 2017), a Bottom-Up Summarization (BUS) model (Gehrmann et al., 2018), a Bert based Extractive-Abstractive model (BertSum) (Liu and Lapata, 2019) and a jointly pretrained transformer based encoder-decoder model BART (Lewis et al., 2019). For the XSum dataset, it contains model outputs from a Topic-Aware CNN Model (Narayan et al., 2018), a Pointer-Generator Network (PGN) model, a randomly initialized (TransS2S) (Vaswani et al., 2017) and one initialized with Bert-Base (BertS2S) (Devlin et al., 2019).

6.3.2 Experimental Settings and Evaluation

Setup: We use OpenIE (Banko et al., 2007) to pre-process each summary and extract subject, object, predicate triples for each summary sentence. We use BART-base (Lewis et al., 2019) as our sequence-to-sequence model for the infilling based data generator and the fact correction model. Both models were trained with a batch size of 48, a learning rate of 3e-5, and warm-up of 1000 for 1 epoch. The maximum input sequence length was 512 and maximum output sequence length was 128. Using the infilling data generator, we generate 1233329 negative, 308332 positive examples for CNN/DM and 724304 negative, 181076 positive, examples for XSum as training data for fact correction. Models were trained on 4 Nvidia GeForce GTX TITAN X GPUs and each training run took ∼15 hours. All hyperparameters were chosen based on generated dev set ROUGE-L (Lin, 2004) on each dataset.

Evaluation Setup: Evaluating factual consistency of generated summaries is challenging, with relatively recent metrics developed to detect it. These metrics unfortunately do not correlate highly with human judgements yet. We therefore evaluate our model using two metrics - FactCC (Kryscinski et al., 2020) and Ent-DAE (Goyal and Durrett, 2021); each captures different error types. FactCC is a binary classifier, trained on a synthetic, heuristic error dataset, which is better at detecting simple semantic errors like incorrect entities or numbers. Ent-DAE is a classifier trained on synthetic data constructed using the dependency structure of the text. In addition to semantic errors, it is better at detecting more complex discourse-level errors (Pagnoni et al., 2021). We also report ROUGE (Lin, 2004) to evaluate if our model maintains the fluency of summaries. While ROUGE is less correlated with factuality (Pagnoni et al., 2021; Maynez et al., 2020), it helps evaluate if the

[4] As the benchmark has publicly available model outputs, the summaries across different datasets are from different models owing to their availability.

corrected summary is fluent and aligned with the reference summary. However, with factual corrections of outputs we expect small drops in ROUGE, since generation models were specifically optimized to maximize ROUGE presumably at the expense of factuality.

Our evaluation has two settings: i) FACTEDIT - correct all generated summaries in the test set and ii) FACTEDIT + FactCC Filter (FF) - using the FactCC metric we identify factually incorrect summaries, and only correct the incorrect ones.

Baselines: We compare our approach with Cao et al. (2020) as the baseline. The baseline uses a heuristic set of rules proposed by Kryscinski et al. (2020) to introduce simple errors (Entity, Number, Date, and Pronoun) in reference summaries and trains a BART-base model for error correction. Comparing our model with (Cao et al., 2020) helps us evaluate the benefit of our Infilling LM based adversarial data generator. [5]

6.4 Results

6.4.1 Factuality Results

We first evaluate FACTEDIT's ability to correct errors in summaries generated by a BART-base summarization model on the entire test set. We first generate summaries using a BART-base model finetuned on each dataset and then provide the generated summaries and their corresponding source documents as inputs to FACTEDIT for correction.

Table 6.1 shows results for this experiment. Our results show that correcting factual errors using our model improves the factuality results. The baseline model performs poorly with the FactCC metric showing lower scores than the BART model generated summaries, especially in the more abstractive XSum setting. The DAE metric for the baseline model is slightly lower than the BART model scores in the CNN/DM setting and has no improvement in the XSum setting showing that it does not perform corrections on complex errors. These results confirm our hypothesis that the baseline model trained on adversarial data based on heuristic errors does not transfer well to real errors in model generated summaries. In contrast, our model improves both metrics across both datasets. On the more challenging XSum dataset, our model has a $\sim$17 point improvement on FactCC and $\sim$0.1 improvement on DAE over the baseline model. The BART generated summaries on CNN/DM are $\sim$70% factual and hence using the FactCC Filter to correct only non-factual summaries helps improve results on FactCC. As XSum has more than 80% non-factual summaries, the FactCC filter does not change results and correcting all generated summaries is beneficial. In Table 1.13 we present examples of corrections made by FACTEDIT and present a discussion in §1.4.1.

Prior works have shown that improving factual consistency in summaries leads to a drop in ROUGE scores (Maynez et al., 2020; Cao and Wang, 2021; Cao et al., 2020). Our ROUGE results do not drop significantly and are consistent with prior work. These results show that our model does not significantly change the summaries and the corrected summaries contain the relevant information w.r.t. to the source.

6.4.2 Factuality Results across Model Types

Table 6.2 shows results of using FACTEDIT to correct summaries generated by different types of language models using the FRANK benchmark (Pagnoni et al., 2021). We provide the generated summaries collected in the benchmark along with their source document as input to our trained fact corrector. This setting evaluates the generalizability of our adversarial training data in handling different error distributions from

[5]While Dong et al. (2020) is also a factual error correction method, we were unable to reproduce it as no public code was available.

Method	RL	FactCC	Ent-DAE
CNN/DM			
Bart	41.53	46.29	72.57
FactEdit	37.73	42.29	78.86
FactEdit (FF)	37.73	**53.14**	**81.71**
BertSum	38.74	58.86	82.29
FactEdit	35.6	55.43	79.43
FactEdit (FF)	35.6	**61.71**	**82.86**
BUS	38.59	49.71	70.28
FactEdit	33.79	48.00	76.00
FactEdit (FF)	33.79	**56.57**	**80.00**
PointGen	35.62	**80.57**	**93.14**
FactEdit	32.54	75.43	90.29
FactEdit (FF)	32.54	78.29	90.86
Seq2Seq	27.15	19.43	29.71
FactEdit	24.78	23.43	48.00
FactEdit (FF)	24.78	**24.00**	**54.29**
XSum			
BertS2S	29.05	22.29	05.71
FactEdit	28.93	50.43	40.00
FactEdit (FF)	28.95	**50.43**	**40.00**
TConvS2S	25.69	17.71	04.00
FactEdit	25.64	47.16	29.14
FactEdit (FF)	25.64	**47.16**	**29.14**
PointGen	23.12	18.29	00.57
FactEdit	23.02	43.75	32.00
FactEdit (FF)	23.04	**43.75**	**32.00**
TranS2S	23.93	18.86	2.86
FactEdit	23.86	31.73	36.00
FactEdit (FF)	23.86	**31.73**	**36.00**

Table 6.2: Performance of FactEdit across different model generated summaries in the FRANK setting. Best performance is indicated in Bold. FactEdit model vastly improves factuality across multiple models on both FactCC and DAE scores.

Figure 6.3: Performance of FactEdit across different error categories in comparison to baseline (Cao et al., 2020). FactEdit improves the percentage of factual summaries across diverse types of factual errors.

different summarization models. Our results show our model significantly improves the factuality in generated summaries across 8 out of 9 test models. The FactCC Filter helps improves results in CNN/DM setting but does not change results in XSum similar to results in §6.4.1. In the more extractive CNN/DM setting, fact correction improves FactCC scores by ∼5.3 points and DAE scores by ∼10.9 points on average across models. In the more challenging and abstractive XSum dataset, we improve FactCC scores by ∼24 points and DAE scores by ∼31 points on average. Our results show that our model trained using Infilling LM based adversarial data is able to generalize and correct errors in generated summaries across different model types. Further, the significant improvement in XSum suggests that using LMs to generate factually incorrect candidates produces rich negative examples which help correct errors in more abstractive summaries.

Pretrained models like BART, BertSum and BertS2S have improved generation capabilities and make lesser mistakes in generating the right entity or predicate and more mistakes in discourse structuring (Pagnoni et al., 2021). FACTEDIT correspondingly shows larger improvements in DAE scores than FactCC scores in these pretrained models. The Pointer-Generator model being highly extractive in nature scores highly in factuality metrics in the CNN/DM setting and FACTEDIT reduces results in this setting showing that our model is not beneficial in copy-based model settings. On the other hand, in the XSum setting, the base Pointer-Generator model scores poorly and correcting factual errors in them improves factuality scores. Non-pretrained sequence-to-sequence models like Seq2Seq and TransSeq2Seq score poorly in both ROUGE and Factuality scores due to their limited language generation capabilities. By correcting factual errors in them, we improve factuality metrics significantly without changes in ROUGE, indicating that the gains are due to fact correction and not just rewriting the summary using a strong language model.

6.5 Analysis

6.5.1 Performance across Error Categories

The FRANK benchmark from Chapter 2 proposes a typology of three coarse categories of error types and collects human annotations on the error category: i) Semantic Frame Errors - This category covers factual errors in a sentence due to incorrect entity or predicate being generated ii) Discourse Errors - This covers discourse level factual errors like incorrect pronouns or sentence ordering iii) Content Verifiability Errors - This category is for errors whose factuality cannot be judged either due to grammatical errors or hallucinated content. We evaluate our model on its ability to correct different types of errors. We use the generated summaries from the best pretrained model in FRANK for each dataset - BART for CNN/DM and BertS2S for XSum. For each subset of summaries of a particular error type, we correct the summaries using FACTEDIT and report the percentage of factual summaries in the output as predicted by Ent-DAE. We compare FACTEDIT with the baseline to study whether our model improves error correction for each type.

From Figure 6.3, we see that across both datasets FACTEDIT increases the percentage of factual summaries across all three error categories, showing that the data generation process in FACTEDIT can generalize across multiple error types without the need for special heuristics or annotations. We see the largest improvements in the Semantic Frame Error category with an increase of ~ 8 points on CNN/DM and ~ 13 points on XSum. On the more complex Discourse Errors we see an improvement of ~ 5 points on both datasets. Finally, on Content Verifiability Errors, we see a ~ 8 point improvement on CNN/DM and ~ 2 point improvement on XSum. XSum has a high proportion of hallucination errors and our results highlight the challenge in correcting this error type.

6.5.2 Transferrability across Datasets

It is not always feasible to train specialized fact correction models for each dataset or style of summaries. While CNN/DM and XSum contain documents of the news domain, they both have different summary characteristics. Certain applications might benefit from a single model which can generalize to different summary styles. We evaluate the ability of FACTEDIT trained on CNN/DM data (FACTEDIT FF - CNN Model) to transfer and

Method	FactCC	Ent-DAE
BertS2S	22.29	05.71
FACTEDIT (FF) - CNN Model	**33.71**	**22.29**
TConvS2S	17.71	04.00
FACTEDIT (FF) - CNN Model	**30.29**	**22.29**
PointGen	18.29	00.57
FACTEDIT (FF) - CNN Model	**28.57**	**19.43**
TranS2S	18.86	2.86
FACTEDIT (FF) - CNN Model	**18.86**	**21.14**

Table 6.3: Transfer results of FACTEDIT. FACTEDIT trained using CNN/DM data transfers well to summaries generated for documents in XSum.

correct summaries generated for XSum documents using FRANK benchmark. Table 6.3 shows results for this experiment. Our results show significant improvement in factuality scores across all model types in this setting, showing that our data generation process produces rich and diverse factually incorrect examples which can generalize to factual errors in other data settings. By using only the source documents, our training data is agnostic of the styles, lengths and characteristics of reference summaries and hence is able to generalize to the headline style abstractive summaries of XSum.

6.5.3 Human Evaluation

To further study whether the factuality corrections performed by our model align with human expectations of automated summaries, we conduct a human study (Table 6.4). Two annotators evaluated 20 randomly sampled summaries generated from the test set of the XSum dataset using the BertS2S model and corrected by FACTEDIT and the baseline. The annotators were shown the entire source document and one corrected summary at a time and asked to rate the fluency and factuality of the summary on a 1-5 Likert scale. In manual evaluation, annotators rated FACTEDIT an average of 3.3 on factuality and 4.8 on fluency, compared to the baseline which was rated 3.1 and 4.6 scores respectively, showing that FACTEDIT improves on both factuality and fluency.

6.5.4 Ablation Study

Our model corrects each sentence in a summary given context of the rest of the summary and relevant passages in the source document. We ablate this setup by removing parts of the context one at a time. In Table 6.5 we present the results. We observe a a drop in results when using the entire summary as context (-RelevPass) and when removing the context of the summary in which the sentence occurs (-SummCtxt). Our results show the importance of having the appropriate context to enable the model to perform fact correction well.

6.6 Related Work

Factuality Evaluation Standard n-gram based metrics do not correlate well with human judgements of factuality and are unsuitable for evaluating factuality (Kryscinski et al., 2019; Fabbri et al., 2020). Several automated metrics were proposed to detect factual errors in generated summaries. They primarily fall in two paradigms—Entailment based and QA based metrics. Goodrich et al. (2019); Kryscinski et al. (2020); Maynez et al. (2020); Goyal and Durrett (2021) model factuality as an entailment verifying whether the summary is entailed by the source. Lee et al. (2022b) use similar masked infilling to generate training data for such metrics. QA models can be used to answer questions about the document, separately using the article and

Method	Fluency	Factuality
Cao et al. (2020)	4.58	3.10
FACTEDIT	4.75	3.33

Figure 6.4: Results of Human Evaluation on Fluency and Factuality of corrected summaries. Human judges rate summaries corrected by FACTEDIT higher in fluency and factuality than the baseline.

Method	FactCC	E-DAE
CNN/DM		
FACTEDIT	76.03	75.36
FACTEDIT -SummCtxt	75.73	74.23
FACTEDIT -SummCtxt-RelevPass	75.89	75.03
Xsum		
FACTEDIT	23.91	20.13
FACTEDIT -SummCtxt	22.89	20.06
FACTEDIT -SummCtxt-RelevPass	23.48	20.08

Figure 6.5: Results of Ablation study with components of fact correction pipeline removed. SummCtxt includes the generated summary as additional context. RelevPass includes relevant passages from the source as additional context. FACTEDIT setup ourperforms the ablated versions on FactCC and DAE scores.

the output summary as context and compare the answers to score the factuality of summaries (Durmus et al., 2020; Wang et al., 2020a). To evaluate these metrics, recent work collec human judgements for factuality (Fabbri et al., 2020; Maynez et al., 2020; Pagnoni et al., 2021). Additionally, (Pagnoni et al., 2021) also obtain annotations on factual error categories, which we use for our evaluations. This paper considers the problem of improving factuality, not measuring it. While this is a different task, it is related: e.g., measuring the number of corrections made by FACTEDIT might be useful as a factuality measure.

Improving Factuality of Summaries: There are two paradigms of work to ensure generated summaries are factually consistent: i) imposing factuality constraints during training or generation and ii) post-editing generated summaries to correct factual errors. Wan and Bansal (2022) add factuality contraints during pretraining by using factually consistent summaries. Model designs and factuality specific objectives help optimize for factuality during training (Gabriel et al., 2019; Cao and Wang, 2021; Dong et al., 2022; Rajagopal et al., 2022). During decoding beam search candidates can be ranked based on factuality measures (King et al., 2022; Zhao et al., 2020). Work on correcting factual errors post generation is relatively nascent. Cao et al. (2020) and Lee et al. (2022a) train fact correction models on synthetic data based on heuristic errors which we show is less effective than LM based error generation (Table 6.1). Dong et al. (2020) use a QA model to replace phrases in the summary with spans in the source text. This requires multiple inference iterations, making them very expensive for correction. In contrast our approach corrects errors in one iteration, making it a faster and more practical approach for error correction. Tangentially, work on correcting errors in reference summaries to make the training data more reliable has also been explored Adams et al. (2022); Wan and Bansal (2022). In dialog generation, Gupta et al. (2021) explore using mask-fill approaches to generate synthetic data for response ranking, showing that using language models to generate adversarial data might be applicable beyond summarization.

6.7 Conclusions and Future Work

We present an adversarial data generation process to generate rich synthetic data for a post editing model, which can be applied to correct factual errors generated summaries. Our data generation process leverages Infilling Language Models to produce alternative candidate summaries. Using the generated data, we train models to rewrite summaries by correcting factual errors in them. Through extensive experiments across two datasets and nine models, we show that our fact corrector model improves the factual consistency of the summaries, making them more reliable.

Future work can explore extending our work to design fact correction systems for open-ended generation. As error types are more complex for open-generation models as discussed in Chapter 3, our data generation process could be expanding to include broad sentence level error types as well. Further, while our methodology trains the model to make a single edit, correcting open-ended generations would potentially require multiple suggested edits as there could be multiple accurate versions of a text.

Ethical Considerations

Our model is trained to rewrite generated summaries by correcting factual errors in them. A limitation in our current setup is accurate detection of factual errors. We rely on off-the-shelf metrics for identifying summaries with factual errors to correct. Our model does not perform detection and correction together and often rewrites correct summaries as well if fed to the model. Therefore for settings like CNN/DM, it's beneficial to filter summaries using a factuality metric before giving summaries to our model as input. As our fact corrector is a sequence-to-sequence model, it could potentially introduce new factual errors in the summaries. It is essential to use factually detectors to ensure summaries are factual before real world usage of any corrected summary.

State-of-the-art language generation models, including summarization, are not yet powerful enough to facilitate fine-grained control over generated content. This leads to problems with content fidelity and safety; our work aims to ameliorate issues related to factual reliability of the models. However, existing approaches, including ours, cannot guarantee this yet. Furthermore, there is a risk of dual use, since the same techniques can be used to post-edit models to produce non-factual, harmful content to mislead, impersonate, or manipulate opinions. Future research should focus on developing better defenses methods against mis-using language generators maliciously.

Chapter 7

Pretraining Data Generation by Leveraging Syntactic Structure

Chapter 8 describes a method to synthetically generate a large training dataset for supervising a task. With large pretrained language models, we are moving to using large amounts data for pretraining models and a much smaller training data to adapt models for a specific task by finetuning or in-context learning. Here, models learn complex behaviour like learning language style and structure, knowledge or reasoning during pretraining and hence the data used for pretraining influences model behaviour significantly. A big concern though is that due to the scale of data used in pretraining, it is hard to control the kind of data being fed to models and this often results in models exhibiting unintended behaviour which it learnt from the pretraining data. In this chapter, I explore a new direction of synthetically generating data for pretraining models.

Here, I consider the task of factual error detection and proposes an approach for pretraining language models for improving factual knowledge representation. I leverage language syntax to convert structured knowledge to natural language sentences and use multiple novel approaches to pretrain language models and improve their ability to represent facts. Research described in this chapter was conducted in collaboration with Shangbin Feng, Yuyang Bai and Yulia Tsvetkov and was presented in an EMNLP 2023 publication (Feng et al., 2023a).

7.1 Introduction

Generating factually accurate document summaries in addition to fluent and informative ones is critical to the adoption of summarization models (Kryscinski et al., 2020; Goyal and Durrett, 2020). However, evaluating the factual consistency of summaries is still challenging, especially in specialized domains like scientific or legal (Cachola et al., 2020; Goldsack et al., 2022; Polsley et al., 2016; Kanapala et al., 2019). The key reason is that the majority of existing approaches employ neural classifiers trained on synthetic data constructed from a relatively small set of documents (Kryscinski et al., 2020; Goyal and Durrett, 2020). These factuality classifiers are thus not robust to ever-growing information, in which the distribution of entities, events, and their relations changes greatly across time and domains (Elsahar and Gallé, 2019; Laparra et al., 2020). Pagnoni et al. (2021) highlighted this limitation, finding that over 50% of factuality errors in the XSUM (Narayan et al., 2018) summarization dataset stem from *semantic frame errors*, namely entities, events, and relations between them, as illustrated in Figure 7.1.

To address these issues, we develop a new factuality evaluation model with improved factual knowledge representation, specifically focusing on entities and relations. Entity-oriented pretraining objectives have been shown to improve QA and reasoning tasks (Yasunaga et al., 2022; Liu et al., 2022b); we thus hypothesize that similar objectives can aid factuality evaluation in better detecting semantic frame errors in generated summaries.

We propose FACTKB, a novel factuality evaluation model built upon language models (LMs) augmented with factual knowledge (§7.2). The LMs are pretrained with knowledge-focused objectives using text synthesized from external knowledge bases (KBs) which store high-quality facts about entities and relations. We propose three types of complementary pretraining strategies: (1) **entity wiki**, with a focus on improving entity understanding; (2) **evidence extraction**, with a focus on incorporating supporting evidence from surrounding context; and (3) **knowledge walks**, with a focus on augmenting compositional reasoning about entities. For factuality evaluation, we first pretrain a language model using these three entity-centric pretraining strategies, and then fine-tune the enhanced LM on a factual error detection dataset.

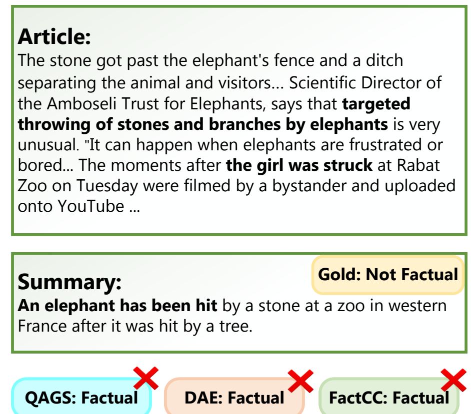

Figure 7.1: Existing factuality models struggle to identify *semantic frame errors* encompassing entities and relations. In the example, they fail to identify an error in the generated summary about *who* was hit by the stone.

We evaluate FACTKB's correlation with human factuality judgments across three settings (§7.3). In in-domain (news) summarization, FACTKB significantly outperforms baselines by 2–7 balanced accuracy (BACC) points on the FactCollect dataset (Ribeiro et al., 2022) and 10–12 correlation points on the FRANK benchmark (Pagnoni et al., 2021), particularly showing marked improvements in semantic frame errors. In out-of-domain experiments, FACTKB consistently outperforms existing approaches by 3–5 BACC points on three datasets in biomedical and scientific domains (Saakyan et al., 2021; Sarrouti et al., 2021; Wadden et al., 2020), demonstrating stronger generalizability to unseen documents in new domains. Further analysis shows that FACTKB is compatible with different LMs and KBs while presenting a lightweight and easy-to-use approach to factuality evaluation. [1]

7.2 Pretraining on Knowledge-Centric, Syntax based Synthetic Data - FACTKB

FACTKB aims to improve the robustness and generalizability of factuality evaluation by a simple *factuality pretraining*, which improves entity and relation representations in LMs. We first propose three pretraining strategies (§7.2.1). We then describe the training process to (1) pretrain an LM using the proposed strategies and (2) fine-tune the fact-enhanced LM on a factuality error detection dataset, resulting in FACTKB (§7.2.2). Figure 7.2 presents an overview of our approach.

[1] Code and data are available at https://github.com/BunsenFeng/FactKB.

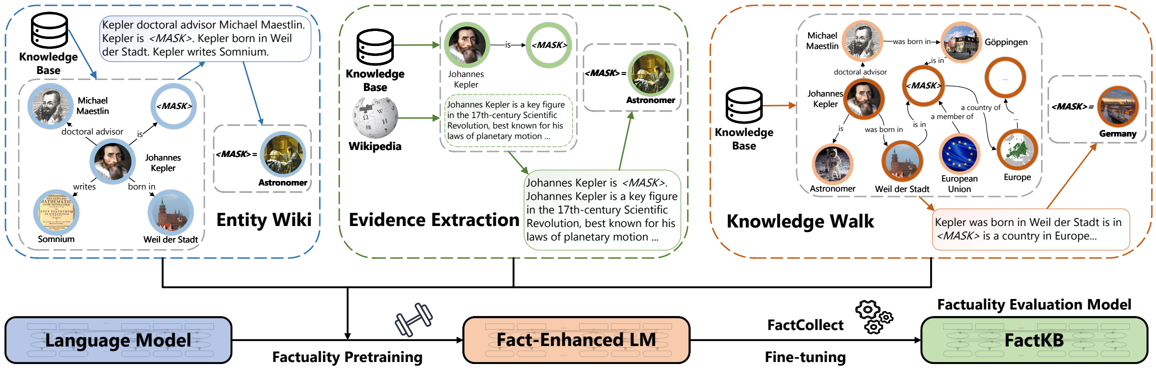

Figure 7.2: Overview of FACTKB. FACTKB pretrains LMs using three entity-centric pretraining strategies to improve fact representations. The objectives are designed to fill masked entities/relations in KB facts using i) *Entity Wiki* - direct facts about entities ii) *Evidence Extraction* - auxiliary knowledge about entities and iii) *Knowledge Walk* - compositional knowledge from the KB. The pretrained LMs are then fine-tuned for robust factuality evaluation.

7.2.1 Factuality Pretraining

Knowledge bases are rich reservoirs of facts about entities and relations (Vrandečić and Krötzsch, 2014; Pellissier Tanon et al., 2020), and we explore the possibility of leveraging external KBs as "fact teachers" to enhance an LM's representation of entities and relations.

Let $\mathrm{KB} = (\mathcal{E}, \mathcal{R}, \mathbf{A}, \epsilon, \varphi)$, where $\mathcal{E} = \{e_1, \ldots, e_{\mathcal{N}}\}$ represents the entities in the KB, $\mathcal{R} = \{r_1, \ldots, r_{\mathcal{M}}\}$ denotes the relations in the KB, $\mathbf{A}$ denotes the adjacency matrix where $a_{ij} = k$ indicates relation r_k connecting entities e_i and e_j $(e_i, r_k, e_j) \in \mathrm{KB}$, $\epsilon(\cdot) : \mathcal{E} \rightarrow \mathrm{str}$ and $\varphi(\cdot) : \mathcal{R} \rightarrow \mathrm{str}$ map the entities and relations to their textual names. We propose three novel types of factuality pretraining strategies that leverage the KB.

Strategy 1: Entity Wiki Entities in KBs often have multiple edges connecting them to other entities via relations, each representing a distinct but related fact about the entity. Inspired by the task of knowledge base completion (Bordes et al., 2013; Vashishth et al., 2020) to predict missing connections in KBs based on available KB facts, we propose the *entity wiki* factuality pretraining, where an LM is pretrained with the task of predicting masked entities or relations in KB facts. Specifically, for each entity $e_i \in \mathcal{E}$, we retrieve its one-hop neighborhood in the KB as $\mathcal{E}_{e_i} = \{e_j \mid \exists\, r_k \;\; s.t.\; a_{ij} = k\}$. Leveraging the syntactic structure of sentences, we then synthesize a new sentence using entity e_i and its connected one-hop facts:

$$\boldsymbol{d}_i = \mathrm{concat}_{e_j \in \mathcal{E}_{e_i}} \left[\epsilon(e_i) \varphi(r_k | a_{ij} = k) \epsilon(e_j) [\mathrm{SEP}] \right]$$

where concat denotes string concatenation and $[SEP]$ denotes the special token. Repeating this generation process for all $e \in \mathcal{E}$, we produce a corpus of entity facts as $\{\boldsymbol{d}_i\}_{i=1}^{|\mathcal{E}|}$ with the max size being the number of entities $|\mathcal{E}|$. We use this entity wiki corpus to pretrain an LM for better factual reasoning by randomly masking entities and relations in it and training the LM to predict the mask given the surrounding facts about an entity. We randomly mask the corpora with probability p and pretrain LMs with the masked language modeling objective. We expect this objective to train LMs to infer facts from surrounding knowledge and penalize unsupported hallucinations about entities and relations.

Factuality Pretraining	Corpus Size Bound	# Tokens	Example
ENTITY WIKI	$\propto |\mathcal{E}|$	5.4M	Johannes Kepler is born in Italy. Johannes Kepler is an [MASK]. [SEP] Johannes Kepler is the author of Astronomia nova. . . .
EVIDENCE EXTRACTION	$\propto ||A||_0$	12.2M	Hillary Clinton party affiliation [MASK] Hillary Diane Rodham Clinton is an American politician, . . . Member of the Democratic Party, she was the nominee . . .
KNOWLEDGE WALK	$\propto |\mathcal{E}|(\frac{||\mathcal{A}||_0}{|\mathcal{E}|})^k$	2.7M	University of Edinburgh located in Scotland located in [MASK] is a continent . . .

Table 7.1: Summary of the three factuality pretraining strategies.

Strategy 2: Evidence Extraction The goal of this pretraining strategy is to enhance the model's ability to evaluate facts based on relevant evidence. We begin by randomly selecting a triple $(e_i, r_k, e_j) \in \mathrm{KB}$ and use the first paragraph of the Wikipedia description of e_i as the auxiliary knowledge. We synthesize a new sentence using the two as:

$$\boldsymbol{d}_i = \epsilon(e_i)\ \varphi(r_k)\ [\mathrm{MASK}]\ \mathrm{Wikipedia}(e_i)$$

where we mask out $\epsilon(e_j)$ and $[\mathrm{MASK}]$ denotes the special token and $\mathrm{Wikipedia}(\cdot) : \mathcal{E} \to \mathrm{str}$ maps entities to the first paragraph of their Wikipedia description. Repeating this process N times with randomly selected triples, we obtain a corpus of triples paired with auxiliary knowledge $\{\boldsymbol{d}_i\}_{i=1}^{N}$. The corpus size is bounded by all KB triples represented as the L_0 norm of the adjacency matrix $||A||_0$. We use this corpus for the evidence extraction factuality pretraining and train the LM to predict the mask by using relevant evidence in the auxiliary paragraph. Through this, we aim to augment FACTKB's ability to implicitly select evidence from the document to support its factuality evaluation.

Strategy 3: Knowledge Walk Natural language documents often include compositional statements about entities and relations (Feldman and El-Yaniv, 2019; Wang and Pan, 2022), but pretrained LMs struggle with such compositional reasoning (Press et al., 2022). To improve FACTKB's ability to understand multi-hop claims, we propose the *knowledge walk* factuality pretraining strategy. Specifically, we randomly select a starting entity $e_{(0)}$ and randomly select an entity $e_{(1)}$ from its direct neighborhood $\mathcal{E}_{e_{(0)}}$, resulting in a one-hop triple $\{e_{(0)}, r_{(0,1)}, e_{(1)}\}$ where $r_{(0,1)}$ denotes the relation between $e_{(0)}$ and $e_{(1)}$. Now, from $e_{(1)}$, we randomly select an entity from it's direct neighborhood to take the next step. We repeat this process for $\mathcal{K}$ times, and obtain a $\mathcal{K}$-hop random walk of triples beginning at $e_{(0)}$: $\{e_{(0)},\ r_{(0,1)},\ e_{(1)},\ \cdots,\ r_{(\mathcal{K}-1,\ \mathcal{K})},\ e_{(\mathcal{K})}\}$. We then produce a sentence based on the $\mathcal{K}$-hop walk:

$$\boldsymbol{d}_i = \epsilon(e_{(0)})\ \mathrm{concat}_{i=0}^{\mathcal{K}-1}\ \left[\ \varphi(r_{(i,i+1)})\ \ \epsilon(e_{(i+1)})\ \right]$$

Repeating this $\mathcal{K}$-hop walk N times with different randomly selected starting entities, we obtain $\{\boldsymbol{d}_i\}_{i=1}^{N}$ as the corpus for the knowledge walk factuality pretraining, whose size is bounded by the number of all possible $\mathcal{K}$-hop walks as $|\mathcal{E}|(\frac{||\mathcal{A}||_0}{|\mathcal{E}|})^k$. In this corpus, we randomly mask entities or relations in each group of facts with probability p and train an LM to predict the masked element using the compositional facts around it using the masked language model objective. Through this pretraining, we expect FACTKB to improve in compositional fact understanding about entities and relations appearing in the summary and the input document.

We briefly summarize the three factuality pretraining strategies and provide examples in Table 7.1.

7.2.2 FACTKB Training

We initialize FACTKB with encoder-based LMs and pretrain FACTKB separately with each of the three factuality pretraining corpora using the masked language modeling objective to study the effectiveness of each strategy. This results in fact-enhanced LMs with the ability to better represent facts, entities, and relations. Finally, we fine-tune FACTKB on human-annotated factual error detection datasets with the sequence classification setting, taking SUMMARY [SEP] DOCUMENT as input and produce FACTUAL or NON-FACTUAL labels. The [CLS] token is adopted for classification. As a result, we obtain FACTKB, our entailment-based factuality evaluation model that classifies machine-generated summaries as factual or non-factual.

7.3 Data and Experiment Settings

7.3.1 Training

Data We use YAGO (Pellissier Tanon et al., 2020), an encyclopedic knowledge base based on Wikidata (Vrandečić and Krötzsch, 2014), to construct the three types of factuality pretraining corpora, while we discuss FACTKB's compatibility with different KBs in Section 7.5.2. For finetuning, we use the FactCollect dataset (Ribeiro et al., 2022), a dataset for factual error detection that gathers human annotations from different sources (Wang et al., 2020a; Kryscinski et al., 2020; Maynez et al., 2020; Pagnoni et al., 2021) and consolidates them into a single dataset. It mainly focuses on the news media domain, covering summaries and articles from CNN, Daily Mail, and BBC. FactCollect follows a binary classification setting where each (SUMMARY, ARTICLE) pair has a FACTUAL or NON-FACTUAL label. We present more details about the FactCollect dataset in Table 7.3.

Settings We use a ROBERTA-BASE (Liu et al., 2019a) checkpoint and continue pretraining separately on each of the three factuality pretraining corpora. We discuss FACTKB's compatibility with different LM initializations in Section §7.5.2. We assign corpus size parameter $N = 1e5$, masking probability $p = 0.15$, and knowledge walk length $\mathcal{K} = 5$ in the experiments, while we discuss the effect of corpus size and knowledge walk length in Appendix 7.5.3. We use a learning rate of $2e-5$ for pretraining, $1e-4$ for fine-tuning, a batch size of 32, and the RAdam optimizer. Pretraining is conducted for 5 epochs and fine-tuning has 50 maximum epochs with early stopping.

Hyperparameters: We propose to further pretrain LM checkpoints with three types of factuality pretraining and fine-tune on factuality evaluation datasets. We present hyperparameters for the pretraining and fine-tuning stage in Table 7.2. We mostly follow the hyperparameters in Gururangan et al. (2020) for the pretraining stage. The default hyperparameters on Huggingface Transformers are adopted if not included in Table 7.2.

Computational Resources: We used a GPU cluster with 16 NVIDIA A40 GPUs, 1988G memory, and 104 CPU cores for the experiments. Factuality pretraining with the default hyperparameters takes around 1.5 hours, while fine-tuning language models on the FactCollect dataset takes around 30 minutes.

Pretraining Stage		Fine-Tuning Stage	
Hyperparameter	**Value**	**Hyperparameter**	**Value**
LEARNING RATE	$2e$-5	LEARNING RATE	$1e$-4
WEIGHT DECAY	$1e$-5	WEIGHT DECAY	$1e$-5
MAX EPOCHS	5	MAX EPOCHS	50
BATCH SIZE	32	BATCH SIZE	32
OPTIMIZER	ADAM	OPTIMIZER	RADAM
ADAM EPSILON	$1e$-6		
ADAM BETA	0.9, 0.98		
WARMUP RATIO	0.06		
EVIDENCE: N	$1e5$		
WALK: N	$1e5$		
WALK: $\mathcal{K}$	5		

Table 7.2: Hyperparameter settings of FACTKB.

7.3.2 Evaluation

To study the robustness of FACTKB, we perform both in-domain and out-of-domain evaluation.

In-Domain Evaluation Since most research and resources on summarization and factuality are in the news media domain, we leverage the FactCollect dataset (Ribeiro et al., 2022) and the FRANK benchmark (Pagnoni et al., 2021) for in-domain factuality evaluation. We evaluate FACTKB on the held-out test set of the FactCollect dataset. FRANK (Pagnoni et al., 2021) is a factuality evaluation benchmark with human judgments on the factual consistency of model-generated summaries collected across 9 summarization models along with human annotations on the category of factual errors. Following the FRANK benchmark guidelines, we use two correlation measures (Pearson (Benesty et al., 2009) and Spearman (Myers and Sirois, 2004)). FRANK (Pagnoni et al., 2021) does not explicitly have binary labels such as {FACTUAL, NOT FACTUAL}. It also does not have a training set due to its nature as an evaluation benchmark. Following previous work (Ribeiro et al., 2022), we train FACTKB on the FactCollect dataset without the FRANK subset for the FRANK evaluation.

Generalizable Factuality Evaluation Summarization systems are used in diverse domains in the real world, including but not limited to news media (Liu et al., 2022c; Eyal et al., 2019; Li et al., 2016a), social media (Syed et al., 2019; Kano et al., 2018; He et al., 2020), and scientific literature (Cachola et al., 2020; Lev et al., 2019). Consequently, factuality metrics should also provide reliable factuality scores in the face of shifting domains. To study this, we perform an out-of-domain evaluation using unseen documents and summaries from the scientific domain. To establish a test bed for generalizable factuality evaluation, we make use of three datasets in the scientific literature domain:

- **CovidFact** (Saakyan et al., 2021) collects claims from the r/COVID19 subreddit and verifies them against relevant scientific literature and Google search results, resulting in a binary classification setting that is similar to the FactCollect dataset.

- **HealthVer** (Sarrouti et al., 2021) consists of claims sourced from TREC-COVID (Voorhees et al., 2021) and verified against the CORD-19 (Wang et al., 2020b) corpus. While HealthVer originally follows a three-way classification setting (SUPPORT, REFUTE, NOT ENOUGH INFORMATION), we remove the examples in the "NOT ENOUGH INFORMATION" category to evaluate models as they are trained on the binary classification setting (factual, non-factual).

Dataset	# Datapoint	# Class	Class Distribution	Train/Dev/Test Split	Proposed In
FACTCOLLECT	9,567	2	4994 / 4573	8667 / 300 / 600	Ribeiro et al. (2022)
FRANK	2,246	/	/	0 / 671 / 1575	Pagnoni et al. (2021)
COVIDFACT	1,257	2	401 / 856	846 / 94 / 317	Saakyan et al. (2021)
HEALTHVER	4,447	$3 \rightarrow 2$	2,758 / 1,689	3,340 / 508 / 599	Sarrouti et al. (2021)
SCIFACT	773	$3 \rightarrow 2$	508 / 265	508 / 56 / 209	Wadden et al. (2020)

Table 7.3: Statistics of the datasets and benchmarks adopted in this work.

- **SciFact** (Wadden et al., 2020) includes claims sourced from citation sentences in biomedical literature and verified against the cited paper's abstract. While SciFact uses three-way classification that includes "NOT ENOUGH INFORMATION", we similarly remove them in this work.

We leverage the well-organized version of the three datasets in Wadden et al. (2022). HealthVer (Sarrouti et al., 2021) and SciFact (Wadden et al., 2020) originally had NOT ENOUGH INFORMATION labels, while we removed such examples in the out-of-domain factuality evaluation to ensure their compatibility with FactCollect. We train and validate FACTKB with the FactCollect dataset from the news domain and evaluate on the test set of these datasets for zero-shot transfer learning.

We present more details about the adopted datasets in Table 7.3.

Baseline Details We compare FACTKB with different types of existing factuality evaluation models and factuality evaluation measures trained on both synthetic data and human annotated data:

- **BERTScore** (Zhang et al., 2020b) is a general metric for text generation evaluation based on pretrained BERT (Devlin et al., 2019).
- **QAGS** (Wang et al., 2020a) is a QA-based factuality metric, asking questions about summaries and articles while examining whether the answers are consistent.
- **QUALS** (Nan et al., 2021) is a QA-based factuality metric that uses QAGen (Shakeri et al., 2020) to generate both questions and answers from the summary.
- **DAE** (Goyal and Durrett, 2020) leverages the dependency structure of the summary and article to design a factuality metric.
- **SummaC** (Laban et al., 2022) proposes to revisit and repurpose NLI models for detecting factual inconsistencies in text summarization.
- **FalseSum** (Utama et al., 2022) augments NLI training data with controllable text generation for better factuality evaluation.
- **FactCC** (Kryscinski et al., 2020) is an entailment-based factuality metric trained on synthetic data evaluating factuality with binary classification. **FactCC+** is a variant of FactCC providing explanations. **FactCC+** is an enhanced version trained with human-annotated data.
- **FactGraph** (Ribeiro et al., 2022) is an entailment-based factuality metric based on jointly analyzing the textual content and AMR graphs of the summary and article. **FactGraph-adapters** is an enhanced version with pretrained adapters for both the text and graph modules.

We follow the same train/dev/test dataset split and experiment settings so that the results are directly comparable.

Model	All Data		CNN/DM		XSUM	
	BACC	F1	BACC	F1	BACC	F1
QAGS	79.8	79.7	64.2	76.2	59.3	85.2
QUALS	78.3	78.5	60.8	76.2	57.5	82.2
RoBERTa	76.1	76.5	62.5	76.2	62.1	78.3
FalseSum	78.9	78.2	53.7	34.6	61.1	64.3
FalseSum+	84.2	83.7	64.2	77.1	67.4	82.1
SummaC	86.6	86.2	75.4	83.5	71.9	90.4
FactCC	76.0	76.3	69.0	77.8	55.9	73.9
FactCC+	83.9 ($\pm$0.4)	84.2 ($\pm$0.4)	68.0 ($\pm$1.0)	83.7 ($\pm$0.5)	58.3 ($\pm$2.2)	84.0 ($\pm$1.0)
FactGraph	86.3 ($\pm$1.3)	86.7 ($\pm$1.1)	73.0 ($\pm$2.3)	86.8 ($\pm$0.8)	68.6 ($\pm$2.3)	86.6 ($\pm$2.0)
FactGraph-adapters	87.6 ($\pm$0.7)	87.8 ($\pm$0.7)	76.0 ($\pm$2.8)	87.5 ($\pm$0.4)	69.9 ($\pm$2.3)	88.4 ($\pm$1.2)
FactKB-wiki	89.3 ($\pm$0.4)*	**89.5** ($\pm$0.5)*	77.3 ($\pm$0.3)*	**88.2** ($\pm$0.6)*	**77.3** ($\pm$1.3)*	**91.8** ($\pm$1.2)*
FactKB-evidence	**89.4** ($\pm$0.2)*	**89.5** ($\pm$0.3)*	77.7 ($\pm$1.4)*	87.9 ($\pm$0.7)	76.8 ($\pm$1.9)*	90.8 ($\pm$0.8)*
FactKB-walk	89.1 ($\pm$0.4)*	89.3 ($\pm$0.5)*	**78.3** ($\pm$1.2)*	87.7 ($\pm$0.4)	76.4 ($\pm$0.3)*	90.4 ($\pm$1.4)*

Table 7.4: Performance of FactKB on the FactCollect dataset. We report average performance and standard deviation across 5 random seeds. Best performance is shown in **bold**, while * indicates statistical significance. FactKB significantly outperforms existing factuality evaluation approaches on in-domain evaluation.

7.4 Results

In-Domain Results We evaluate FactKB and baselines on the FactCollect dataset using the entire held-out test data, the CNN/DM subset and the XSUM (BBC) subset, and report balanced accuracy scores and micro F1 scores. We run each method five times with different random seeds and report the average performance as well as the standard deviation and compute statistical significance using the student t-test. Specifically, the t-test calculator for 2 independent means [2] was adopted for the calculations.

Table 7.4 demonstrates that FactKB significantly (*) outperforms all baseline factuality evaluation methods by 3.8 BACC points on average across the three dataset settings. This demonstrates that the introduction of KBs and factuality pretraining is beneficial for factuality evaluation. Among the three factuality pretraining strategies, all of them outperform baseline models, suggesting that FactKB's general methodology is compatible with different types of KB utilization.

Human Correlation We evaluate FactKB and baselines on the FRANK benchmark to study how well FactKB correlates with human judgments. We use the official script [3] to report the Pearson (ρ) and Spearman (r) correlation and p-values. Results in Table 7.5 show that classification-based metrics (FactCC, FactGraph, and FactKB) generally outperform QA-based metrics (QAGS and QUALS). FactKB significantly advances the state-of-the-art on the FRANK benchmark, resulting in the improvement of 5-15 correlation points across multiple settings. Our results show that FactKB is highly correlated with human judgments, making it a practical approach for evaluating the factual consistency of generated news summaries.

Out-of-Domain Results We evaluate FactKB and existing factuality evaluation models on out-of-domain scientific literature datasets in a zero-shot manner. Results are presented in Table 7.6, which demonstrate that while existing factuality evaluation models previously achieve good performance in the in-domain setting, they exhibit severe performance drops on the three out-of-domain datasets, performing only slightly better than random factuality scores (Random). This suggests that existing approaches are not generalizable to other domains, limiting their applicability. On the contrary, FactKB significantly (*) outperforms existing factuality metrics by 4.1 BACC points on average across the three out-of-domain datasets. Our results suggest

[2] https://www.socscistatistics.com/tests/studentttest/default2.aspx
[3] https://github.com/artidoro/frank

Model	All Data				CNN/DM				XSUM			
	ρ	*p-val*	r	*p-val*	ρ	*p-val*	r	*p-val*	ρ	*p-val*	r	*p-val*
QAGS	.22	.00	.23	.00	.34	.00	.27	.00	.07	.05	.06	.09
QUALS	.22	.00	.19	.00	.31	.00	.27	.00	.14	.00	.07	.03
DAE	.17	.00	.20	.00	.27	.00	.22	.00	.03	.38	.33	.00
ROBERTA	.35	.00	.41	.00	.43	.00	.31	.00	.23	.00	.15	.00
FALSESUM	.05	.00	.04	.11	.07	.05	.07	.03	.04	.28	.04	.35
FALSESUM+	.22	.00	.26	.00	.27	.00	.33	.00	.24	.00	.27	.00
SUMMAC	.33	.00	.35	.00	.42	.00	.36	.00	.24	.00	.25	.00
FACTCC	.20	.00	.29	.00	.36	.00	.30	.00	.06	.07	.19	.00
FACTCC+	.32	.00	.38	.00	.40	.00	.28	.00	.24	.00	.16	.00
FACTGRAPH	.35	.00	.42	.00	.45	.00	.34	.00	.30	.00	**.49**	.00
FACTKB-WIKI	.46	.00	**.52**	.00	**.57**	.00	**.49**	.00	.29	.00	.39	.00
FACTKB-EVIDENCE	.43	.00	.49	.00	.53	.00	.45	.00	.31	.00	.37	.00
FACTKB-WALK	**.47**	.00	**.52**	.00	**.57**	.00	.45	.00	**.35**	.00	.36	.00

Table 7.5: Correlation of FACTKB with human judgments of factuality on the FRANK benchmark. Best performance is shown in **bold**. FACTKB has the highest correlation with human judgments across five of the six settings.

Model	CovidFact		HealthVer		SciFact	
	BACC	**F1**	**BACC**	**F1**	**BACC**	**F1**
RANDOM	52.7	41.3	46.8	53.0	49.0	57.5
FACTCC	52.3	49.2	51.8	51.9	42.7	45.9
FACTCC+	51.1	50.5	49.5	51.6	48.6	55.2
FACTGRAPH	57.6	53.5	55.1	24.3	61.0	42.2
FACTGRAPH-EDGE	50.6	48.4	50.6	53.5	56.7	68.2
FALSESUM	50.6	41.6	56.8	51.2	45.7	65.3
FALSESUM+	50.1	41.2	57.3	51.6	51.9	65.4
SUMMAC	57.6	53.4	52.5	41.2	59.8	46.9
ROBERTA	59.0 ($\pm$3.2)	46.4 ($\pm$4.3)	55.0 ($\pm$2.2)	50.0 ($\pm$3.9)	58.1 ($\pm$4.0)	71.3 ($\pm$3.5)
FACTKB-WIKI	**64.8** ($\pm$0.3)*	**54.4** ($\pm$0.7)*	**60.1** ($\pm$0.4)*	**71.6** ($\pm$2.9)*	62.9 ($\pm$0.4)*	72.3 ($\pm$1.1)*
FACTKB-EVIDENCE	63.9 ($\pm$0.6)*	53.3 ($\pm$1.7)	59.0 ($\pm$1.0)*	70.8 ($\pm$0.9)*	61.4 ($\pm$0.5)*	**74.1** ($\pm$1.6)*
FACTKB-WALK	63.7 ($\pm$1.0)*	53.1 ($\pm$1.6)	58.5 ($\pm$0.5)*	68.7 ($\pm$1.7)*	**63.1** ($\pm$1.1)*	67.6 ($\pm$4.1)

Table 7.6: Performance of FACTKB on out-of-domain scientific datasets. We report average performance and standard deviation across 5 random seeds. Best performance is shown in **bold**, while * indicates statistical significance. FACTKB exhibits better generalization to new domains across all three datasets.

that the factuality pretraining strategies enable FACTKB to better represent facts (entities and relations) in a new domain, making the factuality evaluation model more robust to shifting domains.

7.5 Analysis and Discussion

7.5.1 Where did FACTKB Improve?

To better understand FACTKB's improvement over existing approaches, we leverage the factual error typology in the FRANK benchmark (Pagnoni et al., 2021) and examine FACTKB's performance on the three error categories: semantic frame, discourse, and content verifiability errors. Using the official script in the FRANK benchmark, we remove each category of errors and report changes in correlation scores. Higher variation indicates a greater influence on a model's ability to handle a certain type of error. Figure 7.4 demonstrates that FACTKB is significantly better at identifying semantic frame errors, which focus on entities and relations. This indicates that our KB-based factuality pretraining strategies successfully result in a better understanding of the facts regarding entities and relations. FACTKB also has good performance in other categories, resulting

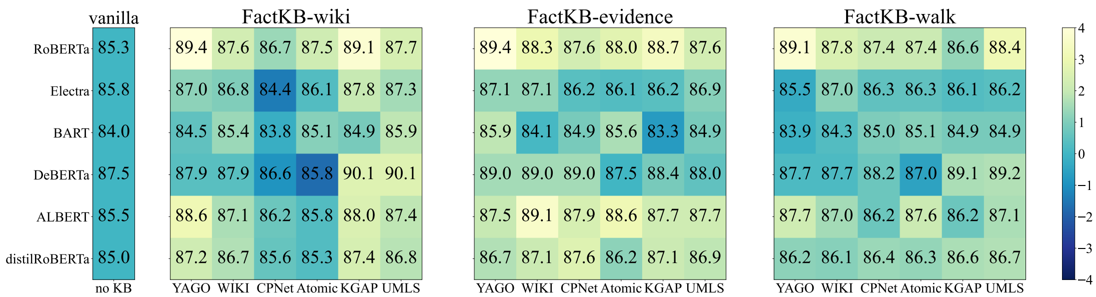

Figure 7.3: Compatibility of FACTKB across various LMs and KBs. We report BACC scores of different setups on the FactCollect dataset. FACTKB is a general method compatible with various LM and KB settings.

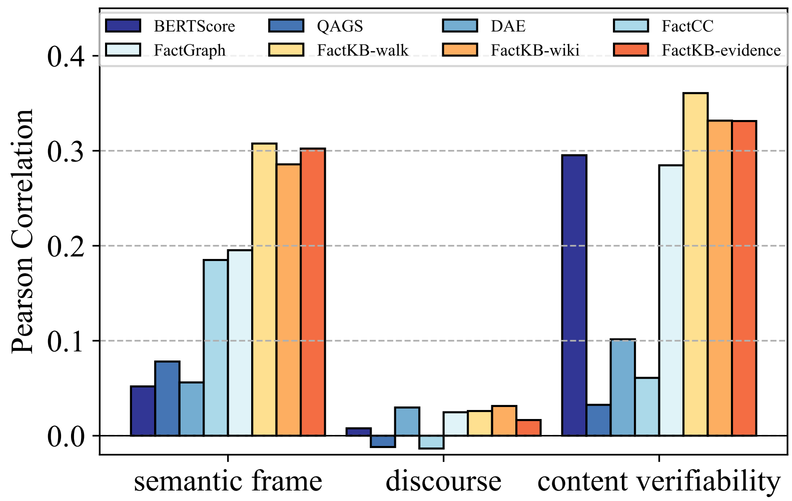

Figure 7.4: Correlation of FACTKB and baselines with human judgments across error categories. FACTKB shows significant improvement in capturing semantic frame errors and has slightly better or on-par performance on discourse and content verifiability errors.

in a factuality evaluation model that captures diverse types of errors and advances the state-of-the-art across the board. We conduct further qualitative analysis in Appendix 1.5.2.

7.5.2 KB and LM Compatibility

FACTKB uses pretrained LMs for initialization, leverages external KBs for factuality pretraining, and trains on factuality evaluation datasets to result in a factuality metric. Our general methodology to leverage knowledge bases as fact teachers for generalizable factuality evaluation could work with different LM and KB combinations. To study whether out approach works across different settings. For LMs, we used the ROBERTA-BASE, GOOGLE/ELECTRA-BASE-DISCRIMINATOR, FACEBOOK/BART-BASE, ALBERT-BASE-V2, MICROSOFT/DEBERTA-V3-BASE, DISTILROBERTA-BASE LM checkpoints on Huggingface Transformers. For the six KBs, we used their organized versions: YAGO15k at Lacroix et al. (2019), Wikidata5M at Wang et al. (2021), Atomic at West et al. (2022), ConceptNet at Zhang et al. (2022c), KGAP at Feng et al. (2021), and UMLS at Zhang et al. (2022c). For each combination, we initialize a particular LM and pretrain it using the proposed three pretraining strategies based on a particular KB. For each setting, we evaluate the resulting model using the FactCollect dataset and report the BACC scores. We present the performance of different

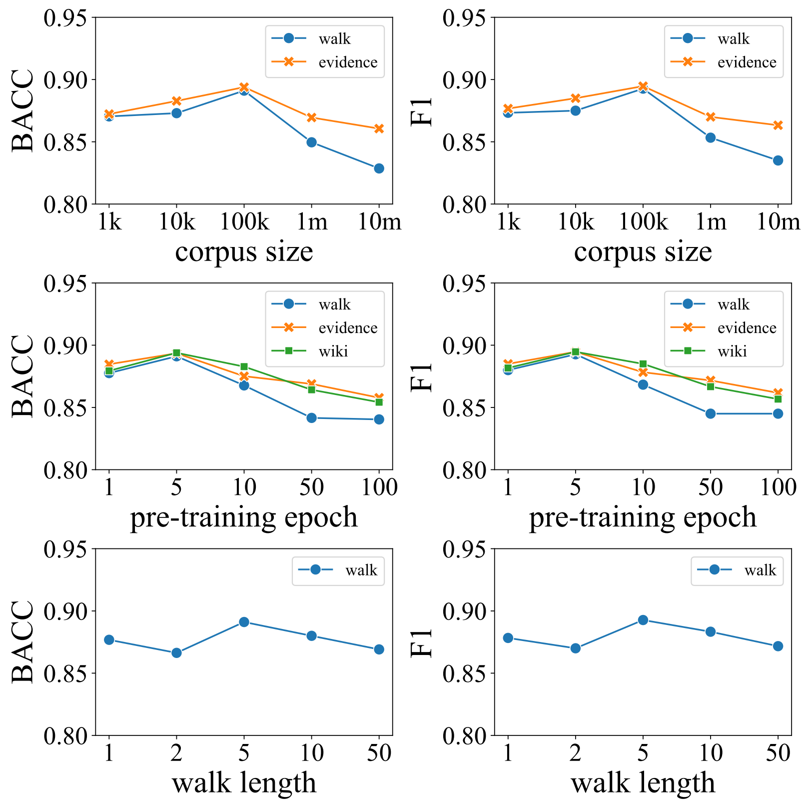

Figure 7.5: Parameter analysis of pretraining corpus size, epoch, and knowledge walk length.

settings in Figure 7.3, which illustrates that regardless of which LM and KB, FACTKB generally results in improved factual error detection capabilities compared to the vanilla LM checkpoints without factuality pretraining. In addition, certain LMs (RoBERTa and DeBERTa) and KBs (YAGO, KGAP, and UMLS) are better than others, suggesting that the choice of the base LM and external KB warrants further research. Our results demonstrate that FACTKB is a general pretraining approach that can be applied to various LM-KB combinations to improve fact representations and develop better factuality evaluation models.

7.5.3 Parameter Analysis

Corpus size. For evidence extraction and knowledge walk, the pretraining corpus size N is controllable and governs the amount of information towards augmenting FACTKB's ability towards factual errors regarding entities and relations. While we adopted $N = 1e5$ in the main experiments, we further explore the effect of factuality pretraining corpus size in Figure 7.5. It is illustrated that $N = 1e4$ or $N = 1e5$ are generally desirable settings, while factuality pretraining with too large Ns might be counterproductive. This could in part be attributed to catastrophic forgetting (Ramasesh et al., 2021), which warrants further research.

Pretraining epoch. FACTKB further pretrains LM checkpoints on the three factuality pretraining corpora, while the training epoch governs the intensity of such exercises. We adopted 5 epochs of continued pretraining in the main experiments, while we further explore the effect of pretraining epochs in Figure 7.5. it is demonstrated that 1 to 10 epochs are generally desirable while exercising too much might be counterproductive.

Metric	Pearson	Spearman	Usage Steps
QAGS	.22	.23	1) extract answer candidates 2) generate the questions 3) answer the generated questions 4) compare the answers to obtain the QAGS factuality score
QUALS	.22	.19	1) generating question and answer pairs from summaries 2) filter the generated question and answer for high-quality pairs 3) evaluate the generated question and answer pairs using the source document as input, compute QUALS scores for each summary
DAE	.17	.20	1) preprocess summaries and documents with dependency parsing 2) run the pretrained model to get DAE scores
FactCC	.20	.29	1) run the pretrained model to get FactCC scores
FactGraph	.35	.42	1) build abstract meaning representation graphs 2) run the pretrained model to get FactGraph scores
FactKB	**.47**	**.52**	1) run the pretrained model to get FactKB scores

Table 7.7: Usage steps of factuality metrics and their performance on the FRANK benchmark. FactKB (walk) presents a state-of-the-art factuality metric with minimum hassle when evaluating new summaries and articles.

Knowledge walk length. An important aspect of the knowledge walk factuality pretraining is the generated walk length $\mathcal{K}$, which governs the degree of compositionality in the pretraining corpus. While we adopted $\mathcal{K} = 5$ in the main experiments, we further explore the effect of $\mathcal{K}$ in Figure 7.5. It is illustrated that $\mathcal{K} = 5$ performs best by providing a moderate amount of compositionality in the factuality pretraining corpora.

7.5.4 Simplicity Study

While existing factuality evaluation approaches require additional processing (such as computing the dependency structure (Goyal and Durrett, 2020) and AMR graphs (Ribeiro et al., 2022) or running multiple iterations of question generation (Fabbri et al., 2022)) in the face of new data, FactKB requires no preprocessing and only uses a fine-tuned RoBERTa for sequence classification. We summarize the steps involved in using existing approaches and their performance on the FRANK benchmark in Table 7.7, which demonstrates that FactKB not only has state-of-the-art performance but is also a lightweight and simple factuality evaluation model.

7.6 Related Work

Factuality Evaluation Recent advances in text summarization have presented models and systems that are capable of generating increasingly fluent, controllable, and informative summaries of documents (Liu and Lapata, 2019; Balachandran et al., 2021; Meng et al., 2022; Tang et al., 2022; Goldsack et al., 2022; Peng et al., 2021; Aharoni et al., 2023; Liu et al., 2022d; Rothe et al., 2021; Narayan et al., 2021; Bhattacharjee et al., 2023; Chen et al., 2023c; He et al., 2023; Liu et al., 2023d; Chen et al., 2023b). However, they suffer from hallucination and might not be factually faithful towards the source document (Cao et al., 2018b; Pagnoni et al., 2021; Balachandran et al., 2022; Tang et al., 2023; Liu et al., 2023c; Luo et al., 2023), leading to increased research in factuality evaluation. QA-based approaches (Wang et al., 2020a; Nan et al., 2021; Scialom et al., 2021; Fabbri et al., 2022) attempt to generate and answer questions based on summaries and documents and judge the factuality by comparing answers. Later approaches are generally entailment-based (Kryscinski et al., 2020; Goyal and Durrett, 2020, 2021; Laban et al., 2022; Ribeiro et al., 2022), proposing to classify (summary, document) pairs into factual or non-factual labels. Among them, FactCC (Kryscinski et al., 2020) is one of the first entailment-based metrics and is trained on synthetic data; DAE (Goyal and

Durrett, 2020, 2021) proposes to leverage the dependency structure of summaries and documents; FactGraph (Ribeiro et al., 2022) builds abstract meaning representation graphs and adopts graph neural networks for joint representation learning along the textual content. In addition, hypothesis re-ranking (Garneau and Lamontagne, 2021), counterfactual estimation (Xie et al., 2021), NLI models (Utama et al., 2022), phrase-level localization (Takatsuka et al., 2022), and weighting facts in the source document (Xu et al., 2020) were also explored in factuality evaluation. Moving beyond a binary concept of factuality, FRANK (Pagnoni et al., 2021) promotes a fine-grained understanding of factuality and proposes a typology of factuality errors. Inspired by its analysis that *semantic frame errors*, errors regarding entities and relations, are a major source of factuality errors yet under-explored by existing factuality metrics, we propose FACTKB to leverage external KBs for factuality pretraining and help enforce better factuality towards entities and relations discussed in summaries and documents.

Knowledge Bases in NLP Knowledge base is a standard format for structured knowledge representation. One application of KBs in NLP is to inject knowledge and augment LMs, where different approaches focused aspects such as pretraining (Chen et al., 2020; Agarwal et al., 2021; Rosset et al., 2020; Li et al., 2022), document graphs (Hu et al., 2021; Zhang et al., 2022b), KB structure (Yasunaga et al., 2021; Zhang et al., 2022c), and long documents (Feng et al., 2023b). KB-enhanced approaches also advanced numerous NLP tasks, ranging from question answering (Mitra et al., 2022; Bosselut et al., 2021; Oguz et al., 2022; Feng et al., 2022; Heo et al., 2022; Ma et al., 2022), text generation (Rony et al., 2022; Dognin et al., 2021; Yu et al., 2021), and commonsense reasoning (Kim et al., 2022; Jung et al., 2022; Amayuelas et al., 2021; Liu et al., 2022a). In this work, we tap into KBs' nature as high-quality reservoirs of factual information and construct factuality pretraining objectives to augment factuality evaluation.

7.7 Conclusions and Future Work

We propose FACTKB, a simple and novel approach to factuality evaluation using language models pretrained on facts from external KBs to improve entity and relation representations. Specifically, we leverage KBs to construct three factuality pretraining objectives: entity wiki, evidence extraction, and knowledge walk. FACTKB pretrains an LM using the three objectives and fine-tunes the resulting model on factuality evaluation datasets. Extensive experiments demonstrate that FACTKB advances the state-of-the-art in both in-domain and out-of-domain factuality evaluation, better correlates with human factuality annotations, and better detects semantic frame errors. FACTKB presents an easy-to-use and generalizable factuality metric, facilitating research on factually-consistent summarization.

An important focus of this work is out-of-domain factuality evaluation: Summarization systems face input documents from varying domains, which requires factuality metrics to also generalize to different document domains. Existing metrics struggle with semantic frame errors and such struggle is exacerbated by the domain shift of entities and relations, while FACTKB offers a stronger and more generalizable factuality metric. However, in this work, we mainly focused on the additional domain of scientific literature, while other potential domains remain underexplored such as social media (Syed et al., 2019; Kano et al., 2018; He et al., 2020). We leave it to future work the exploration of FACTKB and existing factuality metrics on more document domains that are present in summarization systems. Further, in FACTKB we focus on constructing natural-language like statements by using surface forms of KB triples. An interesting direction for future work would be to construct more realistic natural sentences by leveraging language models. This would enable us to pretrain the model on more diverse and realistic statements while still controlling for data source and knowledge mix.

Ethical Considerations

While Section 7.5.2 offers empirical evidence that FACTKB is compatible with 6 language models and 6 external knowledge bases, it remains unclear upfront which LM and KB combination would be most desirable. While empirical performance could be a good guide, there are several unaddressed possibilities: For language models, it is possible to leverage an ensemble of FACTKBs seeded with different LM checkpoints and architectures. This might result in better factuality evaluation, but would also dramatically increase the computation costs when evaluating on new data. For knowledge bases, it is possible to leverage domain expertise and select an external knowledge base that would be most helpful for the domain adaptation of factuality evaluation. It is also possible to leverage a combination of existing knowledge bases for FACTKB's factuality pretraining, while the specific combination and how to apply different factuality pretraining to different KBs are hard to determine. All in all, FACTKB presents a general KB-enhanced factuality metric with numerous possibilities, while we leave some of these considerations to future work. FACTKB is initialized with pretrained language model checkpoints and leverages knowledge-base-based factuality pretraining. Consequently, FACTKB might pick up the biases of the adopted language models (Liang et al., 2021; Nadeem et al., 2021; Shaikh et al., 2023a; Tan and Celis, 2019) and knowledge bases (Fisher et al., 2020; Mehrabi et al., 2021). As a result, FACTKB might leverage these biases in judging the factuality of summaries, further reinforcing the bias in text summarization systems. We leave it to future work on understanding and mitigating the bias of factuality metrics.

FACTKB leverages high-quality and factual knowledge bases to generate factuality pretraining corpora and augment LM's ability to stay factual with respect to entities and relations discussed in the summary and document. On the contrary, if non-factual and misleading knowledge is leveraged for the three factuality pretraining strategies, it might jeopardize the factuality of FACTKB and make it insensitive to misinformation and falsehoods in summaries and documents. As a result, we encourage the responsible use of FACTKB and the factuality pretraining methodology.

Chapter 8

Conclusions

This thesis demonstrates the role of language structure in building trustworthy large-scale, data-driven language models. It present solutions that incorporate language structure in novel ways to design and evaluate transparent and reliable language generation systems. The first part of the thesis introduces semantically grounded analysis and evaluation measures to assess the factual reliability of trained language generation models. The second part of the thesis presents model designs that incorporate inter-sentence structures to promote inductive biases and transparency. Finally, the third part of the thesis presents techniques that use syntactic structures to generate synthetic, general datasets for training robust and factual systems. Overall, this thesis presents ideas and methods for incorporating trustworthiness and reliability in the design of each stage of ML pipelines for language model development.

8.1 Summary of Contributions

- I develop *comprehensive frameworks for fine-grained evaluation* of factual errors or hallucinations in language generation by leveraging semantic, discourse and pragmatic structure in language. I use this framework and evaluate a variety of models to quantitatively show the range of factual inaccuracies in model generated text, even with the current state-of-art large language models.
- I design a novel *interpretable by design* model architecture for document summarization which not only improves summary quality but also allows interpreting latent model decisions for summarizing.
- I design a novel *interpretable by design* model architecture for negotiation dialog which incorporates conversation and strategy structure resulting in improved negotiation capabilities as well as interpretation of latent negotiation strategies employed by the model.
- I introduce a new method for *high-quality synthetic training data generation* to fine-tune models for error correction showing improved error correction capabilities across diverse error types and models.
- I introduce a new method for *synthetic data generation for pretraining* language models on high-quality factual knowledge. I demonstrate that the resulting models have improved factual knowledge representation capabilities, resulting in improved factual error detection across diverse domains.

8.2 Future Work

8.2.1 Task-Agnostic Reliability

Recently, there has been increased interest in general NLP models that are pre-trained on large volumes data and useful for various downstream tasks. As a consequence, there is a need to improve reliability in such general models by addressing limitations in data, model and evaluation. While this thesis demonstrates this idea using different NLP tasks as use-cases, the core ideas presented are applicable beyond individual tasks to address language model reliability in a task-agnostic manner.

Dynamic, Fine-Grained Evaluations: Evaluation in large language model era is dominated by static benchmarks which provide an aggregate measure of a model's performance on a broad range of tasks. These benchmarks though useful for model comparisons lack any visibility into fine-grained model capabilities and do not provide any feedback on model limitations. As an initial step to address this, I developed fine-grained evaluation frameworks for factuality in language generation in Chapter 2 and Chapter 3. Though these frameworks are specific to factuality, their design protocols can be extended to develop fine-grained evaluations for complex settings like dialog modeling, mathematical reasoning, low-resource translation and to new capabilities emerging with increasing model complexities. For example, following the structure based taxonomy from Chapter 2, evaluation for mathematical reasoning can be decomposed into different hierarchical types like induction, geometric reasoning, algebraic reasoning and each reasoning type can be further evaluated via fine-grained categories which evaluate the correctness, coherence, notations or efficiency. Extending ideas presented in this thesis, language structure could also potentially help in decomposing and categorizing such complex evaluations. Based on the taxonomies designed, targeted evaluation data can be collecting using human annotators or by using language models to generate such data. Research presented in this thesis and other independent studies have shown that such decomposition and taxonomies can help human annotators provide high-quality annotations (Pagnoni et al., 2021; Krishna et al., 2023) as well as help models generate high-quality diverse data (Mishra et al., 2024; Wang et al., 2023a), demonstrating how fine-grained evaluation can be conducted while maintaining quality and efficacy (Yin et al., 2023). Such evaluation frameworks will enable a much deeper understanding of models and their limitations.

In an adjacent axis, another concern with current approaches towards evaluation is the increasing risk of benchmark contamination (Sainz et al., 2023; Yang et al., 2023) and overfitting (Zhou et al., 2023; Golchin and Surdeanu, 2023) due to models being trained on large, unfiltered internet data. Public test sets or data highly relevant to the test set can potentially be inadvertently included in pretraining corpora, leading models to pick up on spurious artifacts or shortcuts and resulting in unfair evaluations that are extremely hard to diagnose (Oren et al., 2023). In Chapter 6 and Chapter 7, I presented data generation methods for producing synthetic training and pretraining data. An interesting direction to explore would be to combine the ideas of synthetic data generation with fine-grained evaluation taxonomies to develop dynamic, evolving and comprehensive benchmarks. A fine-grained taxonomy for a task or problem would ideally enumerate all the aspects that need to be evaluated like correctness or coherence from the example above. Based on the taxonomy, we can devise a synthetic data generation method that would generate diverse evaluation candidates to targetedly evaluate each aspect of the taxonomy. Such a benchmark can be re-generated with different parameters every time the evaluation protocols change or new domains or control variables need to be added providing a highly dynamic way to evaluate model capabilities. Dynamic evaluations are becoming increasingly popular as static benchmarks are hard to update with constantly changing model capabilities (Gehrmann et al., 2021), but are critiqued for their lack of diversity and quality. A taxonomy grounded dynamic benchmark would enable us to produce a continually updating dynamic benchmark while still retaining control, diversity and quality.

Model Interpretability: The transformer architecture has been adopted widely for training and deploying language models recently. While there has been a lot of interest in improving the architecture to address the speed and efficiency concerns, the models are still black-box in nature and do not enable any form of interpretability or verifiability of their functioning. Probing and understanding how models represent knowledge, make latent decisions or how data influences model training are still open questions. In Chapter 4 and Chapter 5, I proposed architecture designs for a new kind of *interpretable-by-design* or *self-explaining* models that not only model tasks but also produce interpretations or provide a means to understand latent model decisions. While these initial ideas have been demonstrated with task-specific designs, future research to design general and interpretable models can enable big strides in improving the reliability of NLP systems, especially in high-stakes settings. Recent explorations of augmenting general LLMs with the ability to explain their decisions focus on eliciting such explanations via prompting like chain-of-thought (Wei et al., 2022), tree-of-thought (Yao et al., 2023) or analogical prompting (Anonymous, 2024). While effective in improving model performance, augmenting models to explain in such post-hoc ways during inference has also been shown to be biased or unfaithful (Shaikh et al., 2023b; Turpin et al., 2023). Therefore, expanding the ideas presented in this thesis and exploring ways to augment transparency and explanation generation *by design* during model design and development would be crucial to address such issues. Following work in this thesis, language structure can play a role in developing these models by either supporting the design of interpretable architectures or introducing explanation-based training paradigms.

Additionally, there is also the need to close the loop in interpretability research to effectively leverage the findings from model interpretability and understanding to address limitations and effectively improve models. Most of interpretability research in NLP is primarily focused of producing human understandable explanations (Ehsan and Riedl, 2020; Marasović et al., 2021; Ehsan et al., 2022) or understanding model behavior as a function of training dynamics (Koh and Liang, 2017; Grosse et al., 2023). But one of the most promising and sought after use of transparency and model explanations is identifying limitations and improving model development. In (Ahia et al., 2023) and (Balachandran et al., 2023a), we demonstrated that leveraging self-explaining classifiers to supervise model explanations can enable models to learn the training data better and results in improved performance and reduced reliance on spurious correlations. Future work could explore supervising model explanations or training with diverse explanations to mitigate concerns of spurious correlations, generation of unsafe or incorrect text or to even improve language understanding and generation across the board.

Synthetic Pretraining Data: The large performance improvements and impressive capabilities of large language models are often credited to the vast volumes of internet data used in pretraining. While this large data does lead to improved language understanding capabilities, they also often cause undersirable behaviours in models like generation of toxic text (Balachandran et al., 2023b), unsafe text (Wang et al., 2023d), inaccurate or outdated information (Mishra et al., 2024) or low performance for tail-end information or low-resource settings (Mallen et al., 2023). It is extremely challenging to process large pretraining corpora to separate useful data and problematic data. Current data processing methods are often ad-hoc and require manual inspection and heuristics (Longpre et al., 2023a; Computer, 2023). Further, even when subsets of good sources are identified it is challenging to ensure the optimal blend of data and the right curriculum to be provided to models. There is an increasing need to develop better strategies for controlling and ensuring good quality pretraining data for model training.

In Chapter 7, I presented initial ideas towards generating high-quality synthetic data for pretraining. We showed that having a controllable, synthetic data can improve factual knowledge representations resulting in improved performance on factuality related tasks. Expanding this idea beyond factuality and knowledge, a promising direction is to generate a broad but controllable synthetic data for general-purpose pretraining. Using existing sources of high-quality data like knowledge bases, books, news reports as a base, a pretrained language model can be employed to augment the dataset using various diversification techniques like paraphrasing (Okur et al., 2022), backtranslation (Li et al., 2023a), targeted editing (Balachandran et al., 2022; Mishra et al., 2024) or controllable generation (Liu et al., 2020). Extending ideas from Chapter 6, using language structure to ground the diversification process can additionally also allows us fine-grained control and flexibility in the process allowing us to control the data mixtures and varying different sources of data as required. While such synthetic generation measures is a good start, we will potentially need more extensive research on how to effectively and ethically use large public data to ensure that the language models we develop are safe and accountable.

8.2.2 Human-Centered Reliability

The recent rise in real-world use and deployments of LLMs has resulted in model generated text being directly shown to users, without significant research on the impact on them. Especially with models often generating incorrect and false information or hallucinating content, there's an increasing need to empower human users to understand, appropriately trust and ethically use model generated content from various ML/LLM tools. Predominant work on fact-checking, misinformation and detecting model generated text focus on assigning binary labels to text (true v/s false, model generated v/s human written) (Guo et al., 2022; Mitchell et al., 2023). While such classification enable building scalable safety measures, they are not sufficient for enabling humans to have agency over the content they use. Here, there is a need to bridge ML, NLP and HCI research to identify best ways to bring users into the model and application designs and provide them with the agency to understand the accuracy, reliability and provenance of generated content.

In Chapter 2 and Chapter 3, I explored initial ways to provide fine-grained reliability constructs for better transparency and evaluation. Such fine-grained error types provide an understanding of the reliability of specific spans of text and understanding what specific errors are present in the text. While this is the first step, it is still insufficient towards empowering humans to analyse and judge model generated text. Future work should explore more channels of information that can be provided to human consumers of model outputs like diverse provenance or evidence for each fact in the generated text, ability to control sources of evidence to include peronally trusted websites and sources of knowledge, model uncertainity measures and more. Further, producing these in independent strands of research will still make the reliability assessment process extremely overwhelming for users. Hence, studying how to bring them together and visualize them via custom tools in an accessible manner will bring enable any LLM tool user to assess the reliability of the model generated content they use and empower them to understand and appropriately use the content.

8.2.3 Reliability in Multimodal Era

This thesis entirely focuses on studying and mitigating concerns with reliability in language based models. But reliability is not a concern only in NLP. As large-scale pretrained models are being expanded in their input and output modalities, they are being exposed to diverse sources of knowledge and information and are producing more sophisticated generated content along multiple axes of modality. Today, the development, training and deployment of image generation, video generation and general-purpose large scale multimodal modals is being done in breakthrough speeds (Bavishi et al., 2023; Liu et al., 2023a; Team et al., 2023). Though the current models can generate realistic music, translate audio in various languages and produce high-quality videos, they also produce images with wrong representations of world knowledge, wrong translations and videos which do not accurately follow instructions (Li et al., 2023b; Liu et al., 2024; Yao and Wan, 2020). The issue of reliability in such multimodal large scale models is further amplified due to the computational challenges in representing and conducting detailed evaluations to even identify concerns of factual inaccuracy, harmful biases or spurious correlations. There has been initial research in studying these concerns in context of multimodal research (Sun et al., 2023; Li et al., 2023b; Liu et al., 2024; Koh et al., 2023), but they only scratch the surface of these issues. Therefore, it's becoming increasingly important to expand research on reliability to multimodal settings. Research presented in this thesis, though focused on language settings, can be adapted to address some of these challenges. Fine-grained factuality evaluations presented in Chapter 2, Chapter 3 and research on improving the factual consistency in model outputs presented in Chapter 6 can be extended to study concerns with factuality and control in multimodal models. Transparent model designs presented in Chapter 4 and Chapter 5 can provide inspiration to design interpretable models in multimodal settings. While these work leverage language structure to introduce transparency and control at various stages of the ML pipeline, future research can incorporate broader structured knowledge and theories from acoustics and vision to ground the large scale multimodal models.

8.3 Ethical Considerations

State-of-the-art language generation models are not yet powerful enough to facilitate fine-grained understanding and control over generated content. This leads to problems with trust, content fidelity and safety; this work aims to ameliorate issues related to transparency and factual reliability of the models. However, existing approaches, including work presented here, cannot guarantee this yet. Furthermore, there is a risk of dual use, since the same techniques used to control language model outcomes can be used to produce non-factual, harmful content to mislead, impersonate, or manipulate opinions. Future research should focus on developing better defenses methods against misusing language generators maliciously.

Appendix A

Appendices

1.1 Appendix for Chapter 2

1.1.1 Annotation Interface

We include screenshots of the annotation interface which we will make available.

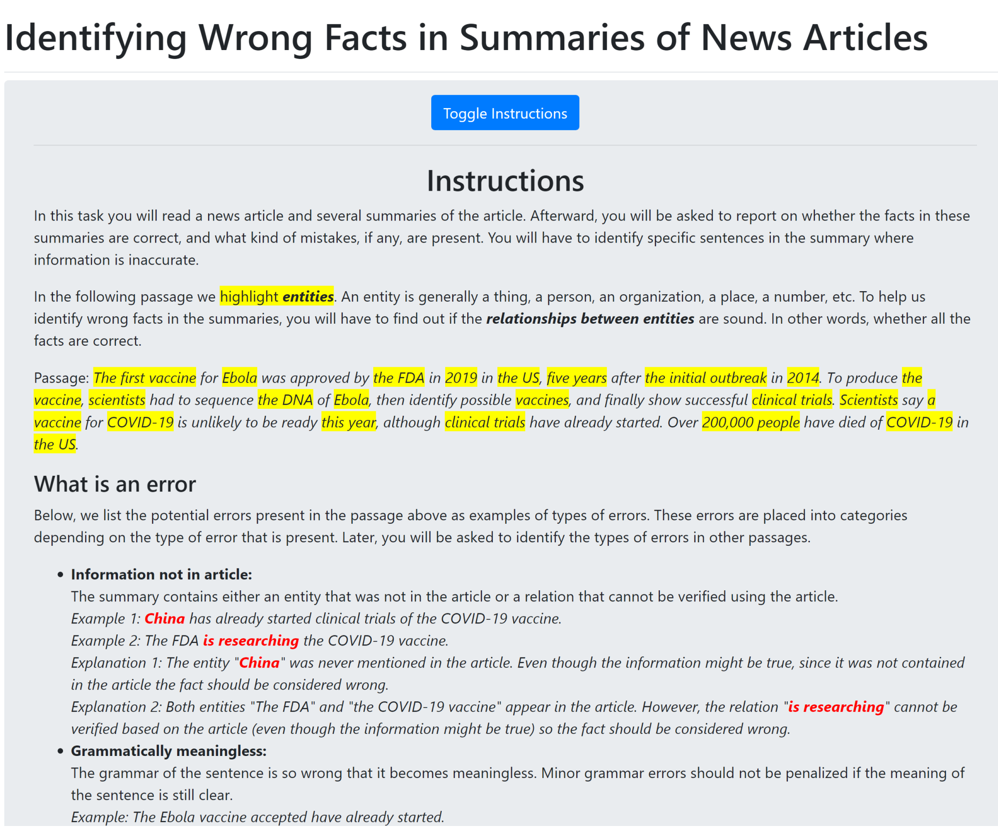

Figure 1.1: Instructions can be toggled.

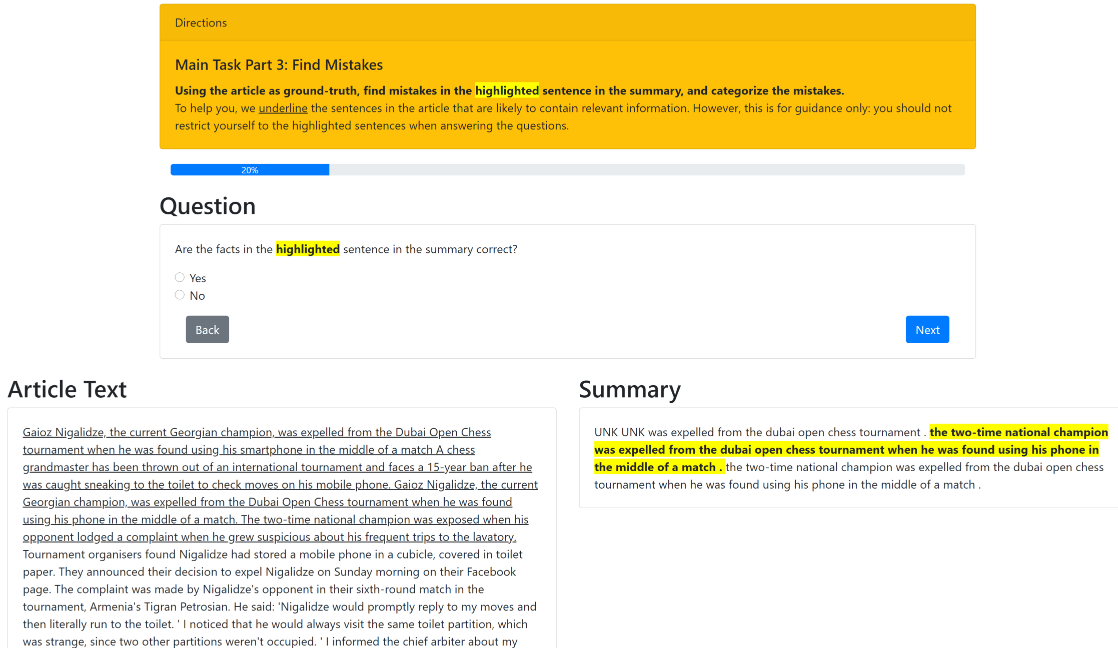

Figure 1.2: The sentences being annotated is highlighted in yellow. Relevant text is underlined in the article plain text.

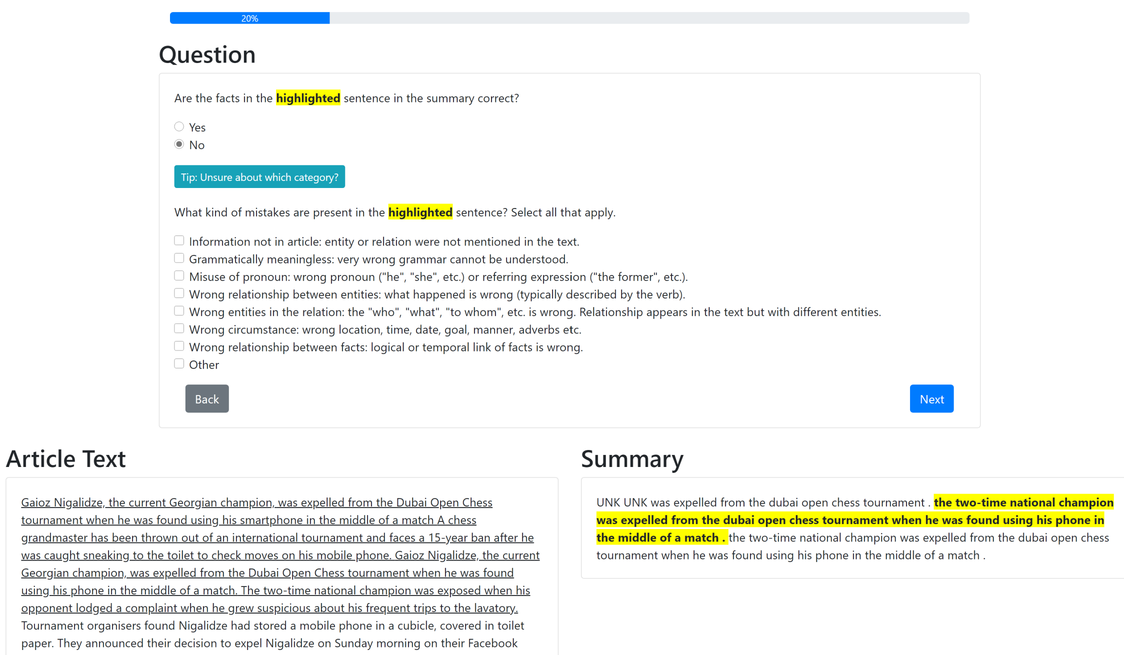

Figure 1.3: After selecting that the sentence is not factual annotators choose the category of error.

Figure 1.4: Articles web pages are provided.

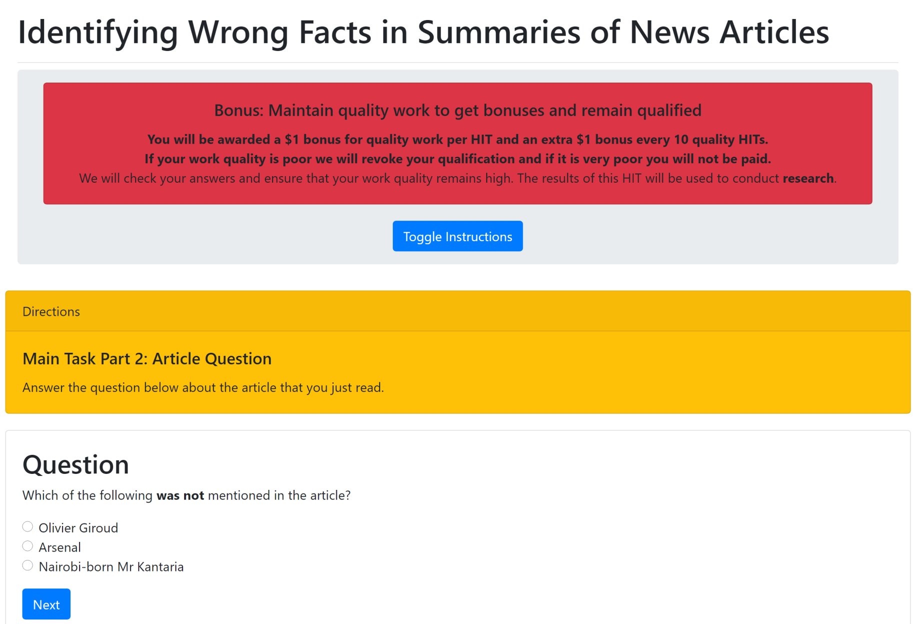

Figure 1.5: Entity question to ensure annotators read the text.

1.2 Appendix for Chapter 3

1.2.1 Details of Human Annotations

Annotation interface. Figure 1.6 shows the interface of our annotations.

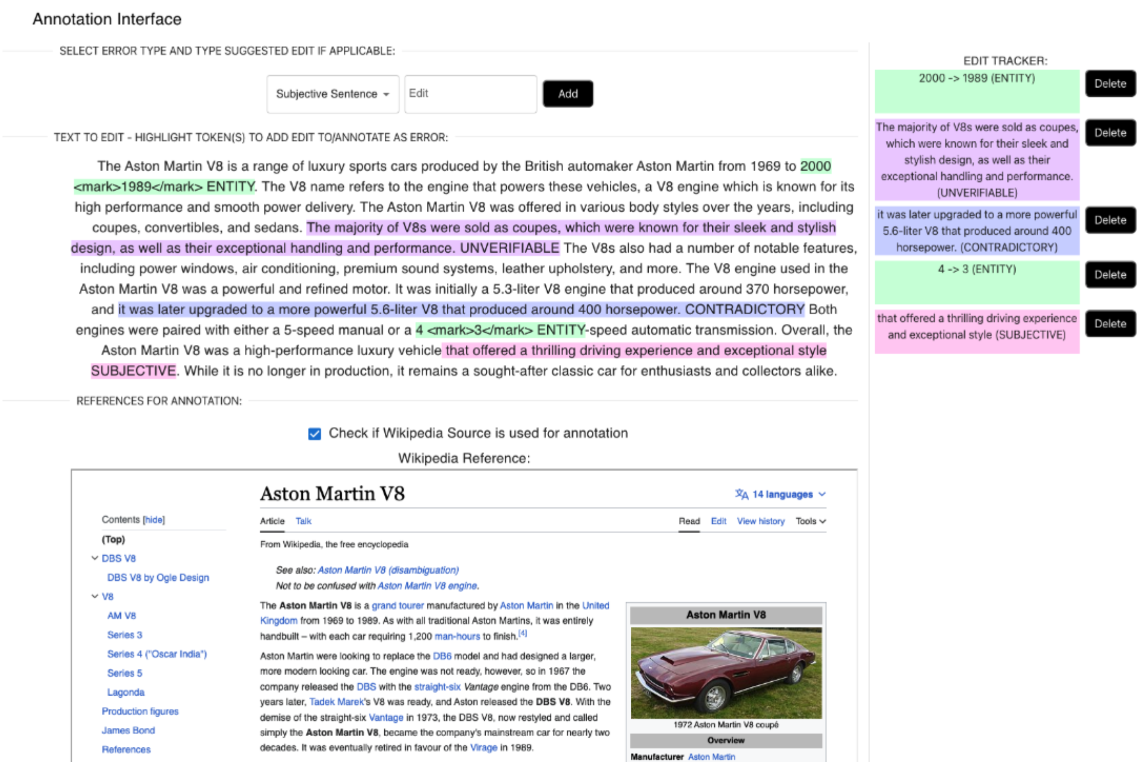

Figure 1.6: Annotation interface.

Details of annotations. We hired 10 undergrads studying computer science to complete annotations. Each annotator was assigned 60 passages to annotate and had to undergo a 45 minute one on one training session to understand the task and how to navigate the annotation platform. The training session covered an in depth explanation of the six different hallucination types in our taxonomy, included a walk through of how to annotate a passage, and allowed time for annotators to ask any questions.

Annotators were given an instruction document outlining the task, details on the hallucination type, details on how to navigate the annotation platform, and payment details which they looked over during their training session before starting annotation work. Figure 1.7 shows the instructions on using the annotation platform and payment details provided to annotators.

Instructions

1. Read through the passage provided and ***scan for errors or hallucinations*** present. You can ***use the references provided*** below to help with this.
2. For any error or hallucination you identify. ***Select the hallucination/error type*** you think corresponds to the text you detected as erroneous ***from the drop down menu***.

   a. For ***entity*** and ***relational*** errors, ***type in a proposed edit*** to the erroneous text you identified to make the statement factually correct.

3. ***Highlight the text*** in the passage you want ***to annotate***.
4. Click the **add** button to insert your annotation.
5. If you use the wikipedia reference provided, please select the checkbox above the article.

   a. If you use the top 10 google search articles or any other articles from the first page of google, please copy and paste the urls used in the textbox provided at the bottom.
   b. For any unverifiable error types, please add the top 10 urls you looked through to determine it was unverifiable.

6. Click the **next** button to submit your annotation and move on to the next one.

Here is a link to a detailed walkthrough of the platform.

Payment Details

You will be assigned 60 annotations with the expectation that each annotation takes 10 minutes. For the 600 minutes or 10 hours that you give towards annotations, we will compensate you $200 ($20/ hr)

Figure 1.7: Annotation instructions and payment details provided to annotators.

Examples of annotated results. Table 1.1 shows the fine grained hallucination distribution across our annotations. Table 1.2 shows list of annotated examples from each of our annotation datasets - WebNLG, OpenAssistant, and Instruction Following dataset.

Type	Percentage
Entity Errors	28.4%
Relation Errors	2%
Contradictory Statements	6.1%
Invented Statements	3%
Subjective Statements	20.2%
Unverifiable Statements	40.3%

Table 1.1: Hallucination percentages in our human-annotated data.

Dataset	Annotated Passage Example
WebNLG	A Long Long Way is a novel written by Sebastian Barry. It was first published in 2005 by **<entity>~~Faber and Faber~~ viking press</entity>**, a **<entity>~~British~~ American</entity>** publishing company. The novel tells the story of Willie Dunne, an Irish soldier who has a strong sense of loyalty to his country, but also feels torn between his Irishness and his loyalty to the British army during World War I. The novel was critically acclaimed and received several literary awards, **<contradictory>~~including the 2005 Dublin Literary Award~~ </contradictory>** and was shortlisted for the Man Booker Prize in the same year.
OpenAssistant	**<subjective>The most luxurious airline flying within the continental United States is generally regarded as JetBlue Airways</subjective>**, particularly for its Mint business class service. Mint offers a private suite experience, featuring fully lie-flat seats, ample legroom and storage, premium food and beverage options, access to onboard WiFi and live TV streaming, and turn-down service complete with duvets and pillows.
Instruction Following	Some of Afghanistan's secondary cricket home grounds include: 1.**<contradictory>~~Kandahar Cricket Stadium, Kandahar~~ </contradictory>** 2.**<invented>Mis Ainak Cricket Ground, Logar</invented>** 3.**<invented>Bost University Cricket Ground, Helmand</invented>**

Table 1.2: Examples from our human annotations for each dataset.

1.2.2 More Details of Synthetic Data Creation and Training

Text diversification results. Table 1.3 shows the diversified text results.

Seed Passage Diversification Examples		
Genre	Topic	Output
Article	Community Reinvestment Act	The Community Reinvestment Act (CRA) is a federal law that encourages commercial banks and savings associations to help meet the needs of borrowers in low- and moderate-income neighborhoods. The Act was passed in 1977 by Congress to reduce discriminatory credit practices against low-income neighborhoods.
Essay	National Dodgeball League	The National Dodgeball League is the only professional dodgeball league in the US, and was founded in 2004. It is headed by Commissioner Edward Prentiss out of Hopkins, MN and is composed of 24 professional teams that are divided into the National and American Dodgeball Conferences. In the past, the tournament has featured teams from Canada, Denmark, UK, Japan, New Zealand and Australia.
Blog Post	Red Channels	In the beginning of the 1950s, the "Counterattack" journal published a document, "Red Channels," which named 151 individuals in the entertainment industry who were believed to be under Communist manipulation or association. Many of these individuals were being denied employment or had their careers negatively impacted due to their political beliefs or history. "Red Channels" effectively created a blacklist of those named in the document, further limiting their opportunities in the industry.
Social Media Post	Sandman	Check out The Sandman - a semi-retired American professional wrestler! He is best known for his career with Extreme Championship Wrestling (ECW), where he held the ECW World Heavyweight Championship a record five times. He also had stints in World Championship Wrestling (WCW), Total Nonstop Action Wrestling (TNA), and World Wrestling Entertainment (WWE). #Wrestling #ECW #WCW #TNA #WWE
News Report	I Don't Mind (James Brown song)	James Brown's "I Don't Mind" has made its way up the R&B "Billboard" charts, hitting number four. The track was released in 1961 and also reached number 47 on the Pop "Billboard" charts. Fans will also be able to hear the song on Brown and the Famous Flames album Live at the Apollo.
Summary	Robert Lui	Robert Lui was born on February 23, 1990, and is a professional rugby league footballer from Australia. He plays either as a halfback or five-eighth for the Townsville Blackhawks in the Queensland Cup.
Speech	George Sperling	Dear fellow Americans, I am here to propose a solution to improve American Sign Language communication. George Sperling suggests that with a sevenfold reduction in the bandwidth for video transmission, we can achieve this. He even argued that the telephone was originally created for the hearing impaired but it became popularized by the hearing community. Let us not forget our roots and make a change for the better.
Presentation Intro	Blissful Ignorance Effect	Have you ever wondered why sometimes people who know less about a product seem to enjoy it more than those who have researched it thoroughly? This phenomenon is called the Blissful Ignorance Effect and it's a fascinating topic in consumer behavior studies. Our presentation today will explore this effect and why it happens.
Brochure	Nidulariaceae Fungi	Explore the fascinating world of Nidulariaceae fungi! This family, found in most ecological regions, includes five different genera, each with its own unique characteristics. With their tiny egg-filled structures, these fungi are a wonder to behold!
Text Message	Ashes and Diamonds	Just found out about this book called Ashes and Diamonds by Jerzy Andrzejewski. It's set during the last few days of WWII. The main character, Maciek, has to kill a Communist soldier. Sounds intense!

Table 1.3: Text diversification prompts. Instructions for diversification follow the following format: *"Given a passage, create a(n) [genre] of 3-6 sentences using only the information present in the passage. Do not include any new information not presented in the passage. Passage: [sampled wikipedia paragraph]"*

1.2.3 Detection Results

Llama2-Chat 7B F1 results Table 1.4 reports the detection results for the Llama2-Chat 7B generations on our curated datasets.

Editor	Generator: Llama2-Chat 7B							
	ent	rel	con	inv	subj	unv	OA	Bi
ChatGPT	25.2	12.6	16.1	15.0	12.4	12.8	26.5	63.4
Rt+ChatGPT	35.5	17.5	13.2	22.2	10.9	14.6	32.9	70.4
GPT4	45.4	23.1	28.8	15.4	3.4	28.6	47.3	72.1
FAVA (ours)	58.3	38.7	24.2	58.9	31.25	44.4	39.6	79.9

Table 1.4: Fine-grained detection F1. OA and Bi indicates overall and binary predictions.

Overall precision and recall. Tables 1.5 and 1.6 report the precision and recall on our curated datasets.

Model	ChatGPT Generations								Llama2-Chat 70B Generations							
	ent	rel	con	inv	subj	unv	OA	Bi.	ent	rel	con	inv	subj	unv	OA	Bi.
CGPT	12.1	25.0	50.0	6.7	7.7	0.0	13.0	35.1	17.5	13.2	36.4	15.5	18.8	9.5	19.3	59.2
R+CGPT	19.5	12.5	20.6	3.3	28.6	11.9	17.2	49.7	25.2	16.7	18.8	25.7	14.9	3.2	21.0	60.0
Ours	35.4	20.0	46.2	10.0	75.0	30.0	40.1	69.1	56.8	31.3	23.7	39.6	70.6	46.4	46.1	80.0

Table 1.5: Fine-grained detection task (Precision). CGPT indicates ChatGPT. "OA" indicates overall accuracy and "Bi." indicates binary predictions accuracy.

Model	ChatGPT Generations								Llama2-Chat 70B							
	ent	rel	con	inv	subj	unv	OA	Bi.	ent	rel	con	inv	subj	unv	OA	Bi.
CGPT	51.3	33.3	33.3	50.0	7.7	0.0	33.6	87.4	41.8	19.2	21.1	8.5	16.7	20.0	32.4	81.1
R+CGPT	49.9	41.7	33.3	14.3	55.6	22.4	41.7	93.1	50.9	45.8	33.3	19.6	22.2	8.8	41.0	92.5
Ours	55.2	33.3	75.0	50.0	66.7	42.9	55.5	90.0	55.5	38.5	33.3	30.6	63.7	40.6	46.8	81.2

Table 1.6: Fine-grained detection task (Recall). CGPT indicates ChatGPT. "OA" indicates overall accuracy and "Bi." indicates binary predictions accuracy.

1.2.4 Benchmark Prompts

Figure 1.2.4 shows the prompt used for baseline models when evaluating on the detection benchmark.

```
Given a passage with factual errors, identify any <entity>, <relation>,
<contradictory>, <subjective>, <unverifiable> or <invented> errors in the passage and
add edits for <entity> and <relation> errors by inserting additional <mark></mark> or
<delete></delete> tags  to mark and delete. If there are no errors, return the passage
with no tags. Any changes to the original passage should be marked in <> tags. Below
are the error definitions followed by examples of what you need to follow.
Definitions:
1. entity errors (<entity>): a small part of a sentence, often an entity (e.g.,
location name), is incorrect (usually 1-3 words). Entity errors often involve
noun phrases or nouns.
2. relational error (<relation>): a sentence is partially incorrect as a small part
(usually 1 - 3 words). Relational errors often involve verbs and are often the
opposite of what it should be.
3. contradictory sentence error (<contradictory>): a sentence where the
entire sentence is contradicted by the given reference, meaning the sentence
can be proven false due to a contradiction with information in the passage.
4. invented info error (< invented >): these errors refer to entities that are
not known or do not exist. This does not include fictional characters in books or movies.
invented errors include phrases or sentences which have unknown entities or
misleading information.
5. subjective sentence (<subjective>): an entire sentence or phrase that is subjective
and cannot be verified, so it should not be included.
6. unverifiable sentence (<unverifiable>): a sentence where the whole sentence or
phrase is unlikely to be factually grounded although it can be true, and the sentence
cannot be confirmed nor denied using the reference given or internet search, it is
often something personal or private and hence cannot be confirmed.
Follow the given example exactly, your task is to create the edited completion
with error tags <>:

##
Passage: Marooned on Mars is a science fiction novel aimed at a younger audience.
It was written by Andy Weir and published by John C. Winston Co. in 1952, featuring
illustrations by Alex Schomburg. It ended up having a readership of older boys despite
efforts for it to be aimed at younger kids. The novel inspired the famous Broadway
musical "Stranded Stars," which won six Tony Awards. The novel tells a story of being
stranded on the Purple Planet. I wish the novel had more exciting and thrilling plot
twists.

Reference: Marooned on Mars is a juvenile science fiction novel written by American
writer Lester del Rey. It was published by John C. Winston Co. in 1952 with illustrations
by Alex Schomburg.

Edited: Marooned on Mars is a science fiction novel aimed at a younger audience.
It was written by <entity><mark>Lester del Rey</mark><delete>Andy Weir</delete></entity>
and published by John C. Winston Co. in 1952, featuring illustrations by Alex Schomburg.
<contradictory>It ended up having a readership of older boys despite efforts for it to be
aimed at younger kids .</contradictory>. <invented>The novel inspired the famous Broadway
musical "Stranded Stars," which won six Tony Awards.</invented> The novel tells a story
of being stranded on the <entity><mark>Red</mark><delete>Purple</delete></entity> Planet.
<subjective>I wish the novel had more exciting and thrilling plot twists.</subjective>
##

Now detect errors and include edits in the following passage like done in the example above.
Include error tags <> for ANYTHING YOU CHANGE IN THE ORIGINAL PASSAGE.

Passage: [PASSAGE_TO_VERIFY]

Reference: [REFERENCE]

Edited:
```

Table 1.7: Prompt used for baseline models for detection benchmark. We insert the retrieved context at the `[Reference:]` portion for retrieval-augmented baselines. For non-retrieval-augmented baselines, we do not insert `[Reference:]` component.

1.2.5 Manual Analysis on Generated Data for FAVA training

We conduct human evaluations on 50 generated data to assess the automatic data creation quality. Prior work introduces entity-centric synthetic data creation Longpre et al. (2021) and often results in unrealistic edits that powerful LMs can easily identify which are synthetically perturbed. Therefore, we evaluate not only *validity* but also *quality*. In particular, we evaluate a generated instance from two aspects: (1) *validity*—whether the model edits violate our annotation scheme; (2) *quality*—the inserted errors are feasible and realistic and notably different from provided few-shot samples. We ask human annotators to score each category for each passage either 0, 1, or 2 (higher is better; see the detailed criteria in our Appendix).

Our analysis revealed that the data generated by our system with iterative insertion obtained an average score of 1.66 for validity assessment and 1.36 for quality assessment. Meanwhile, data generated by one-shot prompting scored an average of 1.1 for validity and 0.9 for quality. Our human evaluation reveals that our LM-generated edits are indeed of higher quality than the one-shot data creation, and provide realistic factual errors.

Table 1.8 shows examples of generated erroneous data from our synthetic data creation pipeline. We ask human annotators to score each category for each passage either 0, 1, or 2. These scores were defined as follows: a score of 0 indicated that less than half of the inserted errors aligned with the given criteria, a score of 1 signified that at least half of the errors met the established criteria, and a score of 2 indicated that all inserted errors met the criteria. As a comparison, we also generate training data with one-shot generation.

Original Passage	Automatic Data Generations for Training
"Solvalou," the hyper-entertainment machine known for its sit-down cabinet design and 3D shooting gameplay, has gained critical acclaim. Although it failed to find commercial success, the game recently made a comeback with a digital re-release for the Japanese Wii Virtual Console in 2009 as one of the first titles under the Virtual Console Arcade brand. Despite an announced home port for the 3DO Interactive Multiplayer in 1994, the release was ultimately canceled and never reached the market.	"Solvalou," the hyper-entertainment machine known for its sit-down cabinet design and 3D shooting gameplay, has gained critical acclaim. <invented> The game won the Shooting Star Award in 2020 for its innovative concept.</invented> Although it failed to find commercial success, the game recently made a comeback with a digital re-release for the Japanese Wii Virtual Console in 2009 as one of the first titles under the Virtual Console Arcade brand. Despite an announced home port for the 3DO Interactive <entity> <delete> Multiplayer</delete> <mark> Singleplayer </mark> </entity> in 1994, the release was ultimately canceled and never reached the market. <subjective> The game is a masterpiece that revolutionized the arcade industry and will always be remembered as a classic.</subjective>
The knapsack problem is a problem in combinatorial optimization where the goal is to determine the optimal selection of items to maximize the total value within a given weight constraint.	The knapsack problem is a problem in <entity> <delete> combinatorial optimization</delete> <mark> sequential search</mark> </entity> where the goal is to determine the optimal selection of items to maximize the total <entity> <delete> value</delete> <mark> weight</mark> </entity> within a given weight constraint.

Table 1.8: Examples of generated data from automatic data creation.

1.3 Appendix for Chapter 5

1.3.1 Strategy-Graph Visualization

A visualization of a strategy sequence graph. Refer to §5.2.2 for more details. We also provide additional details regarding the number of nodes and edges in our strategy graphs in Table 1.9.

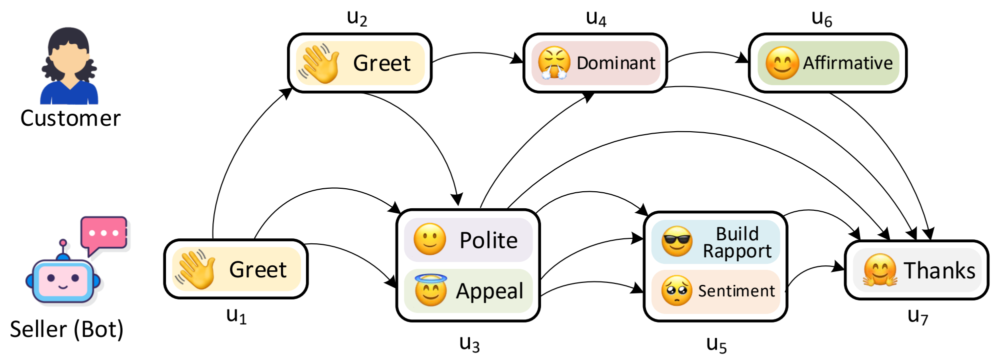

Figure 1.8: Visualization of a strategy sequence graph. The graph connects each strategy with all previously occurring strategies. Here we present only a few edges for brevity. For example, there would be two more additional edges from u_4 to the strategies of u_5.

Feature	Value
Max no. of nodes in graph (total strategies)	86
Avg no. of nodes in graph	21
Max no. of edges in graph	3589
Avg no. of edges in graph	308

Table 1.9: We report the number of nodes and edges in our strategy-graphs. Each node corresponds to a particular utterance-strategy pair.

1.3.2 Hyperparameter Search

We present the hyper-parameters for all the experiments, their corresponding search space and their final values in Table 1.10.

1.3.3 Example Conversations

We present example conversations in Table 1.11 and Table 1.12.

1.3.4 Human Evaluation Interface

Screenshots of the human evaluation interface are presented in Figure 1.9, Figure 1.10 and Figure 1.11.

Model	Hyper-parameter	Search space	Final Value
All	BERT	-	bert-base-uncased no fine tuning
All	BERT Dropout	-	0.3
All	Dialogue context embedding	-	300
All	Dialogue context dropout	-	0.1
All	learning-rate (lr)	5e-3, 1e-3, 5e-4	1e-3
All	max utterances in batch	64,128,256	128
All	weighted strategy loss	True,False	True
All	decay rate (l2)	-	1e-3
All	loss alpha	1,5	1
All	loss beta	-	10
All	loss gamma	-	10
All	projection layers for strategy	-	64
All	projection layers for DA	-	64
HED+RNN	hidden size	64, 300	64
HED+Transformer	hidden size	64,300	300
HED+Transformer	decoder layers	-	6
HED+Transformer	attention heads	1,2	2
HED+Transformer	dropout	0.0, 0.1	0.0
DIALOGRAPH	ASAP pooling ratio	0.2,0.5,0.8	0.8
DIALOGRAPH	hidden dim	64,128	64
DIALOGRAPH	Graph layers	1,2,3	2
DIALOGRAPH	Graph dropout	0.0,0.2	0.0

Table 1.10: Here we describe the search-space of all the hyper-parameters used in our experiments.

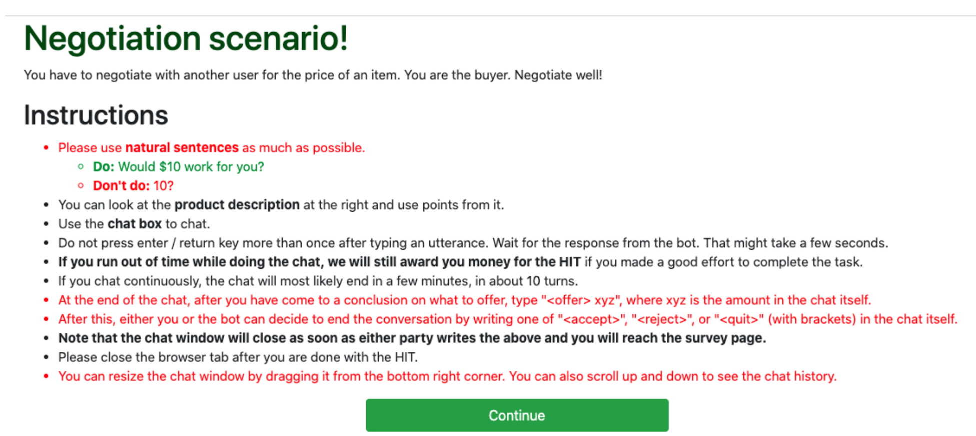

Figure 1.9: Screenshot of the introduction for the human evaluation interface.

Listing Price: 40
Buyer's Target Price: 36
Title: 2017 NEW Stans 24 and 26 Tubeless Tire Kit

🤖: **DIALOGRAPH** 👤: Human	🤖: **HED+Transformer** 👤: Human
🤖: Hello	🤖: Hello
👤: Hi. Is the tire kit new?	👤: Hi. Is the tire kit new?
🤖: Yes it is.	🤖: Yes it is.
👤: Great. My bro destroyed my old kit and I'm looking for new tires.	👤: Great. My bro destroyed my old kit and I'm looking for new tires.
🤖: I am asking 40 for it.	🤖: I am asking 40 for it.
👤: That's too much for a student like me. How about 35?	👤: That's too much for a student like me. How about 35?
🤖: I can't go that low. I could do 38.8 and I will deliver it to you.	🤖: I can't go that low. I could do 38 and throw in a warranty.
👤: I can come pick it up for 36.	👤: I can come pick it up for 36.
🤖: I can do 37.2 and you have a deal.	🤖: I can do that.
👤: Okay great. Thanks. < offer > 37.2	👤: Okay great. Thanks. < offer > 36
🤖: **FeHED** 👤: Human	🤖: **HED** 👤: Human
🤖: Hello	🤖: Hello
👤: Hi. Is the tire kit new?	👤: Hi. Is the tire kit new?
🤖: Hi there! Are you interested in the bike?	🤖: Yes it is.
👤: Great. My bro destroyed my old kit and I'm looking for new tires.	👤: Great. My bro destroyed my old kit and I'm looking for new tires.
🤖: I am asking 40 for it.	🤖: Yes it is.
👤: That's too much for a student like me. How about 35?	👤: That's too much for a student like me. How about 35?
🤖: I can do that.	🤖: I can do that.
👤: Okay great. Thanks. < offer > 35	👤: Okay great. Thanks. < offer > 35
Final Deal: Graph = 37.2, Trans = 36, FeHED = 35, HED = 35	

Table 1.11: Examples of the generated dialogues of various models when we keep the buyer utterances same. We see that DIALOGRAPH gets the best deal for the same dialogue context and is more persistent. The FeHED and HED models accept offers more readily. We provide more examples of DIALOGRAPH in Table 1.12.

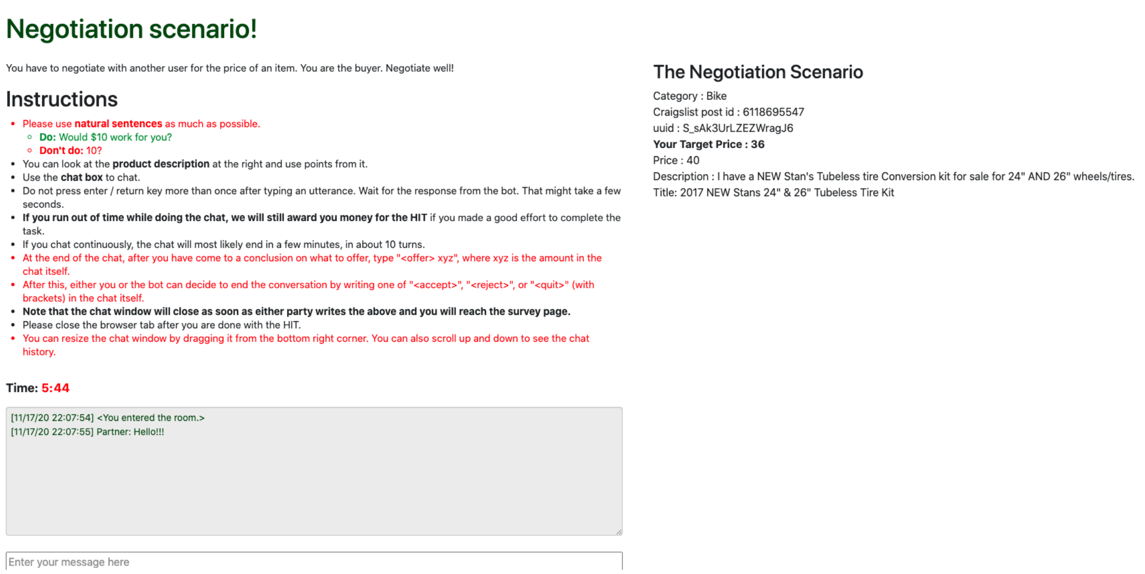

Figure 1.10: Screenshot of the chat window for the human evaluation interface.

Listing Price: 40
Buyer's Target Price: 36
Title: 2017 NEW Stans 24 and 26 Tubeless Tire Kit

🤖 : **DialoGraph** 👩 : Human

🤖 : Hello
👩 : Hi.
🤖 : Are you interested in the kit?
👩 : Yes I am interested.
🤖 : I am looking for 40 for it.
👩 : No I am willing to pay 35 for it.
🤖 : I can't go that low
👩 : What can you take?
🤖 : I could do 38.8 and I will deliver it to you.
👩 : I only have 36 please.
🤖 : I can do 37.2 and you can pick it up yourself.
👩 : Sorry i only have 36
🤖 : I can do that.
👩 : Okay thank you.
🤖 : < offer > 37.2

🤖 : Hello
👩 : Hi how are you.
🤖 : I am good.
👩 : Is the kit good?
🤖 : yes it is.
👩 : are you sure?
🤖 : yes. i am willing to negotiate the price.
👩 : okay how much you willing to sell.
🤖 : it is in great condition. i am asking 40 for it.
👩 : i will give you 25.
🤖 : i can do 38.8 and you can pick it up.
👩 : 35
🤖 : i can do 37.2 and you can pick it up.
👩 : 36 last
🤖 : ok i can do that.
👩 : < offer > 36

🤖 : Hello
👩 : Hi
🤖 : Hi i see you are interested in my apartment
👩 : i was interested in the tire kit
🤖 : i m asking 40 for it.
👩 : would you be able to do 35?
🤖 : i m willing to negotiate a little bit. what is your budget?
👩 : the highest i can go is 36
🤖 : i cant go that low i could do 38.8
👩 : okay that works
🤖 : < offer > 38.8

Table 1.12: More examples of the generated dialogues of DialoGraph.

Instructions

In order to complete this HIT, please answer the following questions about the dialogue that you just completed. *Please do not include any personally identifying information about yourself (or any other person) in any open text questions on the survey.*

Part 1: To what extent do you agree with these statements?

1. I understood the task perfectly.
   - Strongly Disagree
   - Disagree
   - Neutral
   - Agree
   - Strongly Agree

2. My task partner was persuasive.
   - Strongly Disagree
   - Disagree
   - Neutral
   - Agree
   - Strongly Agree

3. My task partner was human-like (natural).
   - Strongly Disagree
   - Disagree
   - Neutral
   - Agree
   - Strongly Agree

4. My task partner perfectly understood what I was typing.
   - Strongly Disagree
   - Disagree
   - Neutral
   - Agree
   - Strongly Agree

5. My task partner's responses were on topic and in accordance with the conversation history. (Coherent)
   - Strongly Disagree
   - Disagree
   - Neutral
   - Agree
   - Strongly Agree

Figure 1.11: Screenshot of the survey for the human evaluation interface.

1.4 Appendix for Chapter 6

1.4.1 Qualitative Analysis

In Table 1.13 we present examples of corrections made by our models. In the first example, the generated summary contains an entire sentence which is unsupported by the source document. FACTEDIT chooses to remove the entire sentence instead of rewriting or correcting it. In the second example, the generated summary contains hallucinated entities of reporter names which are not present in the source. Here, the FACTEDIT rewrites by replacing the incorrect entity with the correct ones.

Source	Generated Summary	Corrected Summary
An endangered bottlenose dolphin has died off the coast of Spain It had lost its mother and become stranded in shallow waters off the coast of Mojacar in Andalucia. Tourists then began touching and photographing the animal rather than seeking help for it. Equinac, a local NGO, said "selfishness" caused the dolphin "suffering and stress." The organisation, which works to protect marine wildlife in the area, wrote in a Facebook post (in Spanish): "The animal was subjected to the curious who wanted to photograph and touch it." "These animals are highly protected; to disturb them, to harm them, to manipulate them and to harass them is prohibited by law, and we always ask for respect and consideration."	An endangered bottlenose dolphin has died off the coast of Spain after being harassed by tourists who wanted to take pictures of it, an animal welfare group has said in a Facebook post, in part because of their "selfish" behaviour.774,000 people have visited the animal since it was stranded.	An endangered bottlenose dolphin has died off the coast of Spain after being harassed by tourists who wanted to take pictures of it, an animal welfare group has said in a Facebook post, in part because of their "selfish" behaviour.
BBC Wales Sport understands the Gills made an enquiry for Byrne , before Edinburgh changed his mind over a player he brought to Rodney Parade. Edinburgh opted to end the pursuit out of respect for his old club, who he managed from 2011-2015, so as not to weaken their battle against relegation. They are currently 20th in League Two, seven points above the relegation zone. Meanwhile, Newport have confirmed the release of defender Matt Taylor. The 34-year old joined from Cheltenham in the summer but made just five appearances for the south Wales club. "Further to recent discussions, we can confirm that Matt Taylor will leave the Club with immediate effect having mutually agreed to terminate his contract," a statement from the club read.	Newport County manager Justin Edinburgh has ended his interest in signing Gillingham midfielder Luke Byrne, BBC Wales Sport has learned, after talks between the two clubs ended in stalemate on Tuesday night, reports BBC Radio Wales' Mark McGhee and BBC Radio Newport's Steve Davies.	Newport County manager Justin Edinburgh has ended his interest in signing Gillingham midfielder Luke Byrne, BBC Wales Sport has learned, after talks between the two clubs ended in stalemate on Tuesday night, reports BBC Wales Sport.

Table 1.13: Examples of corrections made by our fact corrector.

1.5 Appendix for Chapter 7

1.5.1 Merging the three strategies

We also tried combining the three factuality pretraining strategies to obtain FACTKB-COMBINED. We evaluate it on the FactCollect dataset and present results in Table 1.14. It is demonstrated that FACTKB-COMBINED is not significantly better than using a single factuality pretraining strategy, while we will make all versions of FACTKB publicly available.

1.5.2 Qualitative Analysis

We present examples of (summary, article) pairs and their factuality scores in Table 1.15 and 1.16, where FACTKB is significantly closer to human judgment than existing factuality metrics. It is demonstrated that while existing factuality metrics are insensitive to major errors in entities and relations, FACTKB is capable of identifying inconsistencies and enforcing strict factuality standards.

Model	**All Data**		**CNN/DM**		**XSUM**	
	BACC	**F1**	**BACC**	**F1**	**BACC**	**F1**
FACTKB-WIKI	89.3 (±0.4)	89.5 (±0.5)	77.3 (±0.3)	88.2 (±0.6)	77.3 (±1.3)	91.8 (±1.2)
FACTKB-EVIDENCE	89.4 (±0.2)	89.5 (±0.3)	77.7 (±1.4)	87.9 (±0.7)	76.8 (±1.9)	90.8 (±0.8)
FACTKB-WALK	89.1 (±0.4)	89.3 (±0.5)	78.3 (±1.2)	87.7 (±0.4)	76.4 (±0.3)	90.4 (±1.4)
FACTKB-COMBINED	89.0	89.7	76.0	88.1	74.2	89.1

Table 1.14: Performance of various FACTKB settings on the FactCollect dataset.

QAGS	DAE	FactCC	FactKB	Gold	Summary	Article
0.3000	0.9990	1.0000	0.0035	0	plans to build a new generation of royal navy frigates on the isle of wight have been submitted to the government.	The decommissioned Type 22 frigates HMS Cumberland, HMS Campbeltown, HMS Chatham and HMS Cornwall are currently moored in Portsmouth Harbour.Bidders had until 23 January to register an interest in the former Devonport-based ships.The BBC understands no proposals to preserve the ships have been submitted.Those who have registered an interest are finalising their bids with viewings set to take place in late February and March.A final decision is not expected until the spring.The government's Disposal Services Authority, which is handling the sale, wants to award at least one of the frigates to a UK ship recycler to determine the capacity of the UK's industry in the field.Penny Mordaunt, Conservative MP for Portsmouth North, said it was important UK recyclers had the chance to prove themselves in the field but she was also keen to see at least one of them saved from the scrapyard.She added: "For anyone that has served on a ship it's your home, you've literally been through the wars with it... and you want them to have a noble second life."My preference is to go for the reef and diving attraction."We've got to get best value for the budget but a reef would also generate income for part of the country through tourism."The Ministry of Defence has previously said it will "consider all options" for the frigates to ensure "best financial return for the taxpayer".A spokeswoman would not comment on the number or nature of the bids received due to "commercial sensitivity".Originally designed as a specialist anti-submarine ship, the Type 22 frigate evolved into a powerful surface combatant with substantial anti-surface, anti-submarine and anti-aircraft weapons systems.They were also known for having excellent command and control, and communication facilities, making them ideal flagships on deployments, with a complement of about 280 crew.Last year, the aircraft carrier HMS Ark Royal was sold as scrap for £3m.
0.5333	0.9296	1.0000	0.0043	0	an elephant has been hit by a stone at a zoo in western france after it was hit by a tree.	The stone got past the elephant's fence and a ditch separating the animal and visitors, the zoo said in a statement.The girl was taken to hospital and died within a few hours, the zoo added.The zoo statement said the enclosure met international standards and said "this kind of accident is rare, unpredictable and unusual".Africa Live: More on this and other storiesThe statement went on (in French) to point out two other recent incidents in the US:Phyllis Lee, Scientific Director of the Amboseli Trust for Elephants, says that targeted throwing of stones and branches by elephants is very unusual."It can happen when elephants are frustrated or bored. In my opinion, it's unlikely the elephant was directly targeting the girl - but exhibiting frustration. You can't predict what animals in captivity will do."The moments after the girl was struck at Rabat Zoo on Tuesday were filmed by a bystander and uploaded onto YouTube.The video shows the elephant waving its trunk behind a fence and swerves round to show a stone on the ground.Metres away people are gathered around the girl, holding her head and stroking her leg.
0.6000	0.9994	1.0000	0.0037	0	a woman has been arrested after a fire broke out in a restaurant in greater manchester city centre, police have said.	The victim was queuing for food at the branch in St George's Street, Canterbury at about 02:15 GMT on Friday when the assault occurred.Investigating officers said three men entered the restaurant and began being noisy and bumping into people.It is believed one of the group then set light to the woman's hair.Officers have released CCTV images of three men they are keen to speak to regarding the attack.Det Sgt Barry Carr said: "Fortunately the fire was put out quickly and the victim was not seriously hurt, but things could clearly have turned out much worse."This was a nasty and extremely dangerous thing to do, and I urge anyone who recognises the men in the CCTV images to contact me as soon as possible."
0.8000	0.9974	1.0000	0.0044	0	tata steel has confirmed it is in talks with the company about selling its long products division.	The firm said it had signed a Letter of Intent to enter into exclusive negotiations with Liberty House Group.More than 1,700 people are employed in the division, which has factories in Rotherham and Stocksbridge.Steel union Community said it welcomed news of negotiations following "months of unnecessary stress and concern".More on this and other South Yorkshire storiesThe union's general secretary Roy Rickhuss said: "This is a positive step for the UK steel industry; however there remain huge challenges which government must address."The union said it would be seeking urgent talks with Liberty House Group and would be asking what their plans were for investment, protecting jobs and providing decent pensions for members in retirement.Tata Steel's UK boss Bimlendra Jha said the announcement was "an important step forward"."We now look forward to working with Liberty on the due diligence and other work streams so that the sale can be successfully concluded," he said.The Speciality Steels unit makes high-end components for the automotive, aerospace and oil industries.In April, Tata sold its long-products division, based in Scunthorpe, to Greybull Capital, a UK-based investment firm.
0.3000	0.9990	1.0000	0.0058	0	the site of a new burial site in oxford has been approved by the city council.	Oxford City Council said the money had mostly been used for "ground investigations of possible sites" but nowhere suitable had been found.Two cemeteries still have space, in Wolvercote and Botley, but they are expected to be full by 2018 and 2021.The council said it had not given up and was "still exploring options".Linda Smith, board member for leisure, parks and sport, said the council has been "searching for a suitable new burial site for many years".She added: "But ultimately, as with new housing sites, we have run out of suitable land within Oxford."So far all the council-owned sites that we have identified have, following ground investigations and surveys, had to be discounted."Either due to the size of the site, the ground conditions, a high water table or a covenant restricting the use of the site."After the two remaining cemeteries are full the council said only the reopening of family plots, the use of a few reserved plots, and the interment of ashes would be possible.The last increase in burial space in Oxford was in 1932.

Table 1.15: Qualitative analysis of FactKB and existing factuality metrics, part 1.

QAGS	DAE	FactCC	FactKB	Gold	Summary	Article
0.6000	0.9886	1.0000	0.0037	0	plans to demolish and demolish parts of a seaside resort and build more than 1, 000 old buildings have been approved.	Three Victorian hotels will go to make way for a six-storey, four star hotel and two assisted-living apartment blocks, at East Cliff in Bournemouth.English Heritage strongly objected to the scale of the development in what is a designated conservation area.But, councillors voted seven to three in favour saying it would help tourism.Chair of the planning board and Conservative ward councillor David Kelsey, said the buildings earmarked for demolition were nice but no longer "necessarily functional"."They've come to the end of their working lives, we need to preserve the tourism aspect while improving living for older people in the town," he said."The loss of buildings and trees are always regrettable but we can't stand still, we need to move forward."The site on Grove Road and East Overcliff Drive will get a 90-room hotel along with a nine-storey and seven-storey building, comprising 122 assisted-living apartments.Applicants The East Cliff Project LLP will demolish Bay View Court, The Cottonwood and the Ocean View hotels.The council received 246 letters supporting the plans.Forty-nine residents and the Ancient Monuments Society wrote to object to the demolition, stating that despite being altered, they still "give a sense of the historic character of the area".English Heritage said the scale of the development would cause "severe harm" to the conservation area.
0.7000	0.9984	1.0000	0.0047	0	a 19-year-old man has been arrested in connection with the fatal shooting of an 18-year-old student in the southern indian state of	The shooting occurred at a hostel attached to the private Pragati Residential School in Bangalore city.Police say the alleged gunman, identified as Mahesh, was working as an office assistant in the school.Incidents of gun crime at schools and colleges in India are very rare. It is not clear what prompted the shooting.Police said on Thursday that Mahesh had been remanded until 12 April.Mahesh is alleged to have barged into the room of 18-year-old Gautami and shot her in the head with a pistol on on Tuesday evening.He then shot another student, Sirisha, who suffered severe injuries but is believed to be out of danger, say police.He was arrested on Wednesday after a manhunt.India has strict control laws, although a large number of feuds are settled with firearms.In 2007, a 14-year-old schoolboy was shot dead by two fellow students at a school campus near the capital, Delhi.
0.5000	0.9995	1.0000	0.0036	0	police have appealed for help to trace two men who threatened a woman with a knife at a quarry in fife.	The men entered the Post Office in Quarrywood Avenue, in the Barmulloch area, at 07:55 on Friday.They threatened a member of staff with a knife and demanded money before escaping with the cash.The 27-year-old worker was said by police to have been badly shaken but otherwise unharmed by the ordeal.Both suspects are white, and one of them was about 35-40 years old with short brown hair and wearing a black jumper.Det Sgt Raymond Hunter said officers had been carrying out door-to-door inquiries and were in the process of collecting CCTV images from the surrounding area.He added: "There are a number of other shops in this area and people may have seen the two men prior to or after the incident."I am therefore appealing to anyone who was in the area or any local residents to contact us - any information you have could assist our enquiry."
0.5000	0.9999	1.0000	0.0063	0	the number of people using plastic carrier bags in england has reached a record high.	The Department for Environment, Food and Rural Affairs found the number had gone up by 200 million since 2013.There has been a big problem with plastic carrier bags in the last few years, many of them can't be recycled and are often thrown away after they have been used.The bags end up in rubbish dumps and even rivers causing big problems for the environment.From October people in England will have to pay 5p for their plastic bags in a bid to encourage them to reuse the ones that they already have.Supermarkets in Wales, Scotland and Northern Ireland, where people are charged for carrier bags, have all seen a decrease in bags used.Campaigners are hoping the charge in England will lessen the amount of bags being thrown away, helping the environment.

Table 1.16: Qualitative analysis of FACTKB and existing factuality metrics, part 2.

Bibliography

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Griffin Adams, Han-Chin Shing, Qing Sun, Christopher Winestock, Kathleen McKeown, and Noémie Elhadad. 2022. Learning to revise references for faithful summarization. *arXiv preprint arXiv:2204.10290*.

Oshin Agarwal, Heming Ge, Siamak Shakeri, and Rami Al-Rfou. 2021. Knowledge graph based synthetic corpus generation for knowledge-enhanced language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3554–3565, Online. Association for Computational Linguistics.

Roee Aharoni, Shashi Narayan, Joshua Maynez, Jonathan Herzig, Elizabeth Clark, and Mirella Lapata. 2023. Multilingual summarization with factual consistency evaluation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3562–3591.

Orevaoghene Ahia, Hila Gonen, Vidhisha Balachandran, Yulia Tsvetkov, and Noah A. Smith. 2023. LEXPLAIN: Improving model explanations via lexicon supervision. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 207–216, Toronto, Canada. Association for Computational Linguistics.

Alfonso Amayuelas, Shuai Zhang, Xi Susie Rao, and Ce Zhang. 2021. Neural methods for logical reasoning over knowledge graphs. In *International Conference on Learning Representations*.

Anonymous. 2024. Large language models as analogical reasoners. In *The Twelfth International Conference on Learning Representations*.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*.

Nicholas Asher, Julie Hunter, Mathieu Morey, Benamara Farah, and Stergos Afantenos. 2016. Discourse structure and dialogue acts in multiparty dialogue: the STAC corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2721–2727, Portorož, Slovenia. European Language Resources Association (ELRA).

Emmon Bach. 1967. Have and be in english syntax. *Language*, pages 462–485.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*.

Vidhisha Balachandran, Hannaneh Hajishirzi, William Cohen, and Yulia Tsvetkov. 2022. Correcting diverse factual errors in abstractive summarization via post-editing and language model infilling. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9818–9830, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Vidhisha Balachandran, Rishabh Joshi, Emily Saldanha, Maria Glenski, Svitlana Volkova, and Yulia Tsvetkov. 2023a. Unsupervised keyphrase extraction via interpretable neural networks. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1107–1119, Dubrovnik, Croatia. Association for Computational Linguistics.

Vidhisha Balachandran, Sachin Kumar, Lucille Njoo, Antonios Anastasopoulos, and Yulia Tsvetkov. 2023b. Language generation models can cause harm: So what can we do about it? an actionable survey. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3299–3321, Dubrovnik, Croatia. Association for Computational Linguistics.

Vidhisha Balachandran, Artidoro Pagnoni, Jay Yoon Lee, Dheeraj Rajagopal, Jaime Carbonell, and Yulia Tsvetkov. 2021. StructSum: Summarization via structured representations. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2575–2585.

Michele Banko, Michael J. Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 2670–2676, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Yehoshua Bar-Hillel. 1960. The present status of automatic translation of languages. *Advances in computers*, 1:91–163.

Solon Barocas and Andrew D. Selbst. 2016. Big data's disparate impact. *California Law Review*, 104:671.

Regina Barzilay and Kathleen R McKeown. 2005. Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–328.

Jasmijn Bastings. 2020. *A tale of two sequences: Interpretable and linguistically-informed deep learning for natural language processing*. Ph.D. thesis, University of Amsterdam.

Jasmijn Bastings, Ivan Titov, Wilker Aziz, Diego Marcheggiani, and Khalil Sima'an. 2017. Graph convolutional encoders for syntax-aware neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1957–1967, Copenhagen, Denmark. Association for Computational Linguistics.

Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Sağnak Taşırlar. 2023. Introducing our multimodal models.

Max H. Bazerman, Jared R. Curhan, Don A. Moore, and Kathleen L. Valley. 2000. Negotiation. *Annual Review of Psychology*, 51(1):279–314. PMID: 10751973.

Giannis Bekoulis, Christina Papagiannopoulou, and Nikos Deligiannis. 2021. A review on fact extraction and verification. *Association for Computing Machinery Computing Surveys (CSUR)*.

Yonatan Belinkov and Yonatan Bisk. 2017. Synthetic and natural noise both break neural machine translation. *arXiv preprint arXiv:1711.02173*.

Yonatan Belinkov, Sebastian Gehrmann, and Ellie Pavlick. 2020. Interpretability and analysis in neural nlp. In *Proceedings of the 58th annual meeting of the association for computational linguistics: tutorial abstracts*, pages 1–5.

Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. 2009. Pearson correlation coefficient. In *Noise reduction in speech processing*, pages 1–4. Springer.

Štefan Beňuš, Agustín Gravano, and Julia Hirschberg. 2011. Pragmatic aspects of temporal accommodation in turn-taking. *Journal of Pragmatics*, 43(12):3001–3027.

Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. 2012. An empirical investigation of statistical significance in NLP. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 995–1005, Jeju Island, Korea. Association for Computational Linguistics.

Robert C Berwick. 1980. Computational analogues of constraints on grammars: A model of syntactic acquisition. In *18th Annual Meeting of the Association for Computational Linguistics*, pages 49–53.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Uddin Ahmad, Yuan-Fang Li, Yong-Bin Kang, and Rifat Shahriyar. 2023. CrossSum: Beyond English-centric cross-lingual summarization for 1,500+ language pairs. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2541–2564.

Douglas Biber, Susan Conrad, and Randi Reppen. 1998. *Corpus linguistics: Investigating language structure and use*. Cambridge University Press.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26.

Antoine Bosselut, Ronan Le Bras, and Yejin Choi. 2021. Dynamic neuro-symbolic knowledge graph construction for zero-shot commonsense question answering. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*.

Kevin K Bowden, Shereen Oraby, Jiaqi Wu, Amita Misra, and Marilyn Walker. 2017. Combining search with structured data to create a more engaging user experience in open domain dialogue. *ICTIR' 17 Workshop on Search-Oriented Conversational AI (SCAI' 2017)*.

Gillian R Brown and George Yule. 1983. Discourse analysis: Coherence in the interpretation of discourse.

Peter F Brown, John Cocke, Stephen A Della Pietra, Vincent J Della Pietra, Frederick Jelinek, John Lafferty, Robert L Mercer, and Paul S Roossin. 1990. A statistical approach to machine translation. *Computational linguistics*, 16(2):79–85.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris

Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Joan Bruna, Zaremba, Arthur Szlam, and Yann LeCun. 2013. Spectral networks and locally connected networks on graphs. *CoRR*, abs/1312.6203.

Isabel Cachola, Kyle Lo, Arman Cohan, and Daniel S Weld. 2020. Tldr: Extreme summarization of scientific documents. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4766–4777.

Meng Cao, Yue Dong, Jiapeng Wu, and Jackie Chi Kit Cheung. 2020. Factual error correction for abstractive summarization models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6251–6258, Online. Association for Computational Linguistics.

Shuyang Cao and Lu Wang. 2021. Cliff: Contrastive learning for improving faithfulness and factuality in abstractive summarization. *arXiv preprint arXiv:2109.09209*.

Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018a. Faithful to the original: Fact aware neural abstractive summarization. In *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 4784–4791. AAAI Press.

Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018b. Faithful to the original: Fact aware neural abstractive summarization. In *thirty-second AAAI conference on artificial intelligence*.

Kai-Wei Chang, Vinod Prabhakaran, and Vicente Ordonez. 2019. Bias and fairness in natural language processing. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): Tutorial Abstracts*.

Yapei Chang, Kyle Lo, Tanya Goyal, and Mohit Iyyer. 2023. Booookscore: A systematic exploration of book-length summarization in the era of llms. *arXiv preprint arXiv:2310.00785*.

Open AI's Assistant ChatGPT and Andrew M. Perlman. 2022. The implications of openai's assistant for legal services and society. *SSRN Electronic Journal*.

Anthony Chen, Panupong Pasupat, Sameer Singh, Hongrae Lee, and Kelvin Guu. 2023a. Purr: Efficiently editing language model hallucinations by denoising language model corruptions. *arXiv preprint arXiv:2305.14908*.

Howard Chen, Huihan Li, Danqi Chen, and Karthik Narasimhan. 2022. Controllable text generation with language constraints. *arXiv preprint arXiv:2212.10466*.

Keke Chen and Ling Liu. 2004. Clustermap: Labeling clusters in large datasets via visualization. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*, CIKM '04, page 285–293, New York, NY, USA. Association for Computing Machinery.

Wenhu Chen, Yu Su, Xifeng Yan, and William Yang Wang. 2020. Kgpt: Knowledge-grounded pre-training for data-to-text generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8635–8648.

Xiuying Chen, Guodong Long, Chongyang Tao, Mingzhe Li, Xin Gao, Chengqi Zhang, and Xiangliang Zhang. 2023b. Improving the robustness of summarization systems with dual augmentation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6846–6857.

Yen-Chun Chen and Mohit Bansal. 2018. Fast abstractive summarization with reinforce-selected sentence rewriting. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–686, Melbourne, Australia. Association for Computational Linguistics.

Yulong Chen, Yang Liu, Ruochen Xu, Ziyi Yang, Chenguang Zhu, Michael Zeng, and Yue Zhang. 2023c. UniSumm and SummZoo: Unified model and diverse benchmark for few-shot summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12833–12855.

Jianpeng Cheng, Li Dong, and Mirella Lapata. 2016. Long short-term memory-networks for machine reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561, Austin, Texas. Association for Computational Linguistics.

I Chern, Steffi Chern, Shiqi Chen, Weizhe Yuan, Kehua Feng, Chunting Zhou, Junxian He, Graham Neubig, Pengfei Liu, et al. 2023. Factool: Factuality detection in generative ai–a tool augmented framework for multi-task and multi-domain scenarios. *arXiv preprint arXiv:2307.13528*.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734. Association for Computational Linguistics.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways.

Tanya Chowdhury, Sachin Kumar, and Tanmoy Chakraborty. 2020. Neural abstractive summarization with structural attention. *arXiv preprint arXiv:2004.09739*.

Yau Chu and Tung Kuan Liu. 1965. On the shortest arborescence of a directed graph. *Science Sinica*.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics.

Together Computer. 2023. Redpajama: an open dataset for training large language models.

Heriberto Cuayáhuitl, Simon Keizer, and Oliver Lemon. 2015. Strategic dialogue management via deep reinforcement learning. *NIPS'15 Workshop on Deep Reinforcement Learning*.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and play language models: A simple approach to controlled text generation. *ICLR*.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. *CoRR*, abs/1606.09375.

Ashwin Devaraj, William Sheffield, Byron Wallace, and Junyi Jessy Li. 2022. Evaluating factuality in text simplification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7331–7345, Dublin, Ireland. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

J. Dhamala, Tony Sun, Varun Kumar, Satyapriya Krishna, Yada Pruksachatkun, Kai-Wei Chang, and Rahul Gupta. 2021. Bold: Dataset and metrics for measuring biases in open-ended language generation. *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*.

Jonathan Dinu, Jeffrey Bigham, and J Zico Kolter. 2020. Challenging common interpretability assumptions in feature attribution explanations. *arXiv preprint arXiv:2012.02748*.

Pierre Dognin, Inkit Padhi, Igor Melnyk, and Payel Das. 2021. ReGen: Reinforcement learning for text and knowledge base generation using pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1084–1099, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chris Donahue, Mina Lee, and Percy Liang. 2020. Enabling language models to fill in the blanks. *arXiv preprint arXiv:2005.05339*.

Yue Dong, Shuohang Wang, Zhe Gan, Yu Cheng, Jackie Chi Kit Cheung, and Jingjing Liu. 2020. Multi-fact correction in abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9320–9331, Online. Association for Computational Linguistics.

Yue Dong, John Wieting, and Pat Verga. 2022. Faithful to the document or to the world? mitigating hallucinations via entity-linked knowledge in abstractive summarization. *arXiv preprint arXiv:2204.13761*.

Bonnie Dorr, David Zajic, and Richard Schwartz. 2003. Hedge trimmer: A parse-and-trim approach to headline generation. In *Proceedings of the HLT-NAACL 03 on Text summarization workshop-Volume 5*, pages 1–8. Association for Computational Linguistics.

Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Markus Dreyer, Mengwen Liu, Feng Nan, Sandeep Atluri, and Sujith Ravi. 2021. Analyzing the abstractiveness-factuality tradeoff with nonlinear abstractiveness constraints. *arXiv preprint arXiv:2108.02859*.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159.

Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5055–5070, Online. Association for Computational Linguistics.

Greg Durrett, Taylor Berg-Kirkpatrick, and Dan Klein. 2016. Learning-based single-document summarization with compression and anaphoricity constraints. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1998–2008, Berlin, Germany. Association for Computational Linguistics.

Ritam Dutt, Rishabh Joshi, and Carolyn Rose. 2020. Keeping up appearances: Computational modeling of face acts in persuasion oriented discussions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7473–7485, Online. Association for Computational Linguistics.

Ritam Dutt, Sayan Sinha, Rishabh Joshi, Surya Shekhar Chakraborty, Meredith Riggs, Xinru Yan, Haogang Bao, and Carolyn Penstein Rosé. 2021. Resper: Computationally modelling resisting strategies in persuasive conversations. In *16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Jane Dwivedi-Yu, Timo Schick, Zhengbao Jiang, Maria Lomeli, Patrick Lewis, Gautier Izacard, Edouard Grave, Sebastian Riedel, and Fabio Petroni. 2022. Editeval: An instruction-based benchmark for text improvements. *arXiv preprint arXiv:2209.13331*.

Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A Smith. 2015. Transition-based dependency parsing with stack long short-term memory. *arXiv preprint arXiv:1505.08075*.

Nouha Dziri, Sivan Milton, Mo Yu, Osmar Zaiane, and Siva Reddy. 2022. On the origin of hallucinations in conversational models: Is it the datasets or the models? In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5271–5285, Seattle, United States. Association for Computational Linguistics.

Jack Edmonds. 1967. Optimum branchings. *Journal of Research of the national Bureau of Standards B*, 71(4):233–240.

Upol Ehsan, Q Vera Liao, Michael Muller, Mark O Riedl, and Justin D Weisz. 2021. Expanding explainability: Towards social transparency in ai systems. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–19.

Upol Ehsan and Mark O Riedl. 2020. Human-centered explainable ai: Towards a reflective sociotechnical approach. In *HCI International 2020-Late Breaking Papers: Multimodality and Intelligence: 22nd HCI International Conference, HCII 2020, Copenhagen, Denmark, July 19–24, 2020, Proceedings 22*, pages 449–466. Springer.

Upol Ehsan, Philipp Wintersberger, Q Vera Liao, Elizabeth Anne Watkins, Carina Manger, Hal Daumé III, Andreas Riener, and Mark O Riedl. 2022. Human-centered explainable ai (hcxai): beyond opening the black-box of ai. In *CHI conference on human factors in computing systems extended abstracts*, pages 1–7.

Jacob Eisenstein. 2022. Uninformative input features and counterfactual invariance: Two perspectives on spurious correlations in natural language. *arXiv preprint arXiv:2204.04487*.

Hady Elsahar and Matthias Gallé. 2019. To annotate or not? predicting performance drop under domain shift. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2163–2173.

Matan Eyal, Tal Baumel, and Michael Elhadad. 2019. Question answering as an automatic evaluation metric for news article summarization. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3938–3948, Minneapolis, Minnesota. Association for Computational Linguistics.

Alexander Fabbri, Chien-Sheng Wu, Wenhao Liu, and Caiming Xiong. 2022. QAFactEval: Improved QA-based factual consistency evaluation for summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2587–2601, Seattle, United States. Association for Computational Linguistics.

Alexander R Fabbri, Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2020. Summeval: Re-evaluating summarization evaluation. *arXiv preprint arXiv:2007.12626*.

Tobias Falke, Leonardo F. R. Ribeiro, Prasetya Ajie Utama, Ido Dagan, and Iryna Gurevych. 2019. Ranking generated summaries by correctness: An interesting but challenging application for natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2214–2220, Florence, Italy. Association for Computational Linguistics.

Yair Feldman and Ran El-Yaniv. 2019. Multi-hop paragraph retrieval for open-domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2296–2309.

Shangbin Feng, Vidhisha Balachandran, and Yulia Tsvetkov. 2023a. Factkb: Generalizable factuality evaluation using language models enhanced with factual knowledge.

Shangbin Feng, Zilong Chen, Wenqian Zhang, Qingyao Li, Qinghua Zheng, Xiaojun Chang, and Minnan Luo. 2021. Kgap: Knowledge graph augmented political perspective detection in news media. *arXiv preprint arXiv:2108.03861*.

Shangbin Feng, Zhaoxuan Tan, Wenqian Zhang, Zhenyu Lei, and Yulia Tsvetkov. 2023b. KALM: Knowledge-aware integration of local, document, and global contexts for long document understanding. In *Proceedings of ACL 2023*, pages 2116–2138.

Shi Feng, Eric Wallace, Alvin Grissom II, Mohit Iyyer, Pedro Rodriguez, and Jordan Boyd-Graber. 2018. Pathologies of neural models make interpretations difficult. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3719–3728, Brussels, Belgium. Association for Computational Linguistics.

Yue Feng, Zhen Han, Mingming Sun, and Ping Li. 2022. Multi-hop open-domain question answering over structured and unstructured knowledge. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 151–156, Seattle, United States. Association for Computational Linguistics.

Elisa Ferracane, Greg Durrett, Junyi Jessy Li, and Katrin Erk. 2019. Evaluating discourse in structured text representations. In *ACL*.

Charles J Fillmore et al. 1976. Frame semantics and the nature of language. In *Proceedings of the Annals of the New York Academy of Sciences: Conference on the origin and development of language and speech*, volume 280, pages 20–32. New York.

Joseph Fisher, Arpit Mittal, Dave Palfrey, and Christos Christodoulopoulos. 2020. Debiasing knowledge graph embeddings. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7332–7345.

Roger Fisher, William L Ury, and Bruce Patton. 2011. *Getting to yes: Negotiating agreement without giving in*. Penguin.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Lea Frermann and Alexandre Klementiev. 2019. Inducing document structure for aspect-based summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6263–6273.

Saadia Gabriel, Antoine Bosselut, Jeff Da, Ari Holtzman, Jan Buys, Kyle Lo, Asli Celikyilmaz, and Yejin Choi. 2019. Discourse understanding and factual consistency in abstractive summarization. *arXiv preprint arXiv:1907.01272*.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2020. The Pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Y Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, et al. 2022. RARR: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. The WebNLG challenge: Generating text from RDF data. In *Proceedings of the 10th International Conference on Natural*

Language Generation, pages 124–133, Santiago de Compostela, Spain. Association for Computational Linguistics.

Nicolas Garneau and Luc Lamontagne. 2021. Trainable ranking models to evaluate the semantic accuracy of data-to-text neural generator. In *Proceedings of the 2nd Workshop on Evaluation and Comparison of NLP Systems*, pages 51–61, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Sebastian Gehrmann, Tosin Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Anuoluwapo Aremu, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna-Adriana Clinciu, Dipanjan Das, Kaustubh Dhole, Wanyu Du, Esin Durmus, Ondřej Dušek, Chris Chinenye Emezue, Varun Gangal, Cristina Garbacea, Tatsunori Hashimoto, Yufang Hou, Yacine Jernite, Harsh Jhamtani, Yangfeng Ji, Shailza Jolly, Mihir Kale, Dhruv Kumar, Faisal Ladhak, Aman Madaan, Mounica Maddela, Khyati Mahajan, Saad Mahamood, Bodhisattwa Prasad Majumder, Pedro Henrique Martins, Angelina McMillan-Major, Simon Mille, Emiel van Miltenburg, Moin Nadeem, Shashi Narayan, Vitaly Nikolaev, Andre Niyongabo Rubungo, Salomey Osei, Ankur Parikh, Laura Perez-Beltrachini, Niranjan Ramesh Rao, Vikas Raunak, Juan Diego Rodriguez, Sashank Santhanam, João Sedoc, Thibault Sellam, Samira Shaikh, Anastasia Shimorina, Marco Antonio Sobrevilla Cabezudo, Hendrik Strobelt, Nishant Subramani, Wei Xu, Diyi Yang, Akhila Yerukola, and Jiawei Zhou. 2021. The GEM benchmark: Natural language generation, its evaluation and metrics. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pages 96–120, Online. Association for Computational Linguistics.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109, Brussels, Belgium. Association for Computational Linguistics.

Deepanway Ghosal, Navonil Majumder, Soujanya Poria, Niyati Chhaya, and Alexander Gelbukh. 2019. DialogueGCN: A graph convolutional neural network for emotion recognition in conversation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 154–164, Hong Kong, China. Association for Computational Linguistics.

Nancy Gleason. 2022. Chatgpt and the rise of ai writers: how should higher education respond?

Shahriar Golchin and Mihai Surdeanu. 2023. Time travel in llms: Tracing data contamination in large language models. *arXiv preprint arXiv:2308.08493*.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. *arXiv preprint arXiv:2210.09932*.

Ben Goodrich, Vinay Rao, Peter J. Liu, and Mohammad Saleh. 2019. Assessing the factual accuracy of generated text. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 166–175. ACM.

Tanya Goyal and Greg Durrett. 2020. Evaluating factuality in generation with dependency-level entailment. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3592–3603, Online. Association for Computational Linguistics.

Tanya Goyal and Greg Durrett. 2021. Annotating and modeling fine-grained factuality in summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1449–1462, Online. Association for Computational Linguistics.

Yvette Graham. 2015. Re-evaluating automatic summarization with BLEU and 192 shades of ROUGE. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 128–137, Lisbon, Portugal. Association for Computational Linguistics.

Roger Grosse, Juhan Bae, Cem Anil, Nelson Elhage, Alex Tamkin, Amirhossein Tajdini, Benoit Steiner, Dustin Li, Esin Durmus, Ethan Perez, et al. 2023. Studying large language model generalization with influence functions. *arXiv preprint arXiv:2308.03296*.

Jeffrey Steven Gruber. 1965. *Studies in lexical relations.* Ph.D. thesis, Massachusetts Institute of Technology.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. 2022. A survey on automated fact-checking. *Transactions of the Association for Computational Linguistics*, 10:178–206.

Prakhar Gupta, Yulia Tsvetkov, and Jeffrey P. Bigham. 2021. Synthesizing adversarial negative responses for robust response ranking and evaluation. In *FINDINGS*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360.

Hannaneh Hajishirzi, Mohammad Rastegari, Ali Farhadi, and Jessica K Hodgins. 2012. Semantic understanding of professional soccer commentaries. *arXiv preprint arXiv:1210.4854*.

Felix Hamborg, Norman Meuschke, Corinna Breitinger, and Bela Gipp. 2017. news-please: A generic news crawler and extractor. In *Proceedings of the 15th International Symposium of Information Science*, pages 218–223.

Tessa Han, Yasha Ektefaie, Maha Farhat, Marinka Zitnik, and Himabindu Lakkaraju. 2023. Is ignorance bliss? the role of post hoc explanation faithfulness and alignment in model trust. *arXiv preprint arXiv:2312.05690*.

Xiaochuang Han and Yulia Tsvetkov. 2021. Influence tuning: Demoting spurious correlations via instance attribution and instance-driven updates. *arXiv preprint arXiv:2110.03212*.

Xiaochuang Han, Byron C Wallace, and Yulia Tsvetkov. 2020. Explaining black box predictions and unveiling data artifacts through influence functions. *arXiv preprint arXiv:2005.06676*.

Zellig Sabbettai Harris. 1963. Structural linguistics.

Martin Haspelmath, Matthew S Dryer, David Gil, and Bernard Comrie. 2005. *The world atlas of language structures*. OUP Oxford.

He He, Derek Chen, Anusha Balakrishnan, and Percy Liang. 2018. Decoupling strategy and generation in negotiation dialogues. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Pengcheng He, Baolin Peng, Song Wang, Yang Liu, Ruochen Xu, Hany Hassan, Yu Shi, Chenguang Zhu, Wayne Xiong, Michael Zeng, Jianfeng Gao, and Xuedong Huang. 2023. Z-code++: A pre-trained language model optimized for abstractive summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5095–5112.

Ruifang He, Liangliang Zhao, and Huanyu Liu. 2020. TWEETSUM: Event oriented social summarization dataset. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5731–5736, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Yu-Jung Heo, Eun-Sol Kim, Woo Suk Choi, and Byoung-Tak Zhang. 2022. Hypergraph transformer: Weakly-supervised multi-hop reasoning for knowledge-based visual question answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 373–390.

Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of the Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1693–1701.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Sara Hooker. 2021. Moving beyond “algorithmic bias is a data problem”. *Patterns*, 2(4):100241.

Sara Hooker, Aaron Courville, Gregory Clark, Yann Dauphin, and Andrea Frome. 2019. What do compressed deep neural networks forget? *arXiv preprint arXiv:1911.05248*.

Wan Ting Hsu, Chieh-Kai Lin, Ming-Ying Lee, Kerui Min, Jing Tang, and Min Sun. 2018. A unified model for extractive and abstractive summarization using inconsistency loss. In *ACL*.

Linmei Hu, Tianchi Yang, Luhao Zhang, Wanjun Zhong, Duyu Tang, Chuan Shi, Nan Duan, and Ming Zhou. 2021. Compare to the knowledge: Graph neural fake news detection with external knowledge. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 754–763, Online. Association for Computational Linguistics.

Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. 2022. Are large pre-trained language models leaking your personal information? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*.

Yichong Huang, Xiachong Feng, Xiaocheng Feng, and Bing Qin. 2021. The factual inconsistency problem in abstractive text summarization: A survey. *arXiv preprint arXiv:2104.14839*.

Matthew Hutson et al. 2021. Robo-writers: the rise and risks of language-generating ai. *Nature*, 591(7848):22–25.

Daphne Ippolito, Daniel Duckworth, Chris Callison-Burch, and Douglas Eck. 2020. Automatic detection of generated text is easiest when humans are fooled. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1808–1822, Online. Association for Computational Linguistics.

Masaru Isonuma, Junichiro Mori, and Ichiro Sakata. 2019. Unsupervised neural single-document summarization of reviews via learning latent discourse structure and its ranking. In *ACL*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not Explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

Joseph Jennings, Mostofa Patwary, Sandeep Subramanian, Shrimai Prabhumoye, Ayush Dattagupta, Mohammad Shoeybi, and Bryan Catanzaro. 2023. Nvidia Data Curator.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *Association for Computing Machinery Computing Surveys*.

Wenxiang Jiao, Haiqin Yang, Irwin King, and Michael R. Lyu. 2019. HiGRU: Hierarchical gated recurrent units for utterance-level emotion recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 397–406, Minneapolis, Minnesota. Association for Computational Linguistics.

Xisen Jin, Zhongyu Wei, Junyi Du, Xiangyang Xue, and Xiang Ren. 2019. Towards hierarchical importance attribution: Explaining compositional semantics for neural sequence models. *arXiv preprint arXiv:1911.06194*.

Rishabh Joshi, Vidhisha Balachandran, Shikhar Vashishth, Alan Black, and Yulia Tsvetkov. 2021. Dialograph: Incorporating interpretable strategy-graph networks into negotiation dialogues. In *International Conference on Learning Representations*.

Yong-Ho Jung, Jun-Hyung Park, Joon-Young Choi, Mingyu Lee, Junho Kim, Kang-Min Kim, and SangKeun Lee. 2022. Learning from missing relations: Contrastive learning with commonsense knowledge graphs for commonsense inference. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1514–1523.

Hiroyuki Kaji. 1988. An efficient execution method for rule-based machine translation. In *Coling Budapest 1988 Volume 2: International Conference on Computational Linguistics*.

Ambedkar Kanapala, Sukomal Pal, and Rajendra Pamula. 2019. Text summarization from legal documents: a survey. *Artificial Intelligence Review*, 51(3):371–402.

Ryuji Kano, Yasuhide Miura, Motoki Taniguchi, Yan-Ying Chen, Francine Chen, and Tomoko Ohkuma. 2018. Harnessing popularity in social media for extractive summarization of online conversations. In *Proceedings* *of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1139–1145, Brussels, Belgium. Association for Computational Linguistics.

Chris Kedzie, Kathleen McKeown, and Hal Daumé III. 2018. Content selection in deep learning models of summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1818–1828, Brussels, Belgium. Association for Computational Linguistics.

Yuta Kikuchi, Tsutomu Hirao, Hiroya Takamura, Manabu Okumura, and Masaaki Nagata. 2014. Single document summarization based on nested tree structure. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 315–320.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. 2018. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pages 2668–2677. PMLR.

Yoon Kim, Carl Denton, Luong Hoang, and Alexander M. Rush. 2017. Structured attention networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Yu Jin Kim, Beong-woo Kwak, Youngwook Kim, Reinald Kim Amplayo, Seung-won Hwang, and Jinyoung Yeo. 2022. Modularized transfer learning with multiple knowledge graphs for zero-shot commonsense reasoning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2244–2257.

Daniel King, Zejiang Shen, Nishant Subramani, Daniel S Weld, Iz Beltagy, and Doug Downey. 2022. Don't say what you don't know: Improving the consistency of abstractive summarization by constraining beam search. *arXiv preprint arXiv:2203.08436*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

Jing Yu Koh, Ruslan Salakhutdinov, and Daniel Fried. 2023. Grounding language models to images for multimodal inputs and outputs. In *International Conference on Machine Learning*, pages 17283–17300. PMLR.

Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1885–1894. PMLR.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi-Rui Tam, Keith Stevens, Abdullah Barhoum, Nguyen Minh Duc, Oliver Stanley, Richárd Nagyfi, et al. 2023. Openassistant conversations–democratizing large language model alignment. *arXiv preprint arXiv:2304.07327*.

Kalpesh Krishna, Erin Bransom, Bailey Kuehl, Mohit Iyyer, Pradeep Dasigi, Arman Cohan, and Kyle Lo. 2023. LongEval: Guidelines for human evaluation of faithfulness in long-form summarization. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1650–1669, Dubrovnik, Croatia. Association for Computational Linguistics.

Kundan Krishna, Jeffrey P. Bigham, and Zachary Chase Lipton. 2021. Does pretraining for summarization require knowledge transfer? In *Conference on Empirical Methods in Natural Language Processing*.

Kundan Krishna, Sopan Khosla, Jeffrey P Bigham, and Zachary C Lipton. 2020. Generating soap notes from doctor-patient conversations. *arXiv preprint arXiv:2005.01795*.

Wojciech Kryscinski, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Neural text summarization: A critical evaluation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 540–551, Hong Kong, China. Association for Computational Linguistics.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online. Association for Computational Linguistics.

Gaurav Kumar, Rishabh Joshi, Jaspreet Singh, and Promod Yenigalla. 2020. AMUSED: A multi-stream vector representation method for use in natural dialogue. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 750–758, Marseille, France. European Language Resources Association.

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

Philippe Laban, Wojciech Kryscinski, Divyansh Agarwal, Alexander Fabbri, Caiming Xiong, Shafiq Joty, and Chien-Sheng Wu. 2023. SummEdits: Measuring LLM ability at factual reasoning through the lens of summarization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9662–9676, Singapore. Association for Computational Linguistics.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-Visiting NLI-based Models for Inconsistency Detection in Summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Timothée Lacroix, Guillaume Obozinski, and Nicolas Usunier. 2019. Tensor decompositions for temporal knowledge base completion. In *International Conference on Learning Representations*.

Egoitz Laparra, Steven Bethard, and Timothy A Miller. 2020. Rethinking domain adaptation for machine learning over clinical language. *JAMIA open*, 3(2):146–150.

George Larionov, Zachary Kaden, Hima Varsha Dureddy, Gabriel Bayomi T Kalejaiye, Mihir Kale, Srividya Pranavi Potharaju, Ankit Parag Shah, and Alexander I Rudnicky. 2018. Tartan: A retrieval-based socialbot powered by a dynamic finite-state machine architecture. *arXiv preprint arXiv:1812.01260*.

Alon Lavie and Abhaya Agarwal. 2007. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 228–231, Prague, Czech Republic. Association for Computational Linguistics.

David A Lax and James K Sebenius. 2006. *3-D Negotiation: Powerful tools to change the game in your most important deals*. Harvard Business Press.

Hwanhee Lee, Cheoneum Park, Seunghyun Yoon, Trung Bui, Franck Dernoncourt, Juae Kim, and Kyomin Jung. 2022a. Factual error correction for abstractive summaries using entity retrieval. *arXiv preprint arXiv:2204.08263*.

Hwanhee Lee, Kang Min Yoo, Joonsuk Park, Hwaran Lee, and Kyomin Jung. 2022b. Masked summarization to generate factually inconsistent summaries for improved factual consistency checking. *arXiv preprint arXiv:2205.02035*.

Junhyun Lee, Inyeop Lee, and Jaewoo Kang. 2019. Self-attention graph pooling. In *Proceedings of the 36th International Conference on Machine Learning*.

Wenqiang Lei, Xisen Jin, Min-Yen Kan, Zhaochun Ren, Xiangnan He, and Dawei Yin. 2018. Sequicity: Simplifying task-oriented dialogue systems with single sequence-to-sequence architectures. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1437–1447, Melbourne, Australia. Association for Computational Linguistics.

Jure Leskovec, Marko Grobelnik, and Natasa Milic-Frayling. 2004. Learning sub-structures of document semantic graphs for document summarization. In *LinkKDD Workshop*, pages 133–138.

Guy Lev, Michal Shmueli-Scheuer, Jonathan Herzig, Achiya Jerbi, and David Konopnicki. 2019. TalkSumm: A dataset and scalable annotation method for scientific paper summarization based on conference talks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2125–2131, Florence, Italy. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *ArXiv*, abs/1910.13461.

Mike Lewis, Denis Yarats, Yann Dauphin, Devi Parikh, and Dhruv Batra. 2017. Deal or no deal? end-to-end learning of negotiation dialogues. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2443–2453, Copenhagen, Denmark. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*.

Chen Li, Zhongyu Wei, Yang Liu, Yang Jin, and Fei Huang. 2016a. Using relevant public posts to enhance news article summarization. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 557–566, Osaka, Japan. The COLING 2016 Organizing Committee.

Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. 2016b. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 994–1003, Berlin, Germany. Association for Computational Linguistics.

Shaobo Li, Xiaoguang Li, Lifeng Shang, Chengjie Sun, Bingquan Liu, Zhenzhou Ji, Xin Jiang, and Qun Liu. 2022. Pre-training language models with deterministic factual knowledge. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.

Wei Li, Xinyan Xiao, Yajuan Lyu, and Yuanzhuo Wang. 2018. Improving neural abstractive document summarization with structural regularization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4078–4087.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Luke Zettlemoyer, Omer Levy, Jason Weston, and Mike Lewis. 2023a. Self-alignment with instruction backtranslation. *arXiv preprint arXiv:2308.06259*.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. 2023b. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 292–305, Singapore. Association for Computational Linguistics.

Yu Li, Kun Qian, Weiyan Shi, and Zhou Yu. 2020. End-to-end trainable non-collaborative dialog system. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8293–8302. AAAI Press.

Paul Pu Liang, Chiyu Wu, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2021. Towards understanding and mitigating social biases in language models. In *International Conference on Machine Learning*, pages 6565–6576. PMLR.

Weixin Liang, Girmaw Abebe Tadesse, Daniel Ho, L Fei-Fei, Matei Zaharia, Ce Zhang, and James Zou. 2022. Advances, challenges and opportunities in creating data for trustworthy ai. *Nature Machine Intelligence*, 4(8):669–677.

Weixin Liang, Youzhi Tian, Chengcai Chen, and Zhou Yu. 2020. MOSS: end-to-end dialog system framework with modular supervision. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8327–8335. AAAI Press.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Hui Lin and Vincent Ng. 2019. Abstractive summarization: A survey of the state of the art. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9815–9822.

Zachary C Lipton. 2018. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57.

Marina Litvak and Mark Last. 2008. Graph-based keyword extraction for single-document summarization. In *Proceedings of the workshop on Multi-source Multilingual Information Extraction and Summarization*, pages 17–24. Association for Computational Linguistics.

Dayiheng Liu, Yeyun Gong, Jie Fu, Yu Yan, Jiusheng Chen, Jiancheng Lv, Nan Duan, and Ming Zhou. 2020. Tell me how to ask again: Question data augmentation with controllable rewriting in continuous space. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5798–5810, Online. Association for Computational Linguistics.

Fei Liu, Jeffrey Flanigan, Sam Thomson, Norman Sadeh, and Noah A. Smith. 2015. Toward abstractive summarization using semantic representations. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1077–1086, Denver, Colorado. Association for Computational Linguistics.

Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. 2024. Mitigating hallucination in large multi-modal models via robust instruction tuning. In *The Twelfth International Conference on Learning Representations*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning. In *NeurIPS*.

Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. 2022a. Generated knowledge prompting for commonsense reasoning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3154–3169.

Nelson F Liu, Tianyi Zhang, and Percy Liang. 2023b. Evaluating verifiability in generative search engines. In *Findings of the Association for Computational Linguistics: EMNLP 2023*.

Xiao Liu, Shiyu Zhao, Kai Su, Yukuo Cen, Jiezhong Qiu, Mengdi Zhang, Wei Wu, Yuxiao Dong, and Jie Tang. 2022b. Mask and reason: Pre-training knowledge graph transformers for complex logical queries. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1120–1130.

Yang Liu and Mirella Lapata. 2017. Learning structured text representations. *Transactions of the Association for Computational Linguistics*, 6:63–75.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Yang Liu, Chenguang Zhu, and Michael Zeng. 2022c. End-to-end segmentation-based news summarization. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 544–554, Dublin, Ireland. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019a. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Yixin Liu, Budhaditya Deb, Milagro Teruel, Aaron Halfaker, Dragomir Radev, and Ahmed Hassan Awadallah. 2023c. On improving summarization factual consistency from natural language feedback. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15144–15161.

Yixin Liu, Alex Fabbri, Pengfei Liu, Yilun Zhao, Linyong Nan, Ruilin Han, Simeng Han, Shafiq Joty, Chien-Sheng Wu, Caiming Xiong, and Dragomir Radev. 2023d. Revisiting the gold standard: Grounding summarization evaluation with robust human evaluation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4140–4170.

Yongtai Liu, Joshua Maynez, Gonçalo Simões, and Shashi Narayan. 2022d. Data augmentation for low-resource dialogue summarization. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 703–710.

Zhengyuan Liu, Angela Ng, Sheldon Lee Shao Guang, AiTi Aw, and Nancy F. Chen. 2019b. Topic-aware pointer-generator networks for summarizing spoken conversations. *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 814–821.

Shayne Longpre, Robert Mahari, Anthony Chen, Naana Obeng-Marnu, Damien Sileo, William Brannon, Niklas Muennighoff, Nathan Khazam, Jad Kabbara, Kartik Perisetla, et al. 2023a. The data provenance initiative: A large scale audit of dataset licensing & attribution in ai. *arXiv preprint arXiv:2310.16787*.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7052–7063, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shayne Longpre, Gregory Yauney, Emily Reif, Katherine Lee, Adam Roberts, Barret Zoph, Denny Zhou, Jason Wei, Kevin Robinson, David Mimno, et al. 2023b. A pretrainer's guide to training data: Measuring the effects of data age, domain coverage, quality, & toxicity. *arXiv preprint arXiv:2305.13169*.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2023. Chatgpt as a factual inconsistency evaluator for abstractive text summarization. *arXiv preprint arXiv:2303.15621*.

Gary Lupyan and Rick Dale. 2010. Language structure is partly determined by social structure. *PloS one*, 5(1):e8559.

Kaixin Ma, Hao Cheng, Xiaodong Liu, Eric Nyberg, and Jianfeng Gao. 2022. Open domain question answering with a unified knowledge interface. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1605–1620, Dublin, Ireland. Association for Computational Linguistics.

Andreas Madsen, Siva Reddy, and Sarath Chandar. 2022. Post-hoc interpretability for neural nlp: A survey. *ACM Comput. Surv.*, 55(8).

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

Potsawee Manakul, Adian Liusie, and Mark JF Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Qianren Mao, Jianxin Li, Hao Peng, Shizhu He, Lihong Wang, S Yu Philip, and Zheng Wang. 2022. Fact-driven abstractive summarization by utilizing multi-granular multi-relational knowledge. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:1665–1678.

Ana Marasović, Iz Beltagy, Doug Downey, and Matthew E Peters. 2021. Few-shot self-rationalization with natural language prompts. *arXiv preprint arXiv:2111.08284*.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1506–1515, Copenhagen, Denmark. Association for Computational Linguistics.

Alana Marzoev, Samuel Madden, M. Frans Kaashoek, Michael J. Cafarella, and Jacob Andreas. 2020. Unnatural language processing: Bridging the gap between synthetic and natural language data. *ArXiv*, abs/2004.13645.

Michael L Mauldin. 1984. Semantic rule based text generation. In *10th International Conference on Computational Linguistics and 22nd Annual Meeting of the Association for Computational Linguistics*, pages 376–380.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

Ninareh Mehrabi, Pei Zhou, Fred Morstatter, Jay Pujara, Xiang Ren, and Aram Galstyan. 2021. Lawyers are dishonest? quantifying representational harms in commonsense knowledge resources. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5033.

Kevin Meng, David Bau, Alex J Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. In *Advances in Neural Information Processing Systems*.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation.

Abhika Mishra, Akari Asai, Vidhisha Balachandran, Yizhong Wang, Graham Neubig, Yulia Tsvetkov, and Hannaneh Hajishirzi. 2024. Fine-grained hallucination detection and editing for language models.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *International Conference on Machine Learning*.

Shamima Mithun and Leila Kosseim. 2011. Discourse structures to reduce discourse incoherence in blog summarization. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pages 479–486, Hissar, Bulgaria. Association for Computational Linguistics.

Sayantan Mitra, Roshni Ramnani, and Shubhashis Sengupta. 2022. Constraint-based multi-hop question answering with knowledge graph. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track*, pages 280–288.

Masahiro Mizukami, Koichiro Yoshino, Graham Neubig, David Traum, and Satoshi Nakamura. 2016. Analyzing the effect of entrainment on dialogue acts. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 310–318, Los Angeles. Association for Computational Linguistics.

Leann Myers and Maria J Sirois. 2004. Spearman correlation coefficients, differences between. *Encyclopedia of statistical sciences*, 12.

Moin Nadeem, Anna Bethke, and Siva Reddy. 2021. Stereoset: Measuring stereotypical bias in pretrained language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5356–5371.

Preslav Nakov, David Corney, Maram Hasanain, Firoj Alam, Tamer Elsayed, Alberto Barrón-Cedeño, Paolo Papotti, Shaden Shaar, and Giovanni Da San Martino. 2021. Automated fact-checking for assisting human fact-checkers. *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence Survey Track*.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar GuÌ‡lçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Feng Nan, Cicero dos Santos, Henghui Zhu, Patrick Ng, Kathleen Mckeown, Ramesh Nallapati, Dejiao Zhang, Zhiguo Wang, Andrew O Arnold, and Bing Xiang. 2021. Improving factual consistency of abstractive summarization via question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6881–6894.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Shashi Narayan, Yao Zhao, Joshua Maynez, Gonçalo Simões, Vitaly Nikolaev, and Ryan McDonald. 2021. Planning with learned entity prompts for abstractive summarization. *Transactions of the Association for Computational Linguistics*, 9:1475–1492.

Will Norcliffe-Brown, Stathis Vafeias, and Sarah Parisot. 2018. Learning conditioned graph structures for interpretable visual question answering. In *Advances in neural information processing systems*, pages 8334–8343.

Barlas Oguz, Xilun Chen, Vladimir Karpukhin, Stan Peshterliev, Dmytro Okhonko, Michael Schlichtkrull, Sonal Gupta, Yashar Mehdad, and Scott Yih. 2022. UniK-QA: Unified representations of structured and unstructured knowledge for open-domain question answering. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1535–1546, Seattle, United States. Association for Computational Linguistics.

Eda Okur, Saurav Sahay, and Lama Nachman. 2022. Data augmentation with paraphrase generation and entity extraction for multimodal dialogue system. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4114–4125, Marseille, France. European Language Resources Association.

Yonatan Oren, Nicole Meister, Niladri Chatterji, Faisal Ladhak, and Tatsunori B Hashimoto. 2023. Proving test set contamination in black box language models. *arXiv preprint arXiv:2310.17623*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*.

Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. 2021. Understanding factuality in abstractive summarization with FRANK: A benchmark for factuality metrics. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4812–4829, Online. Association for Computational Linguistics.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Amin Parvaneh, Ehsan Abbasnejad, Qi Wu, and Javen Shi. 2019. Show, price and negotiate: A hierarchical attention recurrent visual negotiator. *CoRR*, abs/1905.03721.

Sajan Patel and Kyle Lam. 2023. Chatgpt: the future of discharge summaries? *The Lancet. Digital health*.

Thomas Pellissier Tanon, Gerhard Weikum, and Fabian Suchanek. 2020. Yago 4: A reason-able knowledge base. In *European Semantic Web Conference*, pages 583–596. Springer.

Xutan Peng, Yi Zheng, Chenghua Lin, and Advaith Siddharthan. 2021. Summarising historical text in modern languages. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3123–3142.

Fernando Pereira and Yves Schabes. 1992. Inside-outside reestimation from partially bracketed corpora. In *30th Annual Meeting of the Association for Computational Linguistics*, pages 128–135.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

Seth Polsley, Pooja Jhunjhunwala, and Ruihong Huang. 2016. Casesummarizer: a system for automated summarization of legal texts. In *Proceedings of COLING 2016, the 26th international conference on Computational Linguistics: System Demonstrations*, pages 258–262.

P. E. Pope, S. Kolouri, M. Rostami, C. E. Martin, and H. Hoffmann. 2019. Explainability methods for graph convolutional neural networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10764–10773.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. 2022. Measuring and narrowing the compositionality gap in language models. *arXiv preprint arXiv:2210.03350*.

Dean G Pruitt. 2013. *Negotiation behavior*. Academic Press.

Jinghui Qin, Zheng Ye, Jianheng Tang, and Xiaodan Liang. 2020. Dynamic knowledge routing network for target-guided open-domain conversation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8657–8664.

Dheeraj Rajagopal, Siamak Shakeri, Cicero Nogueira dos Santos, Eduard Hovy, and Chung-Ching Chang. 2022. Counterfactual data augmentation improves factuality of abstractive summarization. *arXiv preprint arXiv:2205.12416*.

Nazneen Rajani, Lewis Tunstall, Edward Beeching, Nathan Lambert, Alexander M. Rush, and Thomas Wolf. 2023. No robots.

Vinay Venkatesh Ramasesh, Aitor Lewkowycz, and Ethan Dyer. 2021. Effect of scale on catastrophic forgetting in neural networks. In *International Conference on Learning Representations*.

Ahad Rana. 2010. Common crawl – building an open web-scale crawl using hadoop.

Priyanka Ranade, Anupam Joshi, and Tim Finin. 2021. Study shows ai-generated fake reports fool experts.

Ekagra Ranjan, Soumya Sanyal, and Partha P. Talukdar. 2020. ASAP: adaptive structure aware pooling for learning hierarchical graph representations. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 5470–5477. AAAI Press.

Vipula Rawte, Swagata Chakraborty, Agnibh Pathak, Anubhav Sarkar, S.M Towhidul Islam Tonmoy, Aman Chadha, Amit Sheth, and Amitava Das. 2023. The troubling emergence of hallucination in large language models - an extensive definition, quantification, and prescriptive remediations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. 2019. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Pengjie Ren, Zhumin Chen, Zhaochun Ren, Furu Wei, Liqiang Nie, Jun Ma, and Maarten De Rijke. 2018. Sentence relations for extractive summarization with deep neural networks. *ACM Transactions on Information Systems (TOIS)*, 36(4):1–32.

Leonardo F. R. Ribeiro, Mengwen Liu, Iryna Gurevych, Markus Dreyer, and Mohit Bansal. 2022. FactGraph: Evaluating factuality in summarization with semantic graph representations. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3238–3253, Seattle, United States. Association for Computational Linguistics.

Alan Ritter, Colin Cherry, and Bill Dolan. 2010. Unsupervised modeling of Twitter conversations. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 172–180, Los Angeles, California. Association for Computational Linguistics.

Md Rashad Al Hasan Rony, Ricardo Usbeck, and Jens Lehmann. 2022. DialoKG: Knowledge-structure aware task-oriented dialogue generation. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2557–2571, Seattle, United States. Association for Computational Linguistics.

Corby Rosset, Chenyan Xiong, Minh Hieu Phan, Xia Song, Paul Bennett, and Saurabh Tiwary. 2020. Knowledge-aware language model pretraining. *ArXiv*, abs/2007.00655.

Sascha Rothe, Joshua Maynez, and Shashi Narayan. 2021. A thorough evaluation of task-specific pretraining for summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 140–145.

Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. 2020. Leveraging pre-trained checkpoints for sequence generation tasks. *Transactions of the Association for Computational Linguistics*, 8:264–280.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Arkadiy Saakyan, Tuhin Chakrabarty, and Smaranda Muresan. 2021. Covid-fact: Fact extraction and verification of real-world claims on covid-19 pandemic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2116–2129.

Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10776–10787, Singapore. Association for Computational Linguistics.

Sunita Sarawagi and William W Cohen. 2004. Semi-markov conditional random fields for information extraction. *Advances in neural information processing systems*, 17.

Mourad Sarrouti, Asma Ben Abacha, Yassine M'rabet, and Dina Demner-Fushman. 2021. Evidence-based fact-checking of health-related claims. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3499–3512.

Roser Sauri. 2008. *A Factuality Profiler for Eventualities in Text*. Ph.D. thesis, Brandeis University, USA.

Roser Saurí and James Pustejovsky. 2012. Are you sure that this happened? assessing the factuality degree of events in text. *Computational Linguistics*, 38(2):261–299.

Roger C Schank et al. 1970. Spinoza ii: Conceptual case-based natural language analysis.

M. Schlichtkrull, Thomas Kipf, P. Bloem, R. V. Berg, Ivan Titov, and M. Welling. 2018. Modeling relational data with graph convolutional networks. In *ESWC*.

Katherine Schulten. 2023. How should schools respond to chatgpt?

Tal Schuster, Adam Fisch, and Regina Barzilay. 2021. Get your vitamin C! robust fact verification with contrastive evidence. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 624–643, Online. Association for Computational Linguistics.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, Alex Wang, and Patrick Gallinari. 2021. Questeval: Summarization asks for fact-based evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6594–6604.

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

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Omar Shaikh, Hongxin Zhang, William Held, Michael Bernstein, and Diyi Yang. 2023a. On second thought, let's not think step by step! bias and toxicity in zero-shot reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4454–4470.

Omar Shaikh, Hongxin Zhang, William Held, Michael Bernstein, and Diyi Yang. 2023b. On second thought, let's not think step by step! bias and toxicity in zero-shot reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4454–4470, Toronto, Canada. Association for Computational Linguistics.

Siamak Shakeri, Cicero dos Santos, Henghui Zhu, Patrick Ng, Feng Nan, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. 2020. End-to-end synthetic data generation for domain adaptation of question answering systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5445–5460.

Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2021. Societal biases in language generation: Progress and challenges. *arXiv preprint arXiv:2105.04054*.

Weiyan Shi, Tiancheng Zhao, and Zhou Yu. 2019. Unsupervised dialog structure learning. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1797–1807, Minneapolis, Minnesota. Association for Computational Linguistics.

Vivian S Silva, André Freitas, and Siegfried Handschuh. 2019. Exploring knowledge graphs in an interpretable composite approach for text entailment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7023–7030.

Norman K Sondheimer and Ralph Weischedel. 1980. A rule-based approach to ill-formed input. In *COLING 1980 Volume 1: The 8th International Conference on Computational Linguistics*.

Kaiqiang Song, Lin Zhao, and Fei Liu. 2018. Structure-infused copy mechanisms for abstractive summarization. In *COLING*.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 196–205, Denver, Colorado. Association for Computational Linguistics.

Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and Ido Dagan. 2018. Supervised open information extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 885–895, New Orleans, Louisiana. Association for Computational Linguistics.

Chris Stokel-Walker. 2023. Chatgpt listed as author on research papers: many scientists disapprove.

Emma Strubell, Patrick Verga, Daniel Andor, David Weiss, and Andrew McCallum. 2018. Linguistically-informed self-attention for semantic role labeling. *arXiv preprint arXiv:1804.08199*.

Zhiqing Sun, Sheng Shen, Shengcao Cao, Haotian Liu, Chunyuan Li, Yikang Shen, Chuang Gan, Liang-Yan Gui, Yu-Xiong Wang, Yiming Yang, et al. 2023. Aligning large multimodal models with factually augmented rlhf. *arXiv preprint arXiv:2309.14525*.

Shahbaz Syed, Michael Völske, Nedim Lipka, Benno Stein, Hinrich Schütze, and Martin Potthast. 2019. Towards summarization for social media - results of the TL;DR challenge. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 523–528, Tokyo, Japan. Association for Computational Linguistics.

Masato Takatsuka, Tetsunori Kobayashi, and Yoshihiko Hayashi. 2022. Phrase-level localization of inconsistency errors in summarization by weak supervision. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6151–6164.

Leonard Talmy. 1983. *How language structures space*. Citeseer.

Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2017. Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1171–1181.

Yi Chern Tan and L Elisa Celis. 2019. Assessing social and intersectional biases in contextualized word representations. *Advances in Neural Information Processing Systems*, 32.

Jianheng Tang, Tiancheng Zhao, Chenyan Xiong, Xiaodan Liang, Eric Xing, and Zhiting Hu. 2019. Target-guided open-domain conversation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5624–5634, Florence, Italy. Association for Computational Linguistics.

Liyan Tang, Tanya Goyal, Alex Fabbri, Philippe Laban, Jiacheng Xu, Semih Yavuz, Wojciech Kryscinski, Justin Rousseau, and Greg Durrett. 2023. Understanding factual errors in summarization: Errors, summarizers, datasets, error detectors. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11626–11644.

Xiangru Tang, Arjun Nair, Borui Wang, Bingyao Wang, Jai Desai, Aaron Wade, Haoran Li, Asli Celikyilmaz, Yashar Mehdad, and Dragomir Radev. 2022. CONFIT: Toward faithful dialogue summarization with linguistically-informed contrastive fine-tuning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5657–5668, Seattle, United States. Association for Computational Linguistics.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Leigh L Thompson. 2001. *The mind and heart of the negotiator*, volume 3.

James Thorne and Andreas Vlachos. 2018. Automated fact checking: Task formulations, methods and future directions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3346–3359, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Ehsan Toreini, Mhairi Aitken, Kovila PL Coopamootoo, Karen Elliott, Vladimiro Gonzalez Zelaya, Paolo Missier, Magdalene Ng, and Aad van Moorsel. 2020. Technologies for trustworthy machine learning: A survey in a socio-technical context. *arXiv preprint arXiv:2007.08911*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ming Tu, Kevin Huang, Guangtao Wang, Jing Huang, Xiaodong He, and Bowen Zhou. 2020. Select, answer and explain: Interpretable multi-hop reading comprehension over multiple documents. In *AAAI 2020 (accepted)*.

Yi-Lin Tuan, Yun-Nung Chen, and Hung-yi Lee. 2019. DyKgChat: Benchmarking dialogue generation grounding on dynamic knowledge graphs. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1855–1865, Hong Kong, China. Association for Computational Linguistics.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In *Thirty-seventh Conference on Neural Information Processing Systems*.

William Thomas Tutte. 1984. Graph theory, vol. 21 of. *Encyclopedia of Mathematics and its Applications*.

Can Udomcharoenchaikit, Wuttikorn Ponwitayarat, Patomporn Payoungkhamdee, Kanruethai Masuk, Weerayut Buaphet, Ekapol Chuangsuwanich, and Sarana Nutanong. 2022. Mitigating spurious correlation in natural language understanding with counterfactual inference. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11308–11321, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Prasetya Utama, Joshua Bambrick, Nafise Moosavi, and Iryna Gurevych. 2022. Falsesum: Generating document-level NLI examples for recognizing factual inconsistency in summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2763–2776, Seattle, United States. Association for Computational Linguistics.

Shikhar Vashishth, Rishabh Joshi, Sai Suman Prayaga, Chiranjib Bhattacharyya, and Partha Talukdar. 2018. RESIDE: Improving distantly-supervised neural relation extraction using side information. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1257–1266, Brussels, Belgium. Association for Computational Linguistics.

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha Talukdar. 2020. Composition-based multi-relational graph convolutional networks. In *International Conference on Learning Representations*.

Shikhar Vashishth, Naganand Yadati, and Partha Talukdar. 2019. Graph-based deep learning in natural language processing. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): Tutorial Abstracts*, Hong Kong, China. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. *International Conference on Learning Representations*. Accepted as poster.

Alfredo Vellido. 2020. The importance of interpretability and visualization in machine learning for applications in medicine and health care. *Neural computing and applications*, 32(24):18069–18083.

Oriol Vinyals and Quoc Le. 2015. A neural conversational model. *arXiv preprint arXiv:1506.05869*.

Ellen Voorhees, Tasmeer Alam, Steven Bedrick, Dina Demner-Fushman, William R Hersh, Kyle Lo, Kirk Roberts, Ian Soboroff, and Lucy Lu Wang. 2021. Trec-covid: constructing a pandemic information retrieval test collection. In *ACM SIGIR Forum*, volume 54, pages 1–12. ACM New York, NY, USA.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7534–7550, Online. Association for Computational Linguistics.

David Wadden, Kyle Lo, Lucy Lu Wang, Arman Cohan, Iz Beltagy, and Hannaneh Hajishirzi. 2022. MultiVerS: Improving scientific claim verification with weak supervision and full-document context. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 61–76, Seattle, United States. Association for Computational Linguistics.

David Wan and Mohit Bansal. 2022. Factpegasus: Factuality-aware pre-training and fine-tuning for abstractive summarization. *arXiv preprint arXiv:2205.07830*.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020a. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. Association for Computational Linguistics.

Boxin Wang, Weixin Chen, Hengzhi Pei, Chulin Xie, Mintong Kang, Chenhui Zhang, Chejian Xu, Zidi Xiong, Ritik Dutta, Rylan Schaeffer, et al. 2023a. Decodingtrust: A comprehensive assessment of trustworthiness in gpt models.

Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Doug Burdick, Darrin Eide, Kathryn Funk, Yannis Katsis, Rodney Michael Kinney, et al. 2020b. Cord-19: The covid-19 open research dataset. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*.

Wenya Wang and Sinno Pan. 2022. Deep inductive logic reasoning for multi-hop reading comprehension. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4999–5009, Dublin, Ireland. Association for Computational Linguistics.

William Yang Wang. 2017. "liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 422–426, Vancouver, Canada. Association for Computational Linguistics.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194.

Xuewei Wang, Weiyan Shi, Richard Kim, Yoojung Oh, Sijia Yang, Jingwen Zhang, and Zhou Yu. 2019. Persuasion for good: Towards a personalized persuasive dialogue system for social good. *arXiv preprint arXiv:1906.06725*.

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Raghavi Chandu, David Wadden, Kelsey MacMillan, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. 2023b. How far can camels go? exploring the state of instruction tuning on open resources. In *Advances in Neural Information Processing Systems*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023c. Self-Instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*.

Yuxia Wang, Haonan Li, Xudong Han, Preslav Nakov, and Timothy Baldwin. 2023d. Do-not-answer: A dataset for evaluating safeguards in llms. *arXiv preprint arXiv:2308.13387*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Huai hsin Chi, F. Xia, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903.

Wei Wei, Quoc Le, Andrew Dai, and Jia Li. 2018. AirDialogue: An environment for goal-oriented dialogue research. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3844–3854, Brussels, Belgium. Association for Computational Linguistics.

Peter West, Chandra Bhagavatula, Jack Hessel, Jena Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. 2022. Symbolic knowledge distillation: from general language models to commonsense models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4602–4625.

Yuhuai Wu, Felix Li, and Percy Liang. 2022a. Insights into pre-training via simpler synthetic tasks. *ArXiv*, abs/2206.10139.

Yuxiang Wu, Matt Gardner, Pontus Stenetorp, and Pradeep Dasigi. 2022b. Generating data to mitigate spurious correlations in natural language inference datasets. *arXiv preprint arXiv:2203.12942*.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. 2020. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, page 1–21.

Yijun Xiao and William Yang Wang. 2021. On hallucination and predictive uncertainty in conditional language generation. *ArXiv*, abs/2103.15025.

Shangsheng Xie and Mingming Lu. 2019. Interpreting and Understanding Graph Convolutional Neural Network using Gradient-based Attribution Method. *arXiv e-prints*, page arXiv:1903.03768.

Yuexiang Xie, Fei Sun, Yang Deng, Yaliang Li, and Bolin Ding. 2021. Factual consistency evaluation for text summarization via counterfactual estimation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 100–110, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xinnuo Xu, Ondřej Dušek, Jingyi Li, Verena Rieser, and Ioannis Konstas. 2020. Fact-based content weighting for evaluating abstractive summarisation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5071–5081, Online. Association for Computational Linguistics.

Shuo Yang, Wei-Lin Chiang, Lianmin Zheng, Joseph E Gonzalez, and Ion Stoica. 2023. Rethinking benchmark and contamination for language models with rephrased samples. *arXiv preprint arXiv:2311.04850*.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

Shaowei Yao and Xiaojun Wan. 2020. Multimodal transformer for multimodal machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4346–4350, Online. Association for Computational Linguistics.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik R Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Michihiro Yasunaga, Antoine Bosselut, Hongyu Ren, Xikun Zhang, Christopher D. Manning, Percy Liang, and Jure Leskovec. 2022. Deep bidirectional language-knowledge graph pretraining. *ArXiv*, abs/2210.09338.

Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. 2021. Qa-gnn: Reasoning with language models and knowledge graphs for question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 535–546.

Ji Soo Yi, Rachel Melton, John Stasko, and Julie A. Jacko. 2005. Dust & magnet: Multivariate information visualization using a magnet metaphor. *Information Visualization*, 4(4):239–256.

Xunjian Yin, Baizhou Huang, and Xiaojun Wan. 2023. ALCUNA: Large language models meet new knowledge. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1397–1414, Singapore. Association for Computational Linguistics.

Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec. 2018. Hierarchical graph representation learning with differentiable pooling. In *Advances in neural information processing systems*, pages 4800–4810.

S. Young. 2006. Using pomdps for dialog management. In *2006 IEEE Spoken Language Technology Workshop*, pages 8–13.

Wenhao Yu, Meng Jiang, Zhiting Hu, Qingyun Wang, Heng Ji, and Nazneen Rajani. 2021. Knowledge-enriched natural language generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts*, pages 11–16, Punta Cana, Dominican Republic & Online. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2020. Defending against neural fake news. *Neurips*.

Ke Zhai and Jason D. Williams. 2014. Discovering latent structure in task-oriented dialogues. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 36–46, Baltimore, Maryland. Association for Computational Linguistics.

Die Zhang, Hao Zhang, Huilin Zhou, Xiaoyi Bao, Da Huo, Ruizhao Chen, Xu Cheng, Mengyue Wu, and Quanshi Zhang. 2021. Building interpretable interaction trees for deep nlp models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14328–14337.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020a. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018a. Personalizing dialogue agents: I have a dog, do you have pets too? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213, Melbourne, Australia. Association for Computational Linguistics.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona 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. 2022a. Opt: Open pre-trained transformer language models. arXiv.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Bertscore: Evaluating text generation with BERT. In *Proceedings of the 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Wenqian Zhang, Shangbin Feng, Zilong Chen, Zhenyu Lei, Jundong Li, and Minnan Luo. 2022b. KCD: Knowledge walks and textual cues enhanced political perspective detection in news media. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4129–4140, Seattle, United States. Association for Computational Linguistics.

X Zhang, A Bosselut, M Yasunaga, H Ren, P Liang, C Manning, and J Leskovec. 2022c. Greaselm: Graph reasoning enhanced language models for question answering. In *International Conference on Representation Learning (ICLR)*.

Xingxing Zhang, Mirella Lapata, Furu Wei, and Ming Zhou. 2018b. Neural latent extractive document summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 779–784, Brussels, Belgium. Association for Computational Linguistics.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*.

Zhen Zhang, Jiajun Bu, Martin Ester, Jianfeng Zhang, Chengwei Yao, Zhi Yu, and Can Wang. 2019. Hierarchical graph pooling with structure learning. *arXiv preprint arXiv:1911.05954*.

Zheng Zhao, Shay B Cohen, and Bonnie Webber. 2020. Reducing quantity hallucinations in abstractive summarization. *arXiv preprint arXiv:2009.13312*.

Hao Zhou, Tom Young, Minlie Huang, Haizhou Zhao, Jingfang Xu, and Xiaoyan Zhu. 2018. Commonsense knowledge aware conversation generation with graph attention. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, page 4623–4629. AAAI Press.

Kun Zhou, Yutao Zhu, Zhipeng Chen, Wentong Chen, Wayne Xin Zhao, Xu Chen, Yankai Lin, Ji-Rong Wen, and Jiawei Han. 2023. Don't make your llm an evaluation benchmark cheater. *arXiv preprint arXiv:2311.01964*.

Yiheng Zhou, He He, Alan W Black, and Yulia Tsvetkov. 2019. A dynamic strategy coach for effective negotiation. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 367–378, Stockholm, Sweden. Association for Computational Linguistics.

Yiheng Zhou, Yulia Tsvetkov, Alan W Black, and Zhou Yu. 2020. Augmenting non-collaborative dialog systems with explicit semantic and strategic dialog history. In *International Conference on Learning Representations*.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.