Four Novel Ways Toward Better Factuality of Abstractive Text Summarization

by **Jianbin Shen**

Thesis submitted in fulfilment of the requirements

for the degree of

Doctor of Philosophy

in

Computer Science

under the supervision of

Christy Jie Liang and Junyu Xuan

University of Technology Sydney

Faculty of Engineering and Information Technology

July 2024

CERTIFICATE OF ORIGINAL AUTHORSHIP

I, Jianbin Shen, declare that this thesis is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the School of Computer Science, Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

SIGNATURE: 

[Jianbin Shen]

DATE: 17th July, 2024

PLACE: Sydney, Australia

Abstract

Abstractive text summarization is a form of deep learning-based generative modeling research in natural language processing. Its goal is to develop models and methods that automatically condense long documents into summaries that are fluent, coherent, relevant, and consistent. It is an important research in the Big Data era, which demands advanced models and methods to turn voluminous and often long text data into concise but informative and factual summaries or abstracts for efficient human consumption.

Abstractive text summarization research has made considerable progress in recent years. This achievement is attributed to the advancement of several research frontiers, especially encoder-decoder modeling, language modeling, and numerous task-specific methods. Even so, abstractive text summarization still faces challenges, particularly when factual issues, often referred to as hallucinations, incurred in the model-generated summaries interfere with its adaptation in real-world applications. Given its importance, researchers have developed a wide range of novel methods to address various factual issues with some success, but some factual issues remain untouched or less explored. There is also room for developing more effective methods.

The research recorded in this thesis was aimed at improving the factuality of abstractive text summarization by investigating a range of related problems, including undesirable (and ungrammatical) word repeats, distorted sub-phrasal hallucinations, endophoric reference errors, intrinsic named entity-related hallucinations, and factual informativeness issues. To do this, four novel methods, including determinantal point process-based sampling with self-critical reinforcement learning, syntactic structure-aware semantic learning, entity alignment learning facilitated by adaptive margin ranking loss, and optimal transport-based informative attention guided by named entity salience, were explored and extensive experiments were conducted on them.

Extensive experiments have been conducted to verify the proposed methods respectively, including quantitative evaluations and qualitative assessments, with benchmark datasets. The experimental results have shown the efficacy of the proposed methods in tackling observed factual issues and hallucinations compared to respective baselines. The research has further provided insightful analyses of the

evaluation and assessment results. We believe that the proposed methods and result analyses contribute to the knowledge of abstractive text summarization research. The understanding acquired from the insights would benefit future research.

Keywords

Abstractive text summarization (ATS), natural language processing (NLP), deep learning (DL), encoder-decoder, determinantal point process (DPP), self-critical reinforcement learning, syntactic structure-aware representation, graph neural network (GNN), semantic similarity learning, adaptive margin ranking loss, optimal transport, Wasserstein (or Kantorovich) distance, accumulative joint entropy reduction.

Dedication

To myself . . .

ACKNOWLEDGMENTS

First and foremost, I would like to express my deepest gratitude to my supervisors, Associate Professor Christy Jie Liang and Dr. Junyu Xuan. They have dedicated countless hours to assisting me with their invaluable advice, continuous support, and patience throughout my PhD candidature.

It is a long and challenging journey to acquire my PhD in a deep learning-based artificial intelligence research field, abstractive text summarization. My principal supervisor, Associate Professor Christy Jie Liang, has constantly encouraged me to pursue this research journey. She granted me freedom and guided me to explore the research field in which I have found the topic of my passion. During her maternity leave, Dr. Junyu Xuan, my co-supervisor, came to my aid. His extended and deep knowledge of probability theory and statistical methods has sharpened my knowledge in the research field. His empirical research experience in deep learning has further improved my skills in research experiments and analyses. My supervisors have also devoted their much valued time to critiquing my work for high-quality outcomes. I have learned a great deal of research methods and academic writing skills from my supervisors over the last few years. Their rigorous research attitude has shaped my research perspective beyond just developing novel ideas and has transformed me from my software engineering experience to the acquisition of an academic research capability. This research would not have been fruitful without their mentorship.

I would also like to thank the University of Technology Sydney (UTS) Australia for providing a flexible, inspiring, and innovative research program and environment, along with its unwavering support of my research topic. Additionally, I do not forget my gratitude to the UTS eResearch High-Performance Computer Cluster centers for providing computational resources and their prompt technical support and service.

Finally, I would like to acknowledge that this research is supported by an Australian Government Research Training Program Scholarship.

Jianbin Shen

Sydney, Australia, 2024

List of Publications

Related to the Thesis :

1. Jianbin Shen, Junyu Xuan, and Christy Liang (2023). "A Determinantal Point Process Based Novel Sampling Method of Abstractive Text Summarization". 2023 International Joint Conference on Neural Networks (IJCNN). IEEE, pp. 1-8.

2. Jianbin Shen, Junyu Xuan, and Christy Liang (2023). "Mitigating Intrinsic Named Entity-Related Hallucinations of Abstractive Text Summarization". Findings of the Association for Computational Linguistics: EMNLP 2023 A*, pp. 15807-15824.

3. Jianbin Shen, Christy Jie Liang, Junyu Xuan (2024). "Improving the Factuality of Abstractive Text Summarization with Syntactic Structure-Aware Latent Semantic Space". 2024 International Joint Conference on Neural Networks (IJCNN). (Accepted and Presentation in July 2024).

4. Jianbin Shen, Christy Jie Liang, Junyu Xuan. "InforME: Improving Informativeness of Abstractive Text Summarization With Informative Attention Guided by Named Entity Salience". Submission to IEEE/ACM Transaction on Audio Speech and Language Processing (TASLP). (Under Review).

Table of Contents

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

1.1 Background

Abstractive text summarization (ATS) has become an advanced form of deep learning-based research in natural language processing (NLP) over the last decade. As first suggested by Kryscinski et al. (2019), fluency, coherence, relevance, and consistency are now the commonly accepted criteria for ATS.

Deep learning-based ATS has manifested its importance in the Big Data era where voluminous and often long text data is beyond the capacity of manual processing. To turn gigantic amounts of unstructured text data into valuable, informative assets, researchers have developed advanced models and methods that turn such data into concise, factual knowledge for efficient human consumption. Beyond this, advanced ATS models and methods may also lay the foundations for many other real-world applications such as document analysis and text categorization.

Text summarization has evolved from extractive text summarization (e.g., Zhong et al., 2019; Zhong et al., 2020) to ATS (e.g., Qi et al., 2020; Liu and Liu, 2021). The former generates a summary by paraphrasing the extracted key phrases and sentences from a source document; the latter is of free form, analogous to how humans abstract documents by forming novel phrases and sentences. Therefore, ATS research is more challenging and important for real-world applications. ATS may also give rise to hybrid approaches (e.g., Dong et al., 2019; Zhang et al., 2020a).

Since the advent of deep learning, many fundamental research areas in NLP have progressed considerably, examples being word embeddings (e.g., Mikolov et al., 2013; Pennington et al., 2014; Devlin et al., 2019), word encoding schemes (e.g., Sennrich et al., 2016; Wu et al., 2016; Kudo and Richardson, 2018), feature extraction models (e.g., Hochreiter and Schmidhuber, 1997; Vaswani et al., 2017), and generative

modeling architectures (e.g., Sutskever et al., 2014; Vaswani et al., 2017; Radford et al., 2019). The advancement of these research frontiers has been essential to the progress of ATS research. The recent innovative language modeling (e.g., Radford et al., 2018; Devlin et al., 2019; Lewis et al., 2020), often referred to as the pre-training phase, has enabled effective transfer learning, which has led to ATS research breakthroughs in terms of improved quality on the aforementioned four criteria (fluency, coherence, relevance, and consistency). Furthermore, numerous task-specific methods (e.g., Zhang et al., 2022b; Lu et al., 2020; Gabriel et al., 2021), developed to fine-tune the pre-trained language models, have adapted ATS to data domains such as news articles and scientific papers. This is also known as the fine-tuning phase.

Despite its recent successes, ATS still faces some challenging issues. Among them, factual issues or hallucinations[1] (Maynez et al., 2020) in model-generated summaries have become the bottleneck problem for its adaptation in real-world applications; for example, a factually erroneous news summary could have severe economic or political repercussions, depending on circumstances. Hallucinations have many forms and their underlying causes are both complex and complicated, examples being linguistic ambiguity inherent in languages (e.g., Sukthanker et al., 2020; Aina et al., 2019), document-summary distributional discrepancy (e.g., Lu et al., 2020), sample data bias (e.g., Lee et al., 2022b), model capacity limitations (e.g., Hahn, 2020; Maynez et al., 2020), training method-induced bias (e.g., Ranzato et al., 2016), along with the constraints resulting from the limited amount of training samples.

Much research has been dedicated to addressing hallucinations by targeting their various causes (e.g., Lukasik et al., 2020; Lee et al., 2021; Liu et al., 2022a) and erroneous phenomena (e.g., Xiao and Carenini, 2023; Bi et al., 2021; Chen et al., 2021). Even so, some factual issues have been left untouched or not fully addressed. This research has observed several factual issues in model-generated summaries and contributed novel solutions to tackle them.

1.2 Research Questions

- Existing ATS models learn attentive correlations among words and generative word likelihood estimations induced on stochastic samples. They tend to show probability distribution mode concentration. Thus, the words of high likelihood may be repeated undesirably and often ungrammatically in the summary generation. This phenomenon not only leads to less comprehensible summaries but can also compromise the factuality of ATS by missing factual information and

[1]The rest of the thesis uses 'factual issues' and 'hallucinations' interchangeably.

even altering the facts. **How can the distribution mode concentration be tackled to mitigate the undesirable word repeats?**

- Beyond generating undesirable word repeats, existing ATS models may also produce summaries incurring distorted sub-phrasal hallucinations (DSPHs) and endophoric reference errors. One plausible cause of such problems may be that the cross entropy (CE)-based maximum likelihood estimation (MLE) objective is not optimally matched for conditional probabilistic modeling. It may also be that the models based on attentive correlation learning with likelihood estimation have a limited capacity to capture complex syntactic structures and sparse long-distance syntactic relations. These problems have a different symptomatic nature from the word repeats and require different solutions. **How can syntactic structures and relations be more effectively captured to address structural and relational hallucinations?**

- A long document often contains many named entities across multiple contexts. The named entities usually interact with each other. As observed in the research conducted for this thesis, ATS models often mistake these entities one for another with their contexts. Furthermore, their reference relations can also be mistaken, such as in cases involving endophoric reference errors. These named entity-related hallucinations (NERHs) pose even more complex contextual learning conditions than just syntactic issues. **How can named entities be aligned properly with their contexts to avoid entity-related hallucinations?**

- Existing ATS modeling centers learning around source document relevance such that the learned models generate summaries relevant to source documents. This is essential, but learning summaries relevant to source documents may not always be optimally informative from a reference summary perspective because they can miss out on the acquisition of informative facts in reference summaries during training. Thus, model-generated summaries can be source-relevant but uninformative. **How can summarization be improved to capture relevant and informative facts from reference summaries more effectively?**

1.3 Thesis Organization

This section outlines the organization of this thesis. The four research method chapters are outlined in Figure 1.1, and define the research objectives for the respective research questions.

- Chapter 2 It provides a broad review of prior research addressing ATS factual issues. This review lays a solid knowledge foundation for this research.

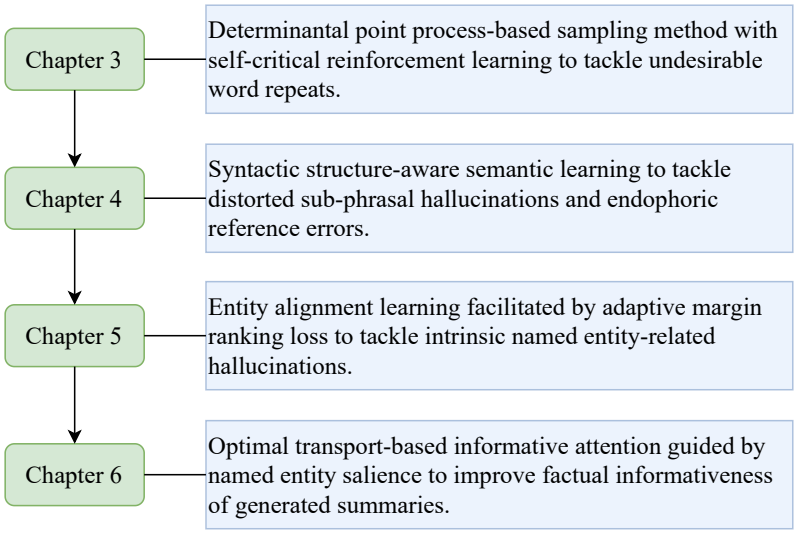


Figure 1.1: The outline of four research method chapters.

- Chapter 3 It presents a determinantal point process-based novel sampling method, accompanied by a self-critical reinforcement learning method, to tackle undesirable word repeat problems. Experiment results and analysis are provided thereafter to demonstrate the capacity of the proposed methods to mitigate such problems.

- Chapter 4 It presents novel syntactic structure-aware semantic learning consisting of syntactic structure-aware semantic similarity regression to tackle distorted sub-phrasal hallucinations, and coreference resolution-based margin ranking regression to reduce endophoric reference errors. The chapter then presents experimental results and analysis to verify the strengths of the proposed learning tasks.

- Chapter 5 It presents an adaptive margin ranking loss, a novel extension to the classical margin ranking loss. Utilizing the loss function, the chapter further presents two entity alignment methods to tackle intrinsic named entity-related hallucinations: (1) the entity-sentence alignment method to mitigate entity-entity hallucinations, and (2) the entity-reference alignment method to mitigate entity-reference hallucinations. Experiment results and analysis are then detailed to confirm the efficacy of the proposed methods.

- Chapter 6 It presents an optimal transport-based informative attention method guided by named entity salience to improve the factual informativeness of ATS. A novel accumulative joint entropy reduction method on named entities is developed to improve named entity salience. Experiment results and analysis are also presented to demonstrate the improvement in factual informativeness achieved by the proposed methods.

- Chapter 7 It summarizes the knowledge gained and the work contributed through this research. The chapter concludes with recommendations for future research directions.

CHAPTER 2

LITERATURE REVIEW

The hallucinations incurring in model-generated summaries have many forms, and their underlying causes are both complex and complicated. This chapter lays the foundation for this study of ATS factuality by reviewing the research literature on the topic.

2.1 Improvement of Data Quality

Imperfect data is an obvious cause of learning models' vulnerability to noisy or imbalanced data distributions and nonfactual outliers. Researchers have shown that training models on improved data can boost performance. For example, Lee et al. (2022b) found that the existence of many similar samples in datasets resulted in the overfitting of models to these samples. By model-training on the de-duplicated datasets using techniques such as exact substring matching and hash-based approximating, the authors subsequently showed improvement in the model's performance. Untruthful samples in datasets are also problematic to learning models. To tackle this issue, Matsumaru et al. (2020) developed a binary entailment classifier to filter out untruthful samples. Consequently, the model trained on the improved data produced better evaluation results. In addition to data filtering approaches, Adams et al. (2022) proposed to rewrite unsupported reference summaries for smaller corpora that may not afford further reduction of dataset size from sample filtering.

Although useful, these data cleaning methods are limited by the amount of data and the granular levels that can be purified on a budget. Additionally, data imperfections such as linguistic ambiguity inherent in languages (e.g., Sukthanker et al., 2020; Aina et al., 2019) and inherent document-summary distributional discrepancy (e.g., Lu et al., 2020) may make data improvement difficult, if not impossible.

2.2 Improvement of Model Capacity

Given that the uncertainty inherent in data quality may not be irreducible, researchers have developed advanced models that are robust to imperfect data. Encoder-decoders, the architecture first proposed by Sutskever et al. (2014), have prevailed in the research field for generative NLP tasks, including ATS. Encoder-decoders evolved from models (e.g., Schuster and Paliwal, 1997; Song et al., 2018) derived from Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) to more recent high-capacity models (e.g., Qi et al., 2020; Zhang et al., 2020a) based on Transformer (Vaswani et al., 2017). Early LSTM-derived models suffered from a limited capacity for learning long text sequences. One of the primary causes is the 'gradient vanishing' issue (Informatik et al., 2003) due to the recurrent model construction. On the other hand, the Transformer-based models, constructed on pairwise correlational learning and global attention mechanisms, have greatly lessened the impact of this obstacle and become widely adopted for ATS and other generative tasks.

Although the Transformer-based models have become the de facto standard, some researchers, for example Kovaleva et al. (2019), found that there are task-agnostic redundancies among Transformer-based attention heads, and other researchers like Clark et al. (2019) discovered the high attention given to less-informative but frequent tokens (e.g., special markers and stop words). These structural deficiency issues caused concerns for the connection to hallucination phenomena and led researchers (e.g., Li et al., 2018; Ghazvininejad et al., 2022; Li et al., 2023) to explore better model structural utilization.

2.3 Training Effectiveness

The studies of Hahn (2020) and Maynez et al. (2020) indicated that models have learning capacity limitations. Hence, the development of effective training methods is crucial for fully unlocking the learning capacity within model limitations. This has been the main focus of the research field in recent years, and numerous methods have been developed to this end. Furthermore, to tackle the various forms of hallucinations and underlying complicated causes, learning methods are often combined with and/or extended to other methods. The rest of this section focuses on prior work closely concerning the factuality of abstractive text summarization and characterizes them according to their main purposes.

2.3.1 Random Mask-Based Language Modeling

Likelihood estimation induced on stochastic sampling is a common approach of stochastic gradient descent-based optimization in supervised learning. However,

recent innovative language modeling of Transformer-based models, often referred to as the pre-training phase, has set a benchmark for transfer learning due to the impressive performance achievements of many downstream generative tasks in NLP, including ATS. The achievements are attributable to the fact that language models pre-trained on large data-diverse corpora, such as Wikipedia (Foundation) and news articles (e.g., Narayan et al., 2018), and are thus encoded with broad prior knowledge. However, annotating labels for supervised learning on such a large amount of diverse samples is neither feasible nor practical. Instead, researchers have developed ingenious random mask-based unsupervised (or self-supervised) learning methods on unlabeled data to learn language models.

"BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" (BERT) (Devlin et al., 2019), "Improving Language Understanding by Generative Pre-Training" (GPT) (Radford et al., 2018), and "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension" (BART) (Lewis et al., 2020) proposed some of the most influential random mask-based learning methods. BERT treated a randomly sampled input token using three replacement options on predefined frequencies: either with the predefined mask token; with a random token; or with no change. Devlin et al. (2019) further developed a next-sentence entailment learning, using a preceding sentence to classify the entailment of the following sentence that may be replaced by a randomly sampled sentence for a percentage of times. In contrast, GPT proposed next-token prediction, similar to the autoregressive decoding process but on an unlabelled source corpus, within a random context window, and with a conditional logarithmic likelihood learning objective. BART further generalized the BERT encoder and the GPT decoder with five denoising techniques on source inputs, namely, masking randomly sampled text spans (either a mask per token or a mask per span), inserting masks at random positions, deleting randomly sampled spans, permuting sentences, and shifting documents by prepending a randomly sampled token. BERT, GPT, and BART also represent three typical types of models: encoder-only, decoder-only, and encoder-decoder. They have become the most adopted backbones for ATS, particularly encoder-decoders.

The general-purpose and highly capable pre-trained language models have significantly improved the generalization of model latent space with their broad prior knowledge and strengthened sequence structure. ATS modeling fine-tuned on the pre-trained language models has benefited from the knowledge transfer and significantly reduced factual issues in model-generated summaries. However, some hallucinations still present a challenge to ATS. Systematic evaluations by Falke et al. (2019) and Maynez et al. (2020) found a considerable number of factual issues

in summaries produced by some state-of-the-art models. Not even the most recently celebrated ChatGPT is immune to making factual errors in ATS (Gao et al., 2023). Over recent years, numerous fine-tuning methods have been proposed to address the challenges of hallucinations.

2.3.2 Factual Consistency-Focused Methods

One way to reduce hallucinations would be to frame them as a problem of general factual consistency between source documents and model-generated summaries. To improve general coherence, Gabriel et al. (2021) proposed a multi-objective discriminator to select the factual and coherent candidate summaries generated by a decoder-only ATS model. Zhou et al. (2021a) trained a model on token-level hallucination detection loss as a sequence-labeling problem. King et al. (2022) developed a beam search that constrained generated summaries to be supported by the source documents, whereas Sridhar and Visser (2022) improved on beam search using a natural language inference (NLI)-enhanced re-ranking method to test entailment relations between a source document and the model-generated summary candidates. To improve models' generation of summaries supported by the source documents (a.k.a. faithfulness), Wang et al. (2022) proposed a reverse generative process to produce realistic negative samples following a back-translation-style approach. Their key idea is to generate unfaithful documents from summaries (reference and augmented negative samples) and thereafter to generate unfaithful summaries (i.e., realistic negative samples) from unfaithful documents. Wang et al. (2022) then trained their model using controlled generation that prepends different control codes to positive and negative samples, respectively.

Observing token uncertainty surge at the beginning of a hallucination and hallucinated tokens dissimilar to the rest of the tokens in a generated summary in model latent space, Liu et al. (2024) formulated thresholded hallucination detection criteria and a correction method that selected the alternative token candidates through a back-tracking algorithm. van der Poel et al. (2022) also directly targeted the uncertainty reduction of models by recognizing that the uncertainty captured in models is a source of problematic conditional generation. The authors explored their solution from mutual information theory and proposed a thresholded conditional pointwise mutual information (CPMI) decoding strategy to reduce the models' uncertainty in probabilistic estimations through training. Contemplating hallucinations concerning faithfulness due to model-encoded domain knowledge influence, Chae et al. (2024) extended the CPMI scoring formulation of van der Poel et al. (2022) by introducing domain keyword prompts on which generative probability estimation terms are conditioned. While these authors thresholded the entropy of conditional probability distribution as a penalty term to regularize the conditional likelihood

prediction in training, Pernes et al. (2022) developed a re-ranking approach based on their energy-based MLE learning objective on a chosen set of estimated summary candidates per source document. On the other hand, by leveraging the total variation distance measure of probability distributions on its robustness to outliers, Ji et al. (2022) developed a learning objective of total variation distance between the model-estimated distribution and a proxy distribution derived from interpolating the one-hot encoded golden labels.

These methods may be considered as making improvements at the macro level and thus might be limited to capturing fine-grained aspects of factuality to reduce various forms of generated hallucinations, particularly in a large corpus of long documents.

2.3.3 Contrastive Learning-Based Methods

Contrastive learning is well adapted to addressing the various factual issues typically caused by exposure bias (e.g., Sun and Li, 2021; Lee et al., 2021) and sample imbalance (e.g., Cao and Wang, 2021; Wu et al., 2022). The idea is to train a model using contrastive learning objectives to distinguish positive (or correct) samples against many negative (or incorrect) samples in context, and thereafter improve the learned model on generative inference. The effectiveness of contrastive learning largely depends on the sampling of negative samples. Cao and Wang (2021) proposed multiple negative sampling methods, which can be categorized as entity/relation text replacement, masked generation, prompted regeneration, and generative entity confidence threshold approaches. It is worth noting that the authors also used back-translation to generate and enrich variant positive samples for learning. Differing from the conventional negative sampling from off-target data samples, adversarial perturbation approaches (e.g., Lee et al., 2021; Wu et al., 2022) created negative samples in model latent space by adding small perturbations to the target summary latent states.

The progress in summarization evaluation research has also inspired new negative sampling approaches for contrastive learning. For example, Sun and Li (2021) first created their negative samples using low-quality summary candidates from beam search measured by ROUGE (Lin, 2004) at inference before fine-tuning their model, whereas Wan and Bansal (2022) combined ROUGE and FactCC (Kryscinski et al., 2020) scores to pre-train their ATS model, and they subsequently fine-tuned the model by contrastive learning using hallucination-corrected summaries and mask-perturbed summaries.

Meanwhile, Liu et al. (2022b) pointed out that the standard MLE learning objective in training did not align well with the estimated probability assignment at inference. To address this mismatching problem, the authors introduced a contrastive learning

objective to complement the MLE in training. The authors first scored and ranked the model-generated summary candidates by a length-normalized log-probability estimation, and then formulated the contrastive learning with the ranked candidates to fine-tune the model.

2.3.4 Syntactic Structure-Aware Learning-Based Methods

Existing ATS modeling prevalently hinges on correlational learning formulation. One of the major challenges to correlational learning-based models is the difficulty in inducing complex syntactic structure knowledge, which is often crucial to addressing factual issues. To acquire such structure knowledge, correlational learning may, to some extent, require extensively large datasets of diverse samples, as evidenced in language modeling trained with multiple large and diverse corpora.

Due to the maturing NLP parsing tools (e.g., Manning et al., 2014; Qi et al., 2018), the functionally rich set of syntactic structures, for example, syntactic dependency (SD) labels, can be acquired cost-efficiently for learning tasks. This has inspired the development of innovative structure-aware methods to tackle ATS factual issues. For instance, SD labels may be transformed into feature embeddings combined with word embeddings as model inputs (e.g., Song et al., 2018; Zhou et al., 2021b). The feature embeddings can also be combined with the latent states of model hidden layers (Song et al., 2018). On the other hand, sparse and directed dependency structures (e.g., SD trees) are commonly represented by graphs. Thus, graph neural networks have been widely adopted to learn the relational representations of such structures (e.g., Fernandes et al., 2019; Liu et al., 2022a). Syntactic structures are also utilized for data reformation. For example, coreference resolution (CR) has been used to cluster concepts in documents to form new data (e.g., Li and Zhuge, 2021; Liao et al., 2018). Additionally, the categorical characteristic of the SD labels has led to the formulation of learning objectives on multi-class label classifications (e.g., Zhou et al., 2021b).

Taking a different step, Song et al. (2020) introduced a syntactic tree decoding strategy accompanying summary decoding in joint training. The authors aimed to generate summaries true to original summaries semantically and grammatically and consequently mitigate hallucinations. They incrementally linearized the decoding of partial syntactic dependency parsing trees onto a stack as part of the autoregressive decoding process. To learn the linearized tree decoding, the authors introduced log-likelihood estimation of parsing tree operations in addition to the standard learning objective.

2.3.5 Entity-Aware Learning-Based Methods

Entity-related hallucinations have been observed as one of the main factual issues (e.g., Goyal and Durrett, 2021). As entities can coexist in multiple contexts and

interact with each other in a long document, entity-related hallucinations pose even more difficult challenges than syntactic issues. Researchers have developed various entity-aware methods to address such challenges. One idea is to encode entity-aware representations to improve latent expressiveness for learning. For example, Bi et al. (2021) encoded contextual representations of named entities using a graph neural network and used them to devise an entity-aware decoding attention mechanism. To improve named entity matches between the generated summaries and the source documents, Xiao and Carenini (2023) proposed a named entity span copy from source documents based on generative likelihood estimation with a global relevance classification task on summary-worthy entities. For the aim of reducing extrinsic entity hallucinations, Nan et al. (2021) introduced a summary-worthy entity classification on the entities occurring in both the source documents and reference summaries, while Zhang et al. (2022a) proposed an entity coverage control method by prepending an entity coverage precision value to the encoder's inputs as a guiding signal. To improve entity specificity in summaries, Narayan et al. (2021) also introduced a guiding signal approach by chaining named entities from the reference summary and prepending them to the decoding inputs. To address named entity-relation hallucinations, Lyu et al. (2022) devised entity-relation generators with log likelihood-based entity-consistency and relation-consistency learning objectives. Observing the source-summary entity aggregation phenomenon in which the named entities are replaced by scoped descriptions that are more general, González et al. (2022) further fine-tuned a pre-trained ATS model using three techniques - aggregation masking, chained aggregation prompting, and aggregation containment classification - to ensure aggregations in summaries are factually aligned with the aggregated entities. Utilizing named entity recognition, Zhao et al. (2020) addressed various forms of quantity-related hallucinations (e.g., numbers, currencies), by ranking beam-searched summary candidates on quantity consistency with the source document.

2.3.6 Extrinsic Knowledge Learning-Based Methods

Given that pre-trained language models encode broad prior knowledge, model-generated summaries often contain extrinsic knowledge that is not evidenced by their source documents. Summaries with extrinsic knowledge are referred to as unfaithfulness. Some methods discussed earlier aim at reducing the likelihood of generating extrinsic entities in summaries as a way to alleviate unfaithfulness. However, unfaithfulness does not automatically imply factual errors. The extrinsic knowledge can be factual in the summarized context as Cao et al. (2022a) studied. Some researchers consider extrinsic but factual knowledge useful for enriching summaries. Thus, their approaches are more concerned with the reduction of non-

factual extrinsic errors while allowing extrinsic but factual information. To this end, Dong et al. (2022) proposed a pipeline that extracted external knowledge graphs (from the Wikidata knowledge base) connected to the entities in the training dataset, then masked out the entities in a model-generated summary, followed by a revision model to replace the masked entities based on the source document and external knowledge graphs. Different from the correction approach by Dong et al. (2022), Gao et al. (2022) enriched source document expressiveness with external knowledge graphs for improving the factuality of model-generated summaries. The authors obtained external knowledge graphs (from Wikipedia) linked to the source document entities. They fused them into source document representations using the ERNIE model (Zhang et al., 2019) by taking advantage of its internal knowledge encoder. Although using the external knowledge base Wikidata, Gunel et al. (2020) formulated them in a margin ranking loss to distinguish positive (factual) entity relations from negative (false) entity relations. In contrast, Cao et al. (2022a) developed a detection method for learning to distinguish factual from non-factual extrinsic entities by external resources such as Wikipedia and Google Search. Their method trained a K-nearest neighbors-based discriminator based on an entity's prior probability learned by a masked language model and its posterior probability learned by an encoder-decoder.

2.3.7 Reinforcement Learning-Based Methods

Reinforcement learning iteratively optimizes actions through trial and error to reach the best possible solution, and it has been successfully applied to many real-world applications (e.g., Google's AlphaGo). ATS researchers have also explored reinforcement learning to tackle factual issues of ATS. For example, Gunasekara et al. (2021) took a reinforcement learning approach using proximal policy optimization to tackle factual issues in generated summaries. They utilized a question-answering generation engine to generate question-answer pairs from ATS-model-generated summaries. Question-answer pairs were also generated from reference summaries by the same generation engine. Additionally, they used an answer-generation model to generate answers by the reference summaries, with questions generated from the corresponding generative summaries. By the same model, the answers were also generated by the ATS-model-generated summaries, with questions generated from the reference summaries. To obtain rewards, the authors used the Normalized Levenshtein distance to score the textual similarity between the paired-up answers from the question-answering generation engine and the similarity between the paired-up answers from the answer-generation model.

In contrast, Celikyilmaz et al. (2018) formulated a self-critical reinforcement learning method whereby a sampling agent formulated on differential ROUGE

metrics is used against the baseline critic agent based on maximum likelihood estimations. Evaluation metrics like ROUGE measure surface or token-level forms, and therefore have limitations in capturing factual semantics. Recognizing the limitation of the ROUGE-based rewards in measuring factuality, Zhang et al. (2020b) complemented ROUGE rewards with factual correctness rewards derived from the accuracy scores between facts extracted from the reference summaries and facts from the model-generated summaries.

The aforementioned limitations of surface-level reward functions may suboptimize reinforcement learning. On the other hand, the model's latent space encodes semantics. This provides an alternative venue to develop reward functions. For example, Jang and Kim (2021) used Word Mover Distance to formulate rewards in embedding space, where rewards were derived from the cost of moving one document distribution to the other. Roit et al. (2023) explored NLI to derive entailment rewards between source documents and generated summaries in the NLI model's latent space. Alternative to the entailment-based rewards, Tang et al. (2023b) used the question-answering (QA) model-based QAFactEval evaluation metric for formulating reward function.

2.3.8 Prompt-Aware Learning-Based Methods

The recent success of using prompts in large language modeling (e.g., Brown et al., 2020) has attracted attention to the development of prompt-aware ATS for improving factuality (e.g., Ghazvininejad et al., 2022; Chen et al., 2023). Different from the prepending approach discussed earlier (e.g., Zhang et al., 2022a), these prompt-aware approaches, also known as prefix-tuning (Li and Liang, 2021) or prompt-tuning (Lester et al., 2021), utilize the pre-trained language models to enable few-shot and zero-shot learning, incorporated with prefix-tuning that prefixes extra trainable parameters to a parameter-frozen pre-trained language model. Chen et al. (2023) indicated that prompt-based learning modifies tasks to fit models instead of modifying models to fit tasks. A possible beneficial effect of prompt-tuning-based learning is that it may allow the consistent activation of the encoded prior knowledge of pre-trained language models relevant to the summarized documents. In this way, it may avoid irrelevant prior knowledge-induced factual errors otherwise.

Depending on purposes, prompts may be discrete, examples being extracted subject-relation-object triplets (Chen et al., 2023) as prepending input tokens to the pre-trained model. Prompts may also be continuous in the form of trainable parameters in prefix-tuning. Lester et al. (2021) presented two prefix-tuning trainable parameter initialization options for generative tasks: random initialization and designed prompt token embeddings drawn from the pre-trained model's vocabulary. Discrete prompts can facilitate continuous prompt learning.

It becomes apparent that prompt-tuning design is crucial to factual summarization. Chen et al. (2023) prefix-tuned the decoder-only GPT-2 for summarization, where the text 'Key relation:' was dedicated for trainable prompts, followed by the sequence of knowledge triples and source documents inputted to the parameter-frozen decoder. The knowledge triples were extracted from source documents by OpenIE6 (Kolluru et al., 2020) and filtered by named entity recognition.

Ghazvininejad et al. (2022) went beyond the vanilla approach and investigated the network structural design to elevate prefix-tuning on summarization, particularly hierarchical prefix structure along with sparse attention. To this end, the authors developed blocking mechanisms. In a nutshell, blocking splits prefix parameters corresponding to the input sequence segments (as discourses), like grouping. The prefix parameter attention occurs within the group. Bring into a hierarchical context, the authors applied blocking in the lower layers while full attention (i.e. no blocking) is computed at the top layer. Additionally, the authors investigated several sparse attention approaches, particularly soft sparse attention that draws sparse but differentiable attention samples through Gumbel-Softmax. The authors found that the hierarchical blocking at the lower layers with soft sparse attention generally achieved more coherent and faithful summaries and outperformed the baselines in the low-resource settings.

To obtain effective prompts for better hallucination control, Ravaut et al. (2023) divided the summary generation process into two phases: generating a chain of entities from the source document for prompts, and summary generation conditioned on the prompts in addition to the source document. Both generation phases were fine-tuned with prompt-tuning, and shared the same parameter-frozen backbone model but prefixed with different trainable prompt parameters on designed prompt token sets.

2.3.9 Rejection Learning-Based Methods

Training models to generate (or classify) each token correctly and thus form a perfectly factual summary is difficult given a large and noisy corpus, but training them to abstain from uncertain predictions may be more achievable. This gives rise to rejection learning. Cao et al. (2022b) introduced a rejection token class formulated as a decoding regularization term when training a model to identify and reject (or demote) noisy tokens as the rejection token class, whereas Kang and Hashimoto (2020) used a rejection approach to exclude samples of the uncertain quantile when formulating their truncated loss learning objective.

2.3.10 Post-Editing Correction Methods

Differing from the methods tackling hallucinations in the end-to-end training-inference paradigm, post-editing methods emphasize the accuracy of test inference as an after-the-fact inference approach. Hence, post-editing usually engages a correction model to correct factual errors in summaries generated by an ATS model. Training a correction model often employs negative sampling. Negative samples may be engineered from a direct reference summary transformation based on a heuristic analysis of the ATS model-generated summaries. For example, Cao et al. (2020), Chen et al. (2021), and Lee et al. (2022a) constructed negative samples by replacing reference summaries with erroneous information such as entities and numbers. By contrast, Balachandran et al. (2022), in constructing their negative samples, trained an infilling language model to generate erroneous phrases as replacements in reference summaries. Aiming at correcting informative but erroneous model-generated summaries, Dong et al. (2020) trained their correct models using entity span selection and masking techniques, inspired by QA modeling. Masking entity spans for the modeled prediction can be considered a special case of negative sampling. It is worth noting that the authors developed two variant correction models. The QA-span fact correction model conducted corrections by answering a query. Thus, this model corrects a localized factual error once at a time. The alternative is the autoregressive fact correction model. The model corrects all errors in the global context. The former is efficient for correcting summaries distorted by few errors while the latter is more robust otherwise.

Additionally, the training objectives of correction models may also differ from each other. For example, Cao et al. (2020), Balachandran et al. (2022) and Dong et al. (2020) framed the training as a conditional likelihood maximization problem of predicting reference summaries, while Dong et al. (2020) had additionally span bound prediction learning objective. On the other hand, Lee et al. (2022a) employed similar objectives to likelihood maximization on evidence-based entity-level error detection and correction, whereas Chen et al. (2021) trained their correction model by combining classification on positive and negative sample estimations with contrastive learning between positive and negative samples.

2.3.11 Active Learning-Based Methods

Research development in deep learning, including abstractive text summarization, is implicitly an iterative improvement process. Still, active learning provides an iterative framework, that incrementally annotates data labels from unlabeled data sources and improves data quality and model performance through iterative model training, generative data sampling, and data purifying and annotating. One key

characteristic of the process is that it commonly keeps human annotators in the loop to improve data quality and annotations. Some researchers recently explored active learning to improve the factuality of abstractive text summarization. In addition to iterative model performance improvement by various training methods, sample data selection for annotation is critical in improving data quality and annotation efficiency for training. Xia et al. (2024), aiming to address a range of hallucination types, developed a score-based sample selection method. The authors used an entailment model to score the semantic frame errors, a QA-based model to score discourse errors, and a precision model to score content verifiability. Utilizing these scores, the authors formulated a hallucination diversity score and a hallucination score. A sample selection criterion was then established from the two scores in a complementary form. Bearing in mind reducing the number of expensive annotations required to achieve expected model performance, Tsvigun et al. (2022) explored query strategies for the diversity of data annotation and noisy outlier avoidance. The authors formulated an acquisition function of two weighted terms in a complementary form. The first term ensures a selected sample within an unlabeled dataset distribution (i.e., not an outlier), and the second term ensures the chosen sample is dissimilar to the labeled or annotated dataset.

CHAPTER 3

DETERMINANTAL POINT PROCESS BASED SAMPLING FOR MITIGATING UNDESIRABLE WORD REPEATS

3.1 Research Problem

The undesirable word repeat problems are incurred in ATS. For example, given the following news article from the Gigaword test dataset,

> *"tommy haas set the tone early with his serve , and he kept his game in tune long enough to upset <unk> # novak djokovic and reach the wimbledon semifinals for the first time .",*

a model trained on the Gigaword training dataset may generate a summary as follows,

> *"haas djokovic djokovic to wimbledon semifinals ."*.

Repeating words in such an undesirable way affects coherence and could even become a source to distort the facts in generated summaries.

We consider the issue one of the root challenges to generative word probability distribution estimation. More specifically, maximum likelihood estimation (MLE) is induced on stochastic samples in stochastic gradient descent-based optimization, where generative word probability distributions are shaped by the max-a-posterior parameters optimized over training iterations. Furthermore, the studies on model attention mechanisms (e.g., Voita et al., 2019; Clark et al., 2019) have revealed distributional concentration behaviors in model latent space. Such behaviors may thus further reinforce the probability distribution mode concentration. Consequently, ATS models may undesirably repeat the words of high likelihood. Holtzman et al.

(2019) has also observed the phenomenon. In addition to the same line of thought on the cause, the authors have further argued that the small subset of the vocabulary tokens from heavy-tailed low-probability token distributions could be over-represented in the aggregated estimations as the other plausible cause. Our sampling method can naturally address both possible causes.

3.2 Encoder-Decoder

We start with a preliminary formation of the encoder-decoder, which is the foundation modeling architecture of ATS. The detailed mechanism of the encoder-decoder is given in Appendix A.1.

In supervised learning, datasets used to train ATS models consist of source documents paired with reference summaries. Given an l-length source document sequence $\mathbf{x}$=$(x_1, x_2, ..., x_l)$, the encoder generates the intermediate latent states $\mathbf{z}$=$(z_1, z_2, ..., z_l)$. The decoder is then autoregressive on its k-length reference summary sequence $\{y_i\}_{i=1}^{k}$, a teacher-forcing learning strategy (Williams and Zipser, 1989), to correlate with the $\mathbf{z}$ sequence to generate an output sequence $\tilde{\mathbf{y}}$=$(\tilde{y}_1, \tilde{y}_2, ..., \tilde{y}_k)$ that is relevant to the input sequence $\mathbf{x}$. The learning objective is the MLE using the cross-entropy (CE) loss function, expressed as:

$$\mathscr{L}_{\text{MLE}} = -\frac{1}{k}\sum_{i=1}^{k} y_i \log p(\tilde{y}_i | y_{1:i-1}, \mathbf{x}) \tag{3.1}$$

where y_i is the i^{th} word or token ground truth, and $p(\tilde{y}_i | y_{1:i-1}, \mathbf{x})$[1] is the estimated i^{th} word conditional probability distribution.

3.3 Determinantal Point Processes

Before proceeding to our method, we provide a brief description of the determinantal point processes[2] (DPPs) following the definition by Kulesza and Taskar (2012a). Given a dataset $\mathbb{Y}$ and a symmetric real-valued positive semidefinite kernel matrix $K \in \mathbb{R}^{|\mathbb{Y}|\times|\mathbb{Y}|}$ defined over the set, a process $\mathscr{P}$ is a determinantal point process if every random subset $Y \subseteq \mathbb{Y}$ is drawn by the process such that $\mathscr{P}$ is a probability measure by the determinant of the kernel formed by the subset, defined as:

$$\mathscr{P}(Y \subseteq \mathbb{Y}) = \det(K_Y) \tag{3.2}$$

where K_Y is the submatrix of K indexed by the data items of Y, $0 \leq K \leq 1$, and all principal minors $\det(K_Y)$ of K are non-negative.

[1] The rest of the thesis may omit the conditional notation for simplicity without loss of generality when there is no ambiguity.

[2] We confine our discussion in the discrete setting.

A diagonal entry of K is an item marginal probability $\mathscr{P}(e_i \in \mathbb{Y})$, which measures item quality in the set. An off-diagonal entry is a marginal probability $\mathscr{P}(e_i, e_j \in \mathbb{Y})$ of pairwise item similarity:

$$\begin{aligned}\mathscr{P}(e_i, e_j \in \mathbb{Y}) &= \det\left(\begin{bmatrix} K_{ii} & K_{ji} \\ K_{ij} & K_{jj} \end{bmatrix}\right) = K_{ii}K_{jj} - K_{ij}K_{ji} \\ &= \mathscr{P}(e_i \in \mathbb{Y})\mathscr{P}(e_j \in \mathbb{Y}) - K_{ij}^2.\end{aligned} \tag{3.3}$$

Equation (3.3) shows the property of similarity repulsion and therefore supports diversity. However, the marginal probability kernel K is computationally intractable in general.

There is a well-known class of DPPs in machine learning, L-ensembles (Borodin and Rains, 2005), which is permissible for tractable computation. The L-ensembles specifies the instance probability of each possible subset Y. An L-ensemble kernel L only needs to be real values and symmetric. The marginal probability of an L-ensemble is given as:

$$\mathscr{P}(\mathbf{Y} = Y) = \frac{\det(L_Y)}{\sum_{Y \subseteq \mathbb{Y}} \det(L_Y)} = \frac{\det(L_Y)}{\det(L + I)} \propto \det(L_Y). \tag{3.4}$$

DPPs are less explored for text summarization. A conditional DPP was formulated by Kulesza and Taskar (2012a) for extractive summarization of multi-document clusters. Their kernel was formed of a gram matrix of quality and diversity components. The quality component was a parameterized log-linear model, while the TF-IDF data formed the diversity component discussed in Kulesza and Taskar (2012b). The authors trained their model with the L-BFGS optimization algorithm. Li et al. (2019), in their seq2seq model, used the encoder outputs to create a similarity matrix while using the cross-attention to form a quality matrix. An L-ensemble kernel was formed of both matrices. The authors developed an approximated DPP score learning objective. Perez-Beltrachini and Lapata (2021) presented a novel attention-weight formula by combining diversity scores. The diversity scores were computed using the determinants of a 2x2 kernel formed of two entities, an incrementally aggregate latent context vector and an encoder-outputted latent state for each input token. Cho et al. (2019) proposed gram matrix decomposition of an L-ensemble. The authors trained two BERT models. The CLS state of one model was used to learn sentence pair similarity scores. The CLS state of the other model parameterized a log-linear function for the quality term of the gram matrix. A max log-likelihood learning objective was defined on their L-ensemble. Our method differs from the previous works in terms of DPP kernel formation and learning objective formulation, as detailed below.

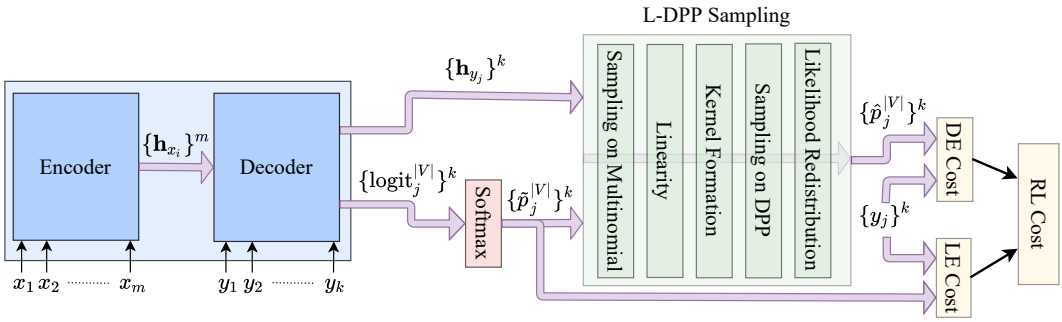


Figure 3.1: An encoder-decoder extended with L-DPP Sampling. The left-hand light-blue block is the encoder-decoder. The right-hand light-teal block is the L-DPP Sampling module extension.

3.4 Our Methods

Figure 3.1 illustrates an encoder-decoder integrated with our L-DPP Sampling method[3], an instance of L-ensembles.

3.4.1 L-DPP Sample Search Algorithm

We model the training vocabulary as an item set Y. A generative subset A of forming a sequence is sampled from Y. To be computationally efficient without compromising the generative power, we take a smaller subset from the top-k most likely candidate words $B \subset Y$ according to $p(Y)$, the estimated probability distributions over the vocabulary of the sequence. Thus, the samples are $A \subseteq B \subset Y$.

In detail, for a generated l-length sequence, we obtain $B^p \in \mathbb{R}^{l \times k}$ highest probabilities and their indices over vocabulary $B^I \in \mathbb{N}^{l \times k}$. We then sample from the multinomial distribution over B^I along the dimension k to get a realization $A \in \mathbb{N}^{l \times 1} \subset B^I$. By A, we can acquire word embeddings $E_w \in \mathbb{R}^{l \times d}$ where d is the embedding dimensionality. A low-rank kernel can be computed by dot product $L_E = E_w E_w^{\mathrm{T}} \in \mathbb{R}^{l \times l}$. The determinant of the kernel is computed thereafter.

Analyses by Peters et al. (2018) and Ethayarajh (2019) have indicated that the latent states at much higher layers of deep neural models show strong task-related contextual semantics. This is important to disambiguate word sense (e.g., Camacho-Collados and Pilehvar, 2018). Thus, we reformulate our kernel elements combined from both the word embeddings and the high-layer latent states. A further linear transformation may be applied to reduce the dimensionality. It is defined as:

$$E = W \cdot (E_w \oplus \mathbf{h}) + \mathbf{b} \tag{3.5}$$

where $\mathbf{h}$ is the corresponding high-layer latent states, $\{W, \mathbf{b}\}$ are the transformation parameters, and the notation $\oplus$ represents a concatenation operator. Note that

[3]We use L-DPP as a short name for the L-DPP Sampling method when there is no ambiguity.

Algorithm 1: LDppSearch

Input : $p, \mathbf{h}, k, n$
Output: A^*
Initial : $s_{max} = null$, $A^* = null$, $j = 0$

$p^{l\times k}, c^{l\times k} = \text{topk}(p, k)$
while $j{+}{+} < n$ **do**
  $index^{l\times 1} \sim \text{multinomial}(p^{l\times k})$
  $c^{l\times 1} = c^{l\times k}[index^{l\times 1}]$
  $e^{l\times d} = \text{embedding}(c^{l\times 1})$
  $e^{l\times d} = W \cdot (e^{l\times d} \oplus \mathbf{h}) + \mathbf{b}$ // (3.5)
  $L^{l\times l} = e^{l\times d} \cdot e^{l\times d^{\mathrm{T}}}$ // (3.6)
  $s = \text{determinant}(L^{l\times l})$
  if $s_{max} == null$ ***or*** $s_{max} < s$ **then**
    $s_{max} = s$
    $A^* = c^{l\times 1}$

Algorithm 2: GenRedist

Input : p, A^*, β
Output: p_{new}

$p' = \beta p[A^*] \overline{+} (1.0 - \beta) p[\overline{A^*}]$
$p_{new} = \text{renormalize}(p')$

Algorithm 3: L-DPP Sampling

Input : $p, \mathbf{h}, k, n, \beta$
Output: p_{new}

$A^* = \text{LDppSearch}(p, \mathbf{h}, k, n)$
$p_{new} = \text{GenRedist}(p, A^*, \beta)$

$\{W, \mathbf{b}\}$ are constant variables since the unnormalized determinantal score is used as a selection criterion for maximizing diversity likelihood and not used in a learning objective for gradient backpropagation. The kernel is then computed as the dot product of the updated elements:

$$L_E = EE^{\mathrm{T}}. \tag{3.6}$$

Equation (3.4) implies that to maximize the probability of an instance subset is to maximize its unnormalized probability. Thus, we can iteratively sample n times and find an optimal subset by the maximum determinantal score:

$$A^* = \underset{A}{\text{argmax}}(\mathscr{P}(A \subseteq B)) = \underset{A}{\text{argmax}}(\det(L_A)). \tag{3.7}$$

We summarize the sample search by the pseudocode in Algorithm 1 where $p \in \mathbb{R}^{l\times|V|}$ is the l decoder-generated word probability distributions over the vocabulary V of size $|V|$, $\mathbf{h}$ represents the high-layer latent states of the decoder, $k \ll |V|$ is the chosen number of the highest probabilities, n is the number of sampling iterations, s_{max} is the updated maximum score used to choose the diverse sample indices over V, and the output A^* is the optimally chosen sample indices. Note that the notation $[\cdot]$ is an indexing operator. For numeric stability, the log-determinant may be used instead of the determinant.

3.4.2 Generative Probability Redistribution

The sample indices cannot be used directly for backpropagation. Borrowing the idea of the reparameterization trick by Kingma and Welling (2014), we use the chosen sample indices to weight the original probability distributions as shown by the pseudocode in Algorithm 2. The input p is the generative probability distributions as before, A^* is the output sample indices from Algorithm 1, and $\beta \in [0.0, 1.0]$ is a redistribution weighting factor. The generative probability distributions indexed by A^* are weighted by β. The rest is weighted by $(1-\beta)$. The updated probability distributions p_{new} are returned upon renormalization. Note that $\overline{A^*}$ is a complementary set of A^*, and $\overline{+}$ represents a masked element-wise addition operator.

3.4.3 L-DPP Sampling Algorithm

Put together, the L-DPP Sampling is summarized in Algorithm 3. It is worth noting that the L-DPP Sampling implies a generative coverage mechanism, which can be an alternative to the well-known coverage mechanisms (e.g., Song et al., 2018; See et al., 2017).

3.4.4 Reinforcement Learning Cost

Having obtained both the standard maximum likelihood estimations and the L-DPP Sampling diversified likelihood estimations, we approach them toward an equilibrium distribution point by a self-critical reinforcement learning (RL), first proposed by Rennie et al. (2017), given that both estimations share the same learning model. We can generally express the self-critical reinforcement learning objective as:

$$\mathscr{L}_{\mathrm{RL}}(\Theta) = -\mathbb{E}_{\hat{y}\sim p'_{\theta'},\tilde{y}\sim p_\theta}(r'(\hat{y}) - r(\tilde{y})), \quad \Theta = \{\theta', \theta\} \tag{3.8}$$

where r is the reward on action (i.e., word prediction) that the baseline (critic) takes, and r' is the reward resulting from the actor's action. p_θ is the baseline's action policy defined by the parameterized network θ while $p'_{\theta'}$ is the actor's policy defined by the network θ'.

We now prescribe a setup of our self-critical RL conceptualized in Figure 3.2. The input samples form a partially observable *environment*. The main encoder-decoder model forms a *value function*. The decoder's classifier and the L-DPP Sampling are viewed as two *agents*. Using the value function, the dual agents derive their distributional *policies* p_θ and $p'_{\theta'}$ over *actions* of word predictions respectively. The goodness of their actions is measured by the CE. The CE outputs are used as the *rewards* $r(\tilde{y}, y)$ and $r'(\hat{y}, y)$ respectively. The agents have different policies and approach their expectations over time. The decoder agent has a greedy policy (i.e., over-confidence) in its actions based on the MLE, which tends to have more bias. The L-DPP Sampling agent maximizes diversified likelihood with high variance in contrast. It is desirable

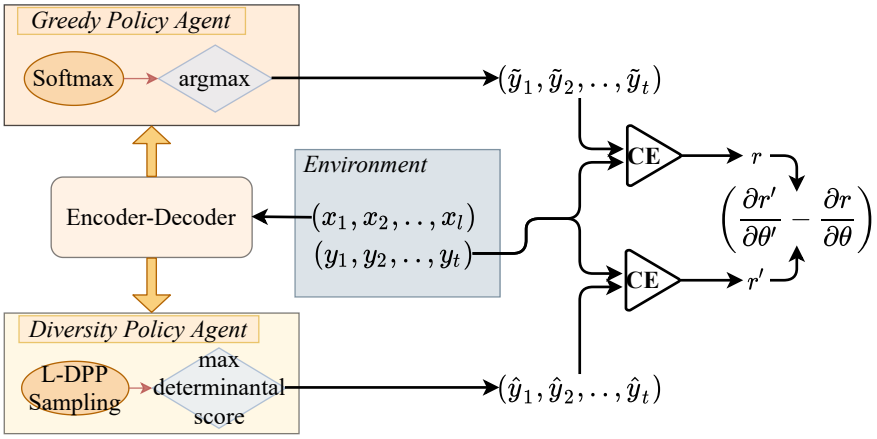


Figure 3.2: Conceptualized self-critical reinforcement learning setting.

to balance bias and variance. As evidenced in RL literature (e.g., Williams, 1992; Rennie et al., 2017), a baseline approach reduces variance, stabilizes and speeds learning convergence. With our dual agents, we use the decoder's expectation as a baseline. As we use CE losses as rewards for sample estimations, we rewrite our self-critical RL objective as follows:

$$\mathscr{L}_{\mathrm{RL}}(\Theta) = \frac{1}{m}\sum_{i=1}^{m}(r'(\hat{y}_i, y_i) - r(\tilde{y}_i, y_i)) \tag{3.9}$$

where

$$r'(\hat{y}_i, y_i) = -\frac{1}{l}\sum_{j=1}^{l} y_{ij} \log p(\hat{y}_{ij}), \tag{3.10}$$

and

$$r(\tilde{y}_i, y_i) = -\frac{1}{l}\sum_{j=1}^{l} y_{ij} \log p(\tilde{y}_{ij}). \tag{3.11}$$

Note that ij denotes the j^{th} word of the i^{th} sample, l is the word sequence length, and m is the number of samples. The $p(\hat{y}_{ij})$ and $p(\tilde{y}_{ij})$ are the estimated probability distributions over vocabulary. The gradients of the learning objective are computed as:

$$\begin{aligned} &\frac{\partial \mathscr{L}_{\mathrm{RL}}(\Theta)}{\partial \Theta} = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{\partial r'(\hat{y}, y)}{\partial \theta'} - \frac{\partial r(\tilde{y}, y)}{\partial \theta}\right) \Rightarrow \\ &\frac{1}{m \times l}\sum_{i=1}^{m}\sum_{j=1}^{l}\left((p(\hat{y}_{ij}) - y_{ij}) - (p(\tilde{y}_{ij}) - y_{ij})\right) = \frac{1}{m \times l}\sum_{i=1}^{m}\sum_{j=1}^{l}\left(p(\hat{y}_{ij}) - p(\tilde{y}_{ij})\right). \end{aligned} \tag{3.12}$$

The gradients are backpropagated to update the *states* of the value function (i.e. the model parameters).

It turns out that the gradients are the difference in the two agent-expected sampling distributions. The gradients approach zero when the agents' expectations

approach an equilibrium distribution point. Therefore, training may then stop 'judicially'.

It is worth noting that the generative sequence is time (sequel) dependent. But a reward is produced at the end of a generative sequence, *trajectory*, and may be considered as a *global return*. Meanwhile, the determinantal scores of the L-DPP Sampling may be viewed as *local rewards* for finding an optimal policy trajectory holistically. The global return is independent of the local rewards once an optimal trajectory is determined. Also, different from Rennie et al. (2017) and the others (e.g., Xiao et al., 2022; Thomas et al., 2022) that use test-time non-differentiable metrics to acquire rewards for self-critical reinforcement learning, the CE used here is more efficient to compute, and confines the rewards to the same value scale without a need for 'whitening' (e.g., Koivunen and Kostinski, 1999).

3.4.5 Total Learning Objective

The total learning objective is a sum of the MLE cost $\mathscr{L}_{\text{MLE}}$ in Equation (3.1) and the self-critical RL cost $\mathscr{L}_{\text{RL}}$ in Equation (3.9), weighted by hyperparameters α_{MLE} and α_{RL} respectively, as follows:

$$\mathscr{L} = \alpha_{\text{MLE}}\mathscr{L}_{\text{MLE}} + \alpha_{\text{RL}}\mathscr{L}_{\text{RL}}. \tag{3.13}$$

3.5 Experimental Results and Analysis

3.5.1 Dataset

We choose two English ATS benchmark datasets, Gigaword (Rush et al., 2015; Graff et al., 2003) and CNN/Daily Mail (CNNDM) (See et al., 2017; Hermann et al., 2015). Both datasets consist of news article and reference summary pairs. CNNDM has much longer articles and reference summaries than those of Gigaword samples. We prepare our datasets as follows.

Gigaword Gigaword 5ed is used. We randomly split the samples into 3,100K training samples, 4096 validation samples, and 2000 test samples. Following Song et al. (2018), we preprocess[4] the training and validation samples, and pad the summaries with a beginning of the sequence (BOS) token and an end of the sequence (EOS) token. We only replace each digit with a hash token for the test samples without further processing.

[4]We follow `https://github.com/KaiQiangSong/struct_infused_summ` to acquire Gigaword dataset, GloVe vocabulary and embeddings, and adopt their data processing code.

CNNDM CNNDM v3.0.0[5] is used. It has 287113 training samples, 13368 validation samples, and 11490 test samples. Following the script[6], we preprocess the CNNDM datasets by using BART model tokenization[7] with pre-trained BART-base profile[8].

3.5.2 Implementation

We experiment with two types of encoder-decoder backbones, including an LSTM-derived model[9] and the Transformer-based BART model (Lewis et al., 2020). We implement the LSTM-derived model by adopting the key components common to Song et al. (2018), See et al. (2017), and Fernandes et al. (2019). Meanwhile, we adopt the BART model implementation from Hugging Face[10] but with a downsized model configuration due to our computational resource constraints. We train the LSTM-derived model on Gigaword and the BART model on CNNDM. All models are trained from scratch. The evaluations of the trained backbone models are free of the L-DPP Sampling module extension since our sampling method with the self-critical RL is only used to train the encoder-decoder backbones. Our source code is accessible on GitHub[11]. The experiment setup is detailed as follows.

3.5.2.1 Backbone Models

LSTM-Derived Model The model includes a BiLSTM encoder, an LSTM decoder, a cross-attention mechanism, and a copyable generative mechanism. Note that the model is also used as a baseline for evaluation comparison. However, we further include a coverage method adopted from Song et al. (2018) when training the model as the baseline. This inclusion gives the baseline an equal footing since the L-DPP Sampling effectuates a generative coverage mechanism.

BART-Based Model The adopted BART model has a standard encoder and decoder construction based on Transformer. It is worth noting that the BART model has learnable embedding modules in the end-to-end training, while the LSTM-derived model uses static input embeddings.

[5]The CNNDM is acquired using Python package datasets released by Hugging Face (`https://huggingface.co`).

[6]`https://github.com/huggingface/transformers/blob/v4.9.2/examples/pytorch/summarization/run_summarization_no_trainer.py`.

[7]`https://huggingface.co/transformers/v4.9.2/model_doc/bart.html#barttokenizer`.

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

[9]We may use encoder-decoder and model exchangeably.

[10]`https://huggingface.co/transformers/v4.9.2/model_doc/bart.html`.

[11]`https://github.com/13114386/ldpp_sampling`.

Table 3.1: LSTM-derived model and training configurations.

Dataset vocabulary size	100K+
Minibatch vocabulary size	5K+
Word embeddings dim	100
Encoder latent state dim	512
Decoder latent state dim	256
Adam optimizer (Adam)	lr = 1e-4 lr decay = 0.98 weight decay = $1e^{-6}$
Gradient clipping	$[-5, 5]$
Minibatch size	36
Max epochs	500
No. iterations per epoch	1000

Table 3.2: BART-based model and training configurations.

Dataset vocabulary size	50264
Model latent dim	512
FFN dim	2048
Attention heads	8
Number of layers	6
Num of beams	1
Adam optimizer (AdamW)	lr = 5e-5 lr decay = linear weight decay = $1e^{-6}$
Minibatch size	16
Max epochs	50
No. iterations per epoch	17944

3.5.2.2 L-DPP Sampling Module

The module consists of embedding transformation and sampling logic. The linearity of the embedding transformation is initialized using a uniform distribution[12]. Meanwhile, we fix the top k to 100, and the probability redistribution weighting factor β to 0.98. The main inputs to the module include the decoder's high-layer latent states and its generative word probability distributions over vocabulary.

With LSTM-Derived Model We follow the mini-batch vocabulary approach used by the adopted prior works. We do not resample for EOS position at the end of a sequence since it is deterministic with no alternative. We simply use the decoder's likelihood estimation at the position.

With BART-Based Model We do not resample for BOS and EOS positions for the same reason described earlier. For numerical stability, we use the log-determinant for the much longer summary sequences of CNNDM.

3.5.2.3 Learning Objective

With LSTM-Derived Model We instantiate α_{MLE} to 0.97 and α_{RL} to 0.03 in the total learning objective.

With BART-Based Model We instantiate both α_{MLE} and α_{RL} to 1.0 to experiment with different weighting factors.

3.5.2.4 Training Setting Summary

With LSTM-Derived Model The key model and training configurations are summarized in Table 3.1. We follow Song et al. (2018) on the model and optimizer

[12] `https://pytorch.org/docs/stable/generated/torch.nn.Linear.html`.

Table 3.3: ROUGE evaluation. 1. Trained with 500 × 1000 (epochs × iterations) on Gigaword. 2. Coverage adopted from Song et al. (2018). 3. Trained using the L-DPP Sampling derived CE in place of the self-critical RL. 4. Trained using the L-DPP Sampling and the self-critical RL.

Model	**Gigaword Test-1951**		
	R-1	R-2	R-L
DRGD (Li et al., 2017)	36.27	17.57	33.62
Struct+2Way +Word (Song et al., 2018)	35.47	17.66	33.52
Model[1]	**Gigaword Test-2000**		
	R-1	R-2	R-L
LSTM-derived +coverage[2]	36.33	15.71	34.39
LSTM-derived /L-DPP[3]	36.61	15.63	34.70
LSTM-derived /L-DPP+RL[4]	36.34	15.28	34.39

Table 3.4: ROUGE evaluation. 1. The downsized BART model and trained with 50 × 17944 (epochs × iterations) on CNNDM. 2. Trained using the L-DPP Sampling derived CE in place of the self-critical RL. 3. Trained using the L-DPP Sampling and the self-critical RL.

Model	**CNNDM Test Set**		
	R-1	R-2	R-L
BART (Lewis et al., 2020)	44.16	21.28	40.90
PEGASUS (Zhang et al., 2020a)	44.17	21.47	41.11
BART+R3F (Aghajanyan et al., 2020)	44.38	21.53	41.17
Model[1]	**CNNDM Test-11490**		
	R-1	R-2	R-L
BART	38.07	15.24	24.95
BART/L-DPP[2]	37.91	14.89	24.73
BART /L-DPP+RL[3]	38.34	15.55	25.23

configurations except for learning decay.

With BART-Based Model The key model and training configurations are listed in Table 3.2. We adopt the training and test run scripts[6] with the downsized model configuration and the weight decay adopted from Table 3.1.

3.5.3 ROUGE Evaluation

ROUGE evaluation results are shown in Table 3.3 for Gigaword and Table 3.4 for CNNDM. We first show the results from some best-performing LSTM-derived models and pre-trained Transformer-based large models at the top of the tables, respectively. Separated by the double lines, the ROUGE scores[13] (on a high confidence interval) from our experiments are followed. Our experiments include the adopted baseline models trained with the used datasets, an ablation study to train the models using the L-DPP Sampling derived CE loss in place of the self-critical RL loss, and the models trained with the L-DPP Sampling and self-critical RL, respectively.

Compared to prior works, we see that the LSTM-based models trained with our methods have competitive results in Table 3.3. On the other hand, the prior works based on the pre-trained large model have an advantage over ours as shown in

[13]ROUGE metrics (R-1, R-2, and R-L) were proposed by Lin (2004). This research uses an implementation from Hugging Face's datasets Python package.

Table 3.5: Word repeat statistics.

Model		**Simple**		**Consecutive**	
		Count	%	Count	%
Gigaword	LSTM-derived+coverage	2260/14028	16.11	1552/14028	11.06
	LSTM-derived/L-DPP+RL	2227/14048	**15.85**	1441/14048	**10.26**
CNNDM	BART	240882/717672	33.56	1743/717672	0.24
	BART/L-DPP+RL	231676/713407	**32.47**	1298/713407	**0.18**

Table 3.4, given that our BART-based backbone is tailored to a much smaller model size and trained from scratch. The results nonetheless provide us with an assessment of the quantitative gap.

With the same model size configuration in our experiments, the BART-based model trained by the L-DPP Sampling and the self-critical RL on CNNDM has an advantage over the adopted baseline on all ROUGE scores, while the LSTM-derived model trained by the L-DPP Sampling and the self-critical RL on Gigaword is competitive to the adopted baseline on ROUGE scores. Compared to the ablation study results, the BART-based model trained by the L-DPP Sampling and the self-critical RL on CNNDM also performs better on all ROUGE scores. Meanwhile, the opposite performance is observed on the LSTM-derived models, where the model trained by the L-DPP Sampling derived CE loss edges ahead of the one trained by the L-DPP Sampling and the self-critical RL.

We think that the performance disparities between the two types of models in our experiments may be due to a few underlying interplay factors. The BART-based model has a model structure advantage with its pairwise token attention mechanism and multi-head feature extraction. Thus, the model may capture more contextual and dimensional features in the token latent states and give them more discriminative power. Also, the RL relies on exploring sample trajectories. The long text sequences of CNNDM may provide the L-DPP Sampling with a larger contextual sample space to explore. Given the two reasons and adequate exploring time, the L-DPP Sampling empowered RL may find better sample trajectories more often. This could suggest that the L-DPP Sampling and the self-critical RL have the potential as a training technique to further improve those pre-trained large models listed in Table 3.4.

3.5.4 Word Count Statistics and Comparison

The ROUGE scores do not directly quantify the word repeats and the word coverage. Thus, we develop counting methods, which are discussed below.

Table 3.6: Unique word coverage.

Model		Count	%
Gigaword	LSTM-derived+coverage	3999/14028	28.51
	LSTM-derived/L-DPP+RL	4047/14048	**28.81**
CNNDM	BART	74155/717672	10.33
	BART/L-DPP+RL	74778/713407	**10.48**

Word Repeat Counting We develop a simple counting method to calculate the ratio of word repeats defined as:

$$r = \frac{\sum_S \sum_{w \in S, \text{Count}(w)>1} \text{Count}(w)}{\sum_S \sum_{w \in S} \text{Count}(w)}. \tag{3.14}$$

We count the repeats of a word w in each generated summary S if it appears more than once. We then sum the counts across all generated summaries as the total repeat counts. A repeat ratio r is calculated as the total repeat counts over the total number of words of all generated summaries. Refined on the simple method, a second method is developed such that it counts the consecutive word repeats as it is observed that the undesirable word repeats are often contiguous. The results are shown in Table 3.5. It is seen that the models trained with the L-DPP Sampling and the self-critical RL reduce the word repeats in generated summaries on both Gigaword and CNNDM datasets.

Generative Word Coverage We also calculate unique word coverage as shown in Table 3.6. It shows that the models trained with the L-DPP Sampling and the self-critical RL infer more unique words and improve the word coverage. It becomes more apparent for the long articles of the CNNDM test set, where our methods result in an increase of 623 unique words despite an overall reduction of 4265 words.

3.5.5 Training Convergence Analysis

Additionally, we also examine the average training loss convergence curves[14] of the LSTM-derived models with and without the L-DPP Sampling and the self-critical RL objective as shown in Figure 3.3. The training with the L-DPP Sampling and the self-critical RL starts with a higher loss, but it converges rapidly. It also shows several alternating patterns of a rapid convergence followed by a short (fluctuating) plateau. We think that the rapid convergence may indicate an effective exploitation of the local optimum, while the short plateau suggests an efficient exploration of a local optimal area. The overall quick convergence trend suggests that the exploration supported by the L-DPP Sampling is not random. We think that the L-DPP Sampling could give rise to a self-amending dynamic data augmentation complementary to

[14]The training loss charts are exported by the Tensorboard package as SVG files. We transform them onto the same chart for proper comparison.

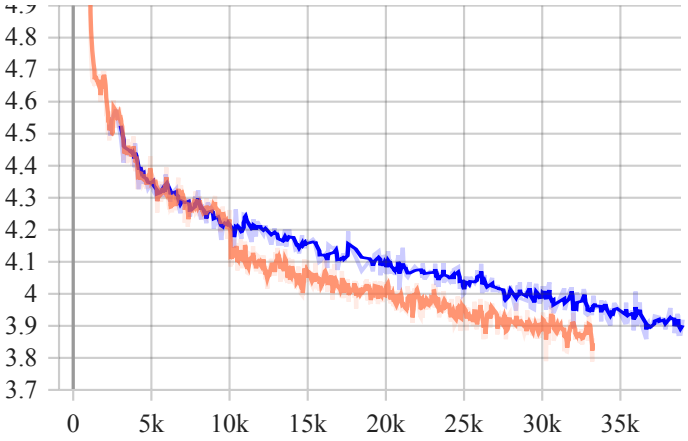


Figure 3.3: Average training losses of LSTM-derived models over a 100 × 1000 run. 1. The blue curve is the convergence of the baseline model. 2. The orange curve is the convergence of the backbone model trained with the L-DPP Sampling and the self-critical RL.

manual annotation for training, and could be complementary to the random mask-based pre-training methods employed in language modeling.

3.5.6 Qualitative Assessment

We further qualitatively assess the summaries generated by the models trained with and without the L-DPP Sampling and the self-critical RL, as examples shown in Table 3.7 on Gigaword test set and Table 3.8 for CNNDM test set. We highlight the undesirable word repeats with red wave symbols and also use blue color to underline the closely related facts in both articles and the summaries generated by the model trained with our methods. We may shorten long source documents by omitting irrelevant content with ellipses.

Summary Assessment on Gigaword The examples in Table 3.7 demonstrate the undesirable word repeat phenomenon and show that the model trained with the L-DPP Sampling and the self-critical RL is capable of reducing the problem. Additionally, the examples demonstrate that the model trained with our methods can improve word coverage and thus include more related facts, which in turn improves comprehensibility and fluency compared to the baseline model.

Table 3.7: Qualitative assessment (examples generated by LSTM-derived models on Gigaword test set).

Source	Text
Article	tommy haas set the tone early with his serve , and he kept his game in tune long enough to upset <unk> # novak djokovic and reach the wimbledon semifinals for the first time .
Reference	haas sets tone early beats djokovic in semifinals
LSTM-derived +coverage	haas djokovic djokovic to wimbledon semifinals .
LSTM-derived /L-DPP+RL	haas beats djokovic to wimbledon semifinals .
Article	russia said wednesday it would review ties with latvia if the baltic state implements citizenship laws which will leave most ethnic russians stateless .
Reference	russia will reconsider relations with latvia after citizenship law
LSTM-derived +coverage	russia says latvia will review ties with latvia .
LSTM-derived /L-DPP+RL	russia to review ties with latvia .
Article	hospitals and clinics in metropolitan manila are dumping <unk> tonnes of infectious waste into the capital 's main river each day , a presidential palace statement said here sunday .
Reference	hospital waste being dumped into manila 's river
LSTM-derived +coverage	hospitals dumping waste into infectious waste into main river .
LSTM-derived /L-DPP+RL	hospitals dumping infectious waste into main river .
Article	former german president johannes rau , ## , died friday after a long illness , his office said friday .
Reference	former german president johannes rau dead at ##
LSTM-derived +coverage	former german president rau rau dies .
LSTM-derived /L-DPP+RL	former german president johannes dies .

Summary Assessment on CNNDM Table 3.8 shows that the undesirable word repeats in the summaries of the long articles are also noticeable at the sub-phrasal level. That is, the sub-phrases share a large portion of a word sequence and convey

the same meaning repeatedly. Despite diversifying token samples, our methods improve the model in such a way that it can compose fluent and coherent summaries from long-distance factual phrases document-wise.

Table 3.8: Qualitative assessment (examples generated by BART-based models on CNNDM test set). For legibility, we replace the generated '\n' with linebreak, and replace generated Unicode codes with corresponding printable symbols.

Source	**Text**
Article	(CNN)Five militants from the Kurdistan Workers' Party were killed and another was wounded in clashes with Turkish armed forces in eastern Turkey, the country's military said Saturday. Four Turkish soldiers also were wounded in the fighting that took place in the eastern city of Agri, the armed forces said in a written statement. ... Turkish Prime Minister Ahmet Davutoglu condemned the violence and said via Twitter that "the appropriate answer to the heinous attack in Agri is being given by the Turkish armed forces." Turkish President Recep Tayyip Erdogan also harshly condemned the attack, describing it as the Kurdish separatists' attempt to "intervene in the resolution process (with the Kurds) in our country." Last month, Abdullah Ocalan, longtime leader of the Kurdistan Workers' Party (PKK), called from his jail cell for the violence to end. ...
Reference	Four Turkish troops were wounded in the flight, according to the country's military. Turkey President Recep Tayyip Erdogan says clashes are attempt to halt a resolution process with Kurds. Violence between Kurds and the Turkish military has been ongoing for more than three decades.
BART	Four Turkish soldiers also are wounded in clashes with Turkish forces. The attack comes ahead of a spring festival. Turkish President condemns the attack, saying it was the Kurdish separatists' attempt to end the violence. Tensions have been over the past three decades. Turkey's population is Kurdish a minority minority long living under cultural oppression.

Continued on next page

Source	Text
BART/L-DPP+RL	Four Turkish soldiers are wounded in fighting in Agri, eastern Turkey. Turkish Prime Minister: "The appropriate answer to the heinous attack in Agir is being given by the Turkish armed forces" Last month, longtime Kurdish leader of the Kurdistan Workers' Party (PKK) called from jail cell for the violence to end.
Article	(CNN)Emergency operators get lots of crazy calls, but few start like this. Caller: "Hello, I'm trapped in this plane and I called my job, but I'm in this plane." Operator: "You're where?" Caller: "I'm inside a plane and I feel like it's up moving in the air. Flight 448 can you please tell somebody (to) stop it." ... The caller was a ramp agent who fell asleep in the plane's cargo hold. ... The pilot radioed air traffic control and said he would make an emergency landing. "There could be a person in there so we're going to come back around," he told air traffic control. The ramp agent who took the untimely nap and caused all the fuss is an employee of Menzies Aviation, ... "The Menzies employee has been permanently banned from working on Alaska Airlines planes," said Bobbie Egan, a spokeswoman for the airline. Flight 448, which was on its way to Los Angeles, only spent 14 minutes in the air. ... The man had been on a four-person team loading baggage onto the flight. "During a pre-departure huddle, the team lead noticed the employee was missing. ...
Reference	The ramp agent fell asleep in the plane's cargo hold. He can no longer work on Alaska Airlines flights.
BART	The man is believed to have been hidden by luggage. He was on a four-person team loading baggage onto the flight. The man was on his cell phone. His co-workers believed he was hidden by the luggage.. It's believed he had been hidden in luggage.
BART/L-DPP+RL	Man was trapped in the plane's cargo hold. He was on a four-person team loading baggage. The ramp agent who fell asleep in the cargo hold was missing. "There could be a person in there so we're going to come back around," a passenger says.

Continued on next page

Source	Text
Article	(CNN)Wanted: film director, must be eager to shoot footage of golden lassos and invisible jets. CNN confirms that Michelle MacLaren is leaving the upcoming "Wonder Woman" movie (The Hollywood Reporter first broke the story). MacLaren was announced as director of the movie in November. CNN obtained a statement from Warner Bros. Pictures that says, "Given creative differences, Warner Bros. and Michelle MacLaren have decided not to move forward with plans to develop and direct 'Wonder Woman' together." (CNN and Warner Bros. Pictures are both owned by Time Warner.) The movie, starring Gal Gadot in the title role of the Amazon princess, is still set for release on June 23, 2017. It's the first theatrical movie centering around the most popular female superhero. Gadot will appear beforehand in "Batman v. Superman: Dawn of Justice," due out March 25, 2016. In the meantime, Warner will need to find someone new for the director's chair.
Reference	Michelle MacLaren is no longer set to direct the first "Wonder Woman" theatrical movie. MacLaren left the project over "creative differences" Movie is currently set for 2017.
BART	Michelle MacLaren is leaving the upcoming "Wonder Woman" movie. The movie is the first theatrical movie centering around the most popular female superhero. It's the first movie cent Ireland's female superhero due out March 25, 2016. In the meantime, Warner will need to find someone new for the director's chair chair.
BART/L-DPP+RL	CNN obtained a statement from Warner Bros. Shortt: Michelle MacLaren is leaving the upcoming "Wonder Woman" movie. The movie is starring Gal Gadot in the title role of the Amazon princess. It's still set for release on June 23, 2017.

Continued on next page

Source	Text
Article	Sanaa, Yemen (CNN)Al Qaeda fighters attacked a prison in the coastal Yemeni city of Al Mukallah early Thursday, freeing at least 270 prisoners, a third of whom have al Qaeda links, a senior Defense Ministry official has told CNN. Khaled Batarfi, a senior al Qaeda figure, was among the escapees, officials said. Dozens of attackers took control of government buildings, including the city's Central Prison, Central Bank and radio station during the assault early Thursday, according to officials. Government troops arrived early Thursday and clashed with the al Qaeda fighters, and most of the militants fled, the officials said. Last month, hundreds of inmates escaped from Al Mansoorah Central Prison in Aden after clashes between Shiite Houthi rebels and forces loyal to ousted Sunni President Abdu Rabu Mansour Hadi. Yemen has been descending into chaos in the weeks since Shiite Houthi rebels removed Hadi, a Sunni, from power. The sectarian nature of the conflict is drawing in regional rivals Saudi Arabia, which is predominately Sunni – and is the country to which Hadi ultimately fled – and Iran, which is predominately Shiite and supports the Houthi rebels. ...
Reference	Al Qaeda fighters attack a prison and other government buildings, freeing many prisoners. Government troops clash with the fighters, most of whom flee. Yemen is descending into chaos as a Shia-Sunni conflict draws in regional rivals Saudi Arabia and Iran.
BART	Yemeni forces clashed with the al Qaeda fighters during the assault early Thursday. Last month, hundreds of inmates escaped from Al Mansoorah Central Prison after clashes. Yemen has been descending into chaos since Shiite Houthi rebels removed rebels. The conflict in Yemen risks becoming a proxy war in the struggle between Iranians and the Saudis.
Continued on next page	

Source	Text
BART/L-DPP+RL	Dozens of attackers take control of government buildings, including Central Prison, Central Bank, radio station. Government troops clashed with al Qaeda fighters, most of the militants fled. Yemen has been descending into chaos since Shiite Houthi rebels removed Hadi, a Sunni, from power.

3.6 Summary

This chapter tackles the undesirable word repeat issues. To this end, a determinantal point process-based sampling method has been developed to address the distributional mode concentration by diversifying samples. A reinforcement-enhanced learning method has also been formulated to guide the determinantal point process-based sampling method under the machine learning principle of bias and variance trade-off.

The experimental results have shown that our methods can diversify word samples and reduce undesirable word repeats. Furthermore, the qualitative assessment has also demonstrated that models trained with our methods draw diverse word samples coherent to the summary contexts and improve the factuality of summaries.

CHAPTER 4

SYNTACTIC STRUCTURE-AWARE SEMANTIC LEARNING FOR MITIGATING DISTORTED SUB-PHRASAL HALLUCINATIONS AND ENDOPHORIC REFERENCE ERRORS

4.1 Research Problem

Our ATS research with the existing models has also observed distorted sub-phrasal hallucinations and endophoric reference errors. A simple example of endophoric reference errors is when a pronoun (e.g., "she") refers to a wrong entity (e.g., a male). The distorted sub-phrasal hallucinations may be less obvious but occur more often. For example, given the following elliptically-shortened source document from the CNN/Daily Mail (CNNDM) test dataset,

> *"... Tim Sherwood and Chris Ramsey know each other inside out. ... The pair worked together at Spurs with Les Ferdinand (second right), now QPR's director of football. ...",*

a pre-trained model fine-tuned with the CNNDM training dataset may generate the summary containing the following sentence,

> *"... Pair worked together at Spurs, now QPR's director of football."*.

It is seen that the sentence misses out on *"with Les Ferdinand (second right)"* in the sub-phrase. Consequently, the summary loses its clarity and may even distort the original meaning of the source document.

These factual issues expose the weakness of the existing models and methods that are formulated to learn sequential word correlations and have a limited capacity to capture complex syntactic structures such as phrase and dependency structures in long text sequences. Recognizing the value of syntactic structure information for ATS, researchers have explored the utilization of syntactic structure data in modeling ATS (e.g., Zhou et al., 2021b; Liu et al., 2022a). These works center on enhancing the token's representational expressiveness by combining token embeddings with structure information for improving generative token classification. Additionally, some researchers (e.g., Zhou et al., 2021b) have also introduced structure label classifications. Both token and structure label classifications use MLE that hinges on the CE loss function for training models.

The CE function suggests that it assumes generative words sampled on independent and identical distribution even though the models are formulated by conditional probabilistic methods. This may lead to generative probability distribution mode concentration such that the models are generalized to the statistically significant individual words. It could be problematic in cases where sub-phrasal words with semantic importance may not statistically co-occur. Thus, learning objectives based solely on the CE in modeling syntactic structures may still be subject to the CE limitations and the distorted sub-phrasal hallucination phenomenon. It is even harder for the CE to generalize long-distance endophoric reference relations. It may thus give rise to erroneous endophoric references.

We believe that utilizing syntactic structure-aware semantic learning tasks in modeling ATS mitigates the factual issues by addressing the CE limitations in a complementary and holistic manner. To this end, we propose a syntactic structure-aware encoder-decoder that incorporates multi-level learning tasks, including syntactic dependency (SD) structure-aware semantic similarity regression, coreference resolution (CR)-based margin ranking regression, and SD label classification. The learning tasks are jointly optimized together with the standard MLE learning objective. Our work uses syntactic structure data, including SD-derived labels, dependency trees, and CR annotations.

4.2 Our Methods

Our syntactic structure-aware semantic learning is illustrated in Figure 4.1. It includes the syntactic dependency (SD) structure-aware semantic similarity regression, the coreference resolution (CR)-based margin ranking regression, and the SD label classification. They are multi-level learning tasks (MLLTs) in that they are applied with the different granularity of syntactic structures and formulated with the latent states at different model layers, detailed as follows.

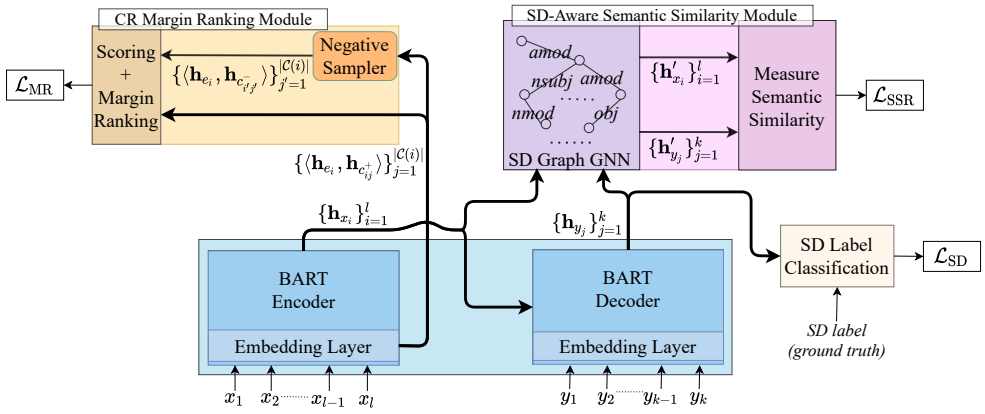


Figure 4.1: Syntactic structure-aware encoder-decoder. It consists of the backbone BART encoder-decoder (blueish block), the syntactic dependency (SD) structure-aware semantic similarity regression (purplish block), the coreference resolution (CR)-based margin ranking regression (yellowish block), and the SD label classification (milky white block).

4.2.1 SD Structure-Aware Semantic Similarity Regression

The goal is to encode holistically the sentential syntax and the semantic alignment of generated summaries with their source documents in the model latent space. We consider that the syntactic structure-aware latent space encodes two levels of semantics, lexical semantics at the word (or token) level and meta-semantics at the structure level. Mikolov et al. (2013) have shown that token semantics can be measured by representational vector similarity in latent space under correlational learning. Our assumption here is that vectors in the syntactic structure-aware latent semantic space still possess such measurable properties under relational learning.

Achieving the goal first requires effective representation learning capable of capturing syntactic structures. We utilize SD graphs and graph neural network (GNN)-based node representation learning[1] for this task. It also requires an effective semantic measure of the learned structure-aware representations between source documents and generated summaries. Conceptually, source documents and corresponding summaries are two coupled distributions. Our aim is to generate a summary from the summary distribution that is semantically and structurally optimal to its source document from the source document distribution. Wasserstein distance, a distance metric between two probability distributions on a metric space, fits our requirement with one exception. That is, it measures two distributions of the same 'mass', for example, to measure predictions against ground truths (e.g., Opitz et al., 2021). Applying the metric to long source documents and short generated summaries of two very different 'masses', we formulate an approximate Wasserstein

[1]The general mechanism and characteristics of GNN for node representation learning is provided in Appendix B.1.

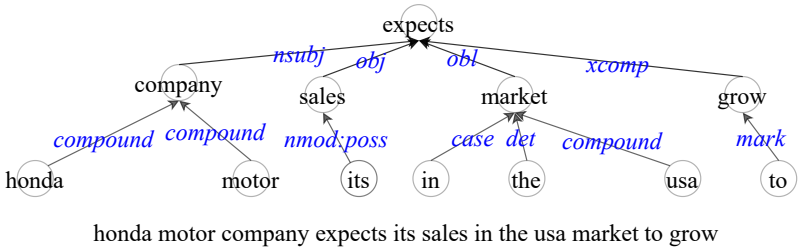


Figure 4.2: A syntactic dependency hierarchy example.

distance that explains away the less relevant (or noise) 'mass' of source documents.

GNN-Based Node Representation Learning We learn the SD structure-aware token representations of both source documents and the generated summaries using a GNN model derived from Principal Neighbourhood Aggregation (PNA) for Graph Nets by Corso et al. (2020). The PNA promotes representational expressiveness by combining multiple aggregation functions with node degree-based scalers. However, our observation of syntactic structure annotations suggests that the structures may be better signified via their dependency hierarchy as illustrated in Figure 4.2. Thus, we instead formulate an SD depth-based scaling function that is also computationally efficient, as discussed below.

Given the word representations of an m-length sentence $(\mathbf{h}_i)_{i=1}^m$, the corresponding SD depths $(d_i)_{i=1}^m$, and the SD relation representation edge set $\{\mathbf{h}^{\mathrm{e}}_{j\rightarrow i}\}_{i,j\leq m}$ (the receiving node[2] i and a connected neighboring node j), we first calculate the depth offset of each node as follows:

$$\Delta d_i = (\max_{j\in m} d_j - d_i) + 1 \tag{4.1}$$

where Δd_i is the one-plus offset of each node depth d_i to the largest depth $\max_{j\in m} d_j$. The average of depth offsets is then computed:

$$\overline{\Delta d} = \frac{1}{m}\sum_{i=1}^{m} \Delta d_i. \tag{4.2}$$

The scaling function is thereafter defined as the ratio of the depth offset over the average offset:

$$s(d_i) = \frac{\Delta d_i}{\overline{\Delta d}}. \tag{4.3}$$

The PNA variant with the scaling function is defined as:

$$\mathrm{PNA}(\mathbf{h}_i) = s(d_i) \cdot \Big[\mu(\mathbf{h}_i), \max(\mathbf{h}_i), \min(\mathbf{h}_i)\Big] \tag{4.4}$$

[2]The graph term 'node' represents the word or token in syntactic relations.

where the aggregation functions are the mean (μ), the maximum (max), and the minimum (min). The notation [,] is a concatenation operator. Put together, a layer of PNA-based GNN has first message passing:

$$\mathbf{h}_{ij} = W^{\mathrm{m}}[\mathbf{h}_i, \mathbf{h}_j, W^{\mathrm{e}}\mathbf{h}^{\mathrm{e}}_{j\to i}] + \mathbf{b}^{\mathrm{m}}, \quad j \in \mathcal{N}(i) \tag{4.5}$$

where W^{e} is the SD relation learning parameters, $\{W^{\mathrm{m}}, \mathbf{b}^{\mathrm{m}}\}$ are the linear transformation parameters, and $\mathcal{N}(i)$ represents the neighboring node set of the receiving node i. This is followed by message aggregation:

$$\mathbf{h}_{\{ij\}} = \mathrm{PNA}(\{\mathbf{h}_{ij}\}) \tag{4.6}$$

where $\{\mathbf{h}_{ij}\}$ is the set of the transformed neighboring node messages, and $\mathbf{h}_{\{ij\}}$ is the aggregated message representation. The node update is then defined as:

$$\mathbf{h}'_i = W^{\mathrm{a}} \cdot [\mathbf{h}_i, \mathbf{h}_{\{ij\}}] + \mathbf{b}^{\mathrm{a}} \tag{4.7}$$

where $\{W^{\mathrm{a}}, \mathbf{b}^{\mathrm{a}}\}$ are the learning parameters. The batch normalization BATCHNORM and non-linearity RELU are further applied:

$$H = \mathrm{RELU}(\mathrm{BATCHNORM}(\mathbf{h}')), \quad \mathbf{h}' = \left[\mathbf{h}'_1, ..., \mathbf{h}'_i, ..., \mathbf{h}'_m\right]^{\mathrm{T}}. \tag{4.8}$$

Semantic Similarity Loss The loss is derived from the Wasserstein distance between the generated summary sequence and its source document sequence, expressed as:

$$\begin{aligned} \mathcal{W}_d &= \min \sum_{i=1}^{l}\sum_{j=1}^{m} \mathrm{T}_{i,j} D_{i,j}, \\ &\text{subject to} \\ \mathrm{T}_{i,j} \geq 0, &\quad \sum_{j=1}^{m} \mathrm{T}_{i,j} = \frac{1}{m}, \quad \sum_{i=1}^{l} \mathrm{T}_{i,j} = \frac{1}{l} \end{aligned} \tag{4.9}$$

where $D_{i,j} \in D$ is the distance between the representations of the i^{th} summary token and the j^{th} source document token, $\mathrm{T}_{i,j} \in \mathrm{T}$ is the joint probability of the (i,j) pair, and l and m are the summary and source document lengths respectively. We now detail the formulations of D and T.

Given a generated summary representation sequence $\mathbf{h}_y$ and its source document representation sequence $\mathbf{h}_x$ obtained from the SD structure-aware node representation learning, the distance is defined as an L2 norm of each pair (i,j):

$$D_{i,j} = ||\mathbf{h}_{y_i} - \mathbf{h}_{x_j}||_2. \tag{4.10}$$

The joint probability is defined as a coupling probability distribution. The coupling between the summary and the source document is learned by a bilinear transformation as follows:

$$\mathrm{T}_{i,j} = \mathbf{h}_{y_i} W_{i,j} \mathbf{h}_{x_j} \tag{4.11}$$

where $W_{i,j} \in W$ is the (i,j) parameter of the weight matrix. The coupling probability distribution is then computed over the source document:

$$\mathrm{T}'_{i,j} = \frac{e^{z_{i,j}}}{\sum_{j'} e^{z_{i,j'}}}, \quad z_{i,j} \in \mathrm{T}_i. \tag{4.12}$$

For the stochastic gradient descent-based minimization training, the loss is finally defined as:

$$\mathscr{L}_{\mathrm{SSR}} = \frac{1}{l \times m} \sum_{i}^{l} \sum_{j}^{m} \mathrm{T}'_{i,j} D_{i,j}. \tag{4.13}$$

The learning objective is to penalize the highly correlated pairs that have large distances. By doing so, it shifts the joint probabilities to the semantically similar or relevant pairs between the summary and the source document and thus explains away the noise 'mass' of the source document.

4.2.2 CR-Based Margin Ranking Regression

Endophoric references are a small set in a long document, but endophoric reference errors could be difficult problems for a learning model because long-distance and sparse characteristics often make references ambiguous. Researchers (e.g., Fernandes et al., 2019; Wu et al., 2021) have used CR to increase the discriminative power of coreference representations in ATS modeling, but the reliance on CE-based MLE may have limited their efficacy. Here we use CR-annotated data to facilitate a margin ranking regression task to address the endophoric reference errors directly, detailed as follows.

Positive Samples A coreference[3] is considered a cluster consisting of an entity (a.k.a. referent or anchor) and its endophoric references throughout a document. The margin ranking task uses the annotated coreferences as positive samples. We express a coreference cluster $\mathbb{C}(i)$ as the representation triplets:

$$\{\langle \mathbf{h}_{e_i}, \mathbf{h}_{r_{ij}}, \mathbf{h}_{c_{ij}} \rangle\}, \quad j \in |\mathbb{C}(i)|, \mathbb{C}(i) \in \mathbb{C}$$

where $\mathbf{h}_{e_i}$ and $\mathbf{h}_{c_{ij}}$ are the representations of the entity i and its reference j respectively, $\mathbf{h}_{r_{ij}}$ is their relation representation, $|\mathbb{C}(i)|$ is the number of references in the cluster, and $\mathbb{C}$ is the cluster set. Note that the annotated CR expresses the relations as coreference attributes. We implement them as attribute embeddings combined with the coreference representations[4]. We thus define the identity relation $\mathbf{h}_{r_{ij}}$ here for completeness.

[3]Named entity coreference is our main interest in this study.

[4]To avoid clutter, the entity $\mathbf{h}_{e_i}$ and the reference $\mathbf{h}_{c_{ij}}$ imply the combination with their attribute embeddings.

Negative Samples The negative samples are constructed from the positive samples by replacing the relation and reference pairs $\langle \mathbf{h}_{r_{ij}}, \mathbf{h}_{c_{ij}} \rangle$ with randomly sampled pairs $\langle \mathbf{h}_{r_{i'j'}}, \mathbf{h}_{c_{i'j'}} \rangle$ within runtime mini-batch samples. To ensure the negative references are not sampled from an anchored cluster itself, we draw the samples only from the other coreference clusters such that the negative samples are given as:

$$\{\langle \mathbf{h}_{e_i}, \mathbf{h}_{r_{i'j'}}, \mathbf{h}_{c_{i'j'}} \rangle\}, \quad j' \in |\mathbb{C}(i')|, \mathbb{C}(i') \in \mathbb{C}, \text{and } i' \neq i. \tag{4.14}$$

Scoring Function The function scores the relations between the references and the anchored referent. It is applied to both positive and negative references. The relations between references (e.g., "he") and referents (e.g., "John") are not symmetric syntactically. We think that keeping the antisymmetric relations in the latent semantic space benefits our learning purposes. Following Trouillon et al. (2016), we design our scoring function in antisymmetric complex embedding space:

$$s_c = \text{Re}(E_s E_r \overline{E_o}^{\text{T}}) \tag{4.15}$$

where E_s is the referent representations, $\overline{E_o}$ is the complex conjugate of the reference representations E_o, and E_r is the relation representations. The score s_c is taken by the real part (expressed by Re operator) since the margin ranking loss computes real values.

Margin Ranking Loss Margin ranking (or triplet ranking) loss, first proposed by Schroff et al. (2015), aims at differentiating between positive and negative samples. The goal now is to disambiguate references among the coreference clusters. It is defined as:

$$\mathscr{L}_{\text{MR}} = \max(0, s_c^+ - s_c^- + m) \tag{4.16}$$

where s_c^+ and s_c^- are the positive and negative reference scores with respect to an anchored referent, and m is a margin hyperparameter. The learning objective encourages the negative samples to have much larger distances than those of the positive samples concerning the anchored referent in the model latent space.

4.2.3 SD Label Classification

We further incorporate an SD label classification. Although the SD structure-aware representations are used, the Wasserstein distance formulated semantic similarity regression is nonetheless unconstrained syntactically. The SD label classification on the decoder's outputs may regularize the syntactic structures of the generated summaries. A multi-class label classification is thus developed for the task. It is similar to the classifier of Krizhevsky et al. (2017).

Classification Loss The multi-class label classification has a CE-based learning objective as follows:

$$\mathscr{L}_{\text{SD}} = -\frac{1}{k}\sum_{i=1}^{k} y_i \log p(\hat{y}_i) \tag{4.17}$$

where $p(\hat{y}_i)$ and y_i are the estimated probability distribution over label vocabulary and the ground truth label of the i^{th} word respectively, and k is the word sequence length.

4.2.4 Total Learning Objective

Having acquired all learning losses, we define the total learning objective as follows:

$$\mathscr{L} = \mathscr{L}_{\text{MLE}} + (\lambda_{\text{SSR}}\mathscr{L}_{\text{SSR}} + \lambda_{\text{MR}}\mathscr{L}_{\text{MR}} + \lambda_{\text{SD}}\mathscr{L}_{\text{SD}}) \tag{4.18}$$

where the first term is the standard MLE loss defined in Equation (3.1), and the multi-level learning task losses are weighted by the configurable hyperparameters $\{\lambda_{\text{SSR}}, \lambda_{\text{MR}}, \lambda_{\text{SD}}\}$.

4.3 Experimental Results and Analysis

4.3.1 Dataset

Our datasets are built from two English ATS benchmark datasets, CNNDM and XSum (Narayan et al., 2018). Similar to CNNDM, XSum consists of the news article and reference summary pairs. The articles of XSum may be either long text comparable to CNNDM's or short in a few sentences similar to Gigaword's. On the other hand, the reference summaries of XSum are much more abstractive and extrinsic[5] compared to CNNDM's. XSum is a more challenging dataset for ATS and has recently become a popular benchmark dataset along with CNNDM for ATS research. We use Stanford CoreNLP[6] v4.4.0 to annotate the required syntactic structure data from the two datasets and thereafter prepare our datasets detailed as follows.

Structure-Aware CNNDM Datasets CNNDM datasets (train, validation, and test) are downloaded as JSON files using Hugging Face's datasets package. We use the Stanford CoreNLP parsing package to acquire syntactic dependency (SD), constituency, and coreference resolution (CR) annotations. We adopt a third-party published code[7] to extract SD labels, dependency structures, and the corresponding words from the dependency annotation data. We have also developed a tool to extract

[5] In the context of text summarization, 'extrinsic' refers to the facts in a summary not evidenced by its source document. It means 'intrinsic' otherwise.

[6] `https://stanfordnlp.github.io/CoreNLP`.

[7] `https://github.com/KaiQiangSong/struct_infused_summ`.

entity referents, endophoric references, and their attributes from the CR annotation data. The CR entities and references are annotated with their positions concerning their sentences and the associated sentence numbers in relation to their documents. The attributes include animacy (e.g., ANIMATE and INANIMATE), gender (e.g., MALE and FEMALE), number (e.g., SINGULAR and PLURAL), and type (e.g., PROPER and PRONOMINAL). We further build the vocabularies for embeddings from the SD label and CR attributes respectively. We calculate sentence lengths and the total number of sentences in each document used for runtime word-token mapping.

The Stanford CoreNLP, by default, uses Penn Treebank to build structure trees of document sentences for parsing. In doing so, the Penn Treebank may tokenize (segment) a word into multiple tokens (subwords). Therefore, we build our article and summary datasets from these tokens. We consider such tokens as words and the corresponding structures as word-level annotations. So, we distinguish them from the runtime tokens created by the model tokenization process, which may divide or transform a word into multiple tokens following a simple or certain algorithmic encoding scheme. Our experiments use the BART model tokenization[8] that employs byte-pair encoding (BPE) (Sennrich et al., 2016).

To avoid repeating the time-consuming token encoding process over every training session, we also preprocess the token encodings of the annotated documents using the model tokenizer with pre-trained BART-base model profile[9]. The length of a tokenized sample sequence may exceed the length limit of the model. To properly apply the syntactic structures (e.g., SD and CR), we exclude these samples instead of truncating them for training. It is worth noting that the Penn Treebank segmented words can be tokenized well using the pre-trained BART model tokenization as observed.

While preparing the token encodings, we have also created a word-token map for the training dataset. The map enables runtime mapping between the annotated word indices and the model-encoded token indices. Additionally, we create the token sequence dependency graph of each word. The graphs are used to facilitate the super token representation learning.

The validation and test datasets are also built from their annotated document words so that they have consistent data distributions with the annotated training dataset, although the validation and test evaluations do not involve structure annotations. We similarly preprocess the token encodings of the built validation and test datasets. Note that the parsing tool may occasionally fail to annotate dataset sam-

[8] `https://huggingface.co/transformers/v4.9.2/model_doc/bart.html#barttokenizer`.

[9] The profile (`https://huggingface.co/facebook/bart-base`) contains the token vocabulary.

Table 4.1: Structure-aware dataset sizes. 1. For XSum, the first column lists the training dataset size without CR-annotated data, and the second column shows the training dataset size with CR-annotated data. Validation and test datasets remain the same respectively, since we do not use structure data for validation and test evaluations.

Dataset	**Size**		
	CNNDM	**XSum**[1]	
Train	149634	186873	65698
Validation	7782	10391	10391
Test	11483	11328	11328

ples properly. Therefore, these samples are excluded. Table 4.1 lists the preprocessed dataset sizes for CNNDM.

Structure-Aware XSum Datasets We similarly use Hugging Face's datasets package to download XSum datasets. We build XSum datasets the same way as we build the annotated CNNDM datasets. It yields a smaller XSum training dataset with CR-annotated data. The small dataset may not be adequate to train models with our learning tasks. Thus, we also build a larger training dataset without CR-annotated data. We first train our models using the larger dataset without the CR-based margin ranking task. We then include the CR-based regression task to further fine-tune the trained models with the CR-annotated data included dataset. The preprocessed dataset sizes for XSum are listed in Table 4.1.

4.3.2 Implementation

We continue adopting the BART encoder-decoder as before. However, we are able to scale up the model to a pre-trained BART-base implementation as our backbone model, given that more computational resources are made available to our experiments. Once the backbone is trained with our learning task modules (MLLTs), the model evaluation uses the backbone only (without the MLLTs modules). Our source code is accessible on GitHub[10]. The experiment setup is detailed as follows.

4.3.2.1 Super Token Representation Learning

Researchers have applied the word-level structures either to the leading tokens of words (e.g., Heinzerling and Strube, 2019) or to the aggregated token representations (e.g., Ek and Bernardy, 2020). We take into consideration of byte-pair encoding (BPE) used by the BART tokenizer. That is, the first token by the BPE scheme may not be unique to a word. The aggregation approach would mitigate potential ambiguity among words encoded with the same leading token. For this reason, we take the latter approach and apply a GNN-based super token representation learning using token

[10] https://github.com/13114386/jhHWqEsMPSlxuIx.

sequence graphs created during data preparation. The super token representation learning is detailed in Appendix B.2.

4.3.2.2 Index Mapping For Applying Word-Level Structure to Token Representation

The model-encoded tokens and the word-level structures are not aligned by their sequential position indices. That is, their corresponding indices are not the same. To apply the structures to the aforementioned super token representations, we have developed source codes for index mapping by utilizing the word-token map created during data preparation.

4.3.2.3 SD Structure-Aware Semantic Similarity Regression

We default the GNN model for structure-aware node representation learning to a two-layer and undirected message-passing configuration after experimenting with several configurations (e.g., four-layer and directed). The SD node representation learning adopts the PNA implementation from PyTorch Geometric package[11]. We extend the PNA implementation to include our depth-based scaling function.

4.3.2.4 CR-Based Margin Ranking Regression

Entity referents and endophoric references may consist of multiple words. To simplify the learning task without loss of representational discriminative power, we take a super node representation learning approach similar to the super token representation learning. In short, we center on the keyword of an entity or endophoric reference and apply a GNN to aggregate its neighboring (up to) n-gram words at each side of the keyword if applicable[12]. We set n-gram to 2 in our experiments.

Coming to choosing a margin, we have experimented with values of 40, 50, and 60. The best performance is achieved at 50.

Our CR margin ranking regression refactors several functions from the knowledge graph learning package PyKeen[13]. The changes are related to the complex vector-based model, interaction function, and margin ranking loss function.

4.3.2.5 Training Setting Summary

We summarize the key model and training configurations in Table 4.2. We initialize the backbone BART model with the pre-trained BART-base model weight profile[14]. The model has 6 layers, 12 attention heads, a filter size of 3072, and hidden state dimensions of 768. The optimizer configuration follows the same configuration in

[11] `https://pytorch-geometric.readthedocs.io/`.

[12] Entity annotations may occasionally include the restrictive clause (e.g., which clause) as a whole improperly. Imposing an n-gram constraint mitigates the problem.

[13] `https://github.com/pykeen/pykeen/tree/master/src/pykeen`.

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

Table 4.2: Key model and training configurations.

Model latent dim FFN dim Attention heads Number of layers	768 3072 12 6
Adam optimizer (AdamW)	lr = $5e-5$ lr decay = linear weight decay = $1e^{-6}$
Max epoch Early stop epoch number	50 4

Table 4.3: Inference settings.

Setting	**CNNDM**	**XSum**
Maximum Length	142	62
Minimum Length	56	11
Beam No.	4	6
Length Penalty	2.0	1.0

Chapter 3. Additionally, we instantiate Equation 4.18 by setting λ_{SSR} to 0.5 and defaulting the other loss weighting factors to 1.0.

We use an early-stop training approach up to the configured maximum epoch. The criterion is the ROUGE metric-based evaluation of the validation dataset at each training epoch. The same ROUGE metrics for test time inference evaluation are used. The training stops when the validation ROUGE scores flatten consecutively for the number of configured times. We set the number to 4 epochs in our experiments.

4.3.2.6 Training on Multi-GPUs

We employ a multi-GPU running procedure based on the reference runtime script[15] from Hugging Face. We train models using a dual-GPU setting with two shared NVIDIA Quadro RTX 8000/48GB cards. A model with the fully configured MLLTs has a size of 599.98MB. A training session of the model with CNNDM datasets (training and validation) on the configured early stop setting takes about 60 hours. A training session with XSum datasets without the CR-based margin ranking regression takes roughly 95 hours on the same early stop setting. Further fine-tuning the model with the CR-based margin ranking regression takes about 6 hours on the same early stop setting.

4.3.2.7 Inference Setting

For evaluation, we extract the inference settings from the pre-trained BART-base configuration as shown in Table 4.3.

[15] https://github.com/huggingface/transformers/blob/v4.9.2/examples/pytorch/summarization/run_summarization_no_trainer.py.

Table 4.4: ROUGE evaluation. 1. Our test samples are 11483 (11490 in the original test dataset) for CNNDM and 11328 (11334 in the original test dataset) for XSum. 2. Our summary-level R-L (equivalent to the R-Lsum ROUGE metric this research uses). 3. The baseline BART-base is fine-tuned on our annotated training datasets. 4. The model (denoted by *) is trained with our fully configured MLLTs.

Model	**CNNDM Test Set**[1]			**XSum Test Set**[1]		
	R-1	R-2	R-L[2]	R-1	R-2	R-L[2]
QA-Span/BertSumExtAbs (Dong et al., 2020)	41.75	19.27	38.81	36.86	14.82	29.70
ERPGN/BART-Base (Lyu et al., 2022)	42.28	19.64	38.93	39.60	16.90	31.74
BART-base[3]	42.69	19.36	39.66	41.54	18.63	33.48
BART-base*/MLLTs[4]	**43.00**	**19.67**	**39.91**	41.50	18.58	**33.56**

4.3.3 ROUGE Evaluation

We first evaluate our trained models using ROUGE metrics for both CNNDM and XSum as shown in Table 4.4. We provide comparisons with the results of recent ATS factuality research having pre-trained backbone model sizes same as or similar to ours. Separated by double lines, the table includes prior works, followed by the ROUGE scores (on high confidence interval) from the fine-tuned baseline BART-base and our fully configured MLLTs-trained model. The models trained using our annotated datasets have shown improved performance on ROUGE scores mostly compared to the prior works. The MLLTs-trained model also scores higher than the fine-tuned baseline BART-base on CNNDM. On XSum, the MLLTs-trained model achieves better R-L score, while the fine-tuned baseline has an edge on R-1 and R-2. We are to explore the plausible reasons for the discrepancy in the model performance on the two datasets later in our human evaluation and qualitative assessment.

4.3.4 Ablation Study

We have also conducted the ablation studies of our multi-level learning tasks with our annotated CNNDM[16] to assess their impact and effectiveness. Table 4.5 shows the results. The ablation experiments indicate that excluding both the SD structure-aware semantic similarity and the CR-based margin ranking has the most negative impact on the backbone model's performance. Although omitting the SD label classification has the least impact, incorporating it can complement the proposed regression tasks to yield an even better model performance.

It is worth noting that we configure the different learning tasks with different layer latent states of the backbone model. The CR-annotated data express sparse and long-

[16]The ablation experiments are only conducted on CNNDM due to computational resource constraints.

Table 4.5: Ablation Study. 1. Summary-level R-L (equivalent to the R-Lsum ROUGE metric this research uses). 2. SD label classification is excluded. 3. SD structure-aware semantic similarity is excluded. 4. CR-based margin ranking is excluded. 5. Both SD structure-aware semantic similarity and CR-based margin ranking are excluded. 6. The results from the model trained with fully configured MLLTs for comparison.

Model	**CNNDM Test Set**		
	R-1	R-2	R-L[1]
BART-base/MLLTs ∌(SD)[2]	42.85	19.59	39.80
BART-base/MLLTs ∌(SD-SSR)[3]	42.79	19.50	39.73
BART-base/MLLTs ∌(CR-MRR)[4]	42.80	19.48	39.76
BART-base/MLLTs ∌(CR-MRR,SD-SSR)[5]	42.76	19.39	39.73
BART-base*/MLLTs[6]	**43.00**	**19.67**	**39.91**

distance relations that tend to be less contextual on their own, while the SD data annotate contextual sentence structures. Previous studies (e.g., Ethayarajh, 2019; Peters et al., 2018) have indicated that higher-layer latent states (representations) encode more contextual information while lower-layer latent states tend to be generic or context-agnostic. Taking into account these studies, we formulate the CR-based margin ranking regression with the lower-layer latent states of the encoder in our experiments, as illustrated in Figure 4.1. On the other hand, both the SD structure-aware semantic similarity regression and the SD label classification fit well with the last layer latent states of the encoder and/or the decoder.

4.3.5 Automatic Factuality Consistency Evaluation

We carry out an automatic factuality consistency evaluation, following the studies by Kryscinski et al. (2020) and Maynez et al. (2020). They have indicated that ROUGE scores may not be a sufficient indicator of summarized factuality in terms of factual errors in model-generated summaries. Researchers have developed several automatic evaluation metrics to assess the ATS factuality consistency (e.g., Kryscinski et al., 2020; Scialom et al., 2021). We adopt the more recent and publicly accessible evaluation metric, SummaC (Laban et al., 2022), which outperforms several prior evaluation metrics on six benchmark datasets. SummaC has two variants: $\text{SummaC}_{\text{ZS}}$ and $\text{SummaC}_{\text{Conv}}$. The authors have indicated that the former is highly sensitive to extrema, while the latter is developed to mitigate the sensitivity issue and be more objective. Thus, we use $\text{SummaC}_{\text{Conv}}$ for our evaluation.

We first use the metric[17] to score each summary related to its source document over the test set. We thereafter compute their mean score. The results are presented in Table 4.6 for both CNNDM and XSum. We further conduct a paired t-test[18]

[17] https://github.com/tingofurro/summac.

[18] We use ttest_rel API from Python's scipy.stats package.

Table 4.6: $\text{SummaC}_{\text{Conv}}$ mean score statistic.

Model	**$\text{SummaC}_{\text{Conv}}$**	
	CNNDM μ(%)	XSum μ(%)
BART-base	67.1	23.7
BART-base*/MLLTs	**68.7**	**23.8**

Table 4.7: $\text{SummaC}_{\text{Conv}}$ scores statistical significance (by paired t-test).

Model	**$\text{SummaC}_{\text{Conv}}$**	
	CNNDM	XSum
BART-base vs BART-base*/MLLTs	2.87e-15 (<0.05)	0.24 (>0.05)

to examine the statistical significance of the scores between the two models. The null hypothesis is that there is no significant difference between the baseline and our scores (p-value < 0.05). The results are shown in Table 4.7. The results in Table 4.6 and Table 4.7 indicate that the MLLTs-trained model generates summaries significantly better for the extractive type of summaries (i.e., CNNDM), whereas the extremely abstractive and concise nature of XSum may limit our tasks of learning localized structures. However, the characteristics of XSum may have enabled the learning tasks to gather concepts following structure signals and compose them from long-distance text spans, as this could be suggested by the better R-L score on XSum shown in Table 4.4.

It is worth noting that the scores for XSum are considerably lower than those for CNNDM. This might be due to the reference summaries of XSum being more abstractive than those of CNNDM, as studied by Narayan et al. (2018) and Lu et al. (2020). Lu et al. (2020) have also found that the XSum reference summaries are highly extrinsic. Their study shows that XSum has 35.76% and 83.45% novel unigrams and bigrams, respectively, in contrast to CNNDM's 17.00% and 53.91%. The distributional divergence between source documents and reference summaries in XSum may expose the data imbalance in training model-based factuality consistency metrics like SummaC for evaluating summaries that tend to be characteristically extrinsic.

4.3.6 Human Evaluation

We[19] have further performed human evaluations in addition to auto-metric evaluations. Similar to the approaches of Gabriel et al. (2021), Lewis et al. (2020), and Chen and Bansal (2018), our evaluation compares model-generated summaries on two key criteria (Faithfulness[20] and Fluency). The detailed guidelines for this assessment on

[19]The author takes part in the evaluation.

[20]Faithfulness is closely related to factuality. Faithfulness seeks all information in a summary supported by its source document, while factuality may allow extrinsic facts that are not evidenced in the source document.

Table 4.8: Human evaluation (better summary statistics) on randomly drawn samples of generated summaries.

Annotator	**Model**	**CNNDM** (50 samples)				**XSum** (50 samples)			
		Faithfulness		Fluency		Faithfulness		Fluency	
		No.	%	No.	%	No.	%	No.	%
A	BART-base	15/50	30.0	8/50	16.0	11/50	22.0	11/50	22.0
	BART-base* /MLLTs	14/50	28.0	14/50	28.0	16/50	32.0	15/50	30.0
B	BART-base	22/50	44.0	24/50	48.0	12/50	24.0	17/50	34.0
	BART-base* /MLLTs	21/50	42.0	24/50	48.0	25/50	50.0	28/50	56.0
Average	BART-base	**18.5**/50	**37.0**	16/50	32.0	11.5/50	23.0	14/50	28.0
	BART-base* /MLLTs	17.5/50	35.0	**19**/50	**38.0**	**20.5**/50	**41.0**	**21.5**/50	**43.0**

the criteria are given in Appendix B.3.1. An evaluation tool with a user interface is developed to facilitate the evaluation (See Appendix B.3.2). The user interface presents each article and its model-generated summaries side-by-side on a page. The interface asks an annotator to select the better summary for each of the two criteria. An annotator can also rank the summaries as 'tie' otherwise. The tool draws 50 random samples from CNNDM and XSum test sets, respectively. A total of 100 samples are shown to an annotator page-by-page. The tool presents summaries in a randomly shuffled order to ensure the evaluation is as fair as possible. The user interface also does not indicate which dataset a sample is drawn from. The tool records an annotator's choices in a back-end database from which the queried analysis is conducted upon completion of the evaluation.

The evaluation results are shown in Table 4.8 in which the better summary statistics are listed. It is seen that two annotators have distinguishable opinions on what forms better summaries. The annotator A seems more cautious and thus opts for 'tie' (i.e., fewer counts of better summaries), while the annotator B prefers summaries one way or another more often.

On CNNDM, annotator A considers that the MLLTs-trained model produces considerably more fluent summaries than the baseline, whereas annotator B has a view that both models generate summaries fluent on par. Meanwhile, both annotators have found that the MLLTs-trained model has one more summary incurring faithful issues than the baseline.

On XSum, two annotators are consistent in the inter-summary comparison of the generated summaries. They have found that the MLLTs-trained model generates noticeably more numbers of better summaries than the baseline on both criteria. This suggests that the latent space encoded with syntactic structures improves the reorganization of long-distance factual information for extremely abstractive

summaries. It agrees with our belief discussed earlier based on the results of auto-metric evaluations.

In summary, the MLLTs-trained model consistently outperforms the baseline on the generated summaries of XSum with noticeable margins, while it is less conclusive on those of CNNDM given the relatively smaller margins in results. This suggests that longer summaries of CNNDM could be more challenging to evaluate than the concise ones of XSum.

4.3.7 Qualitative Assessment

We provide a qualitative assessment to further shed some light on the plausible reasons behind the ROUGE, factuality consistency evaluation scores, and human evaluation results. Table 4.9 for CNNDM and Table 4.10 for XSum compare several generated summary examples respectively. The generated summaries demonstrate that the MLLTs-trained models improve the factual consistency of sub-phrases. For CNNDM, the MLLTs-trained model improves the phrasal factual consistency, which tends to be localized and extractive. On the other hand, the summaries for XSum exhibit different characteristics as the models often reorganize sparse concepts and related factual information document-wise into new sub-phrases. Therefore, these summaries are more abstractive and difficult to summarize, yet the MLLTs-trained model performs better than the baseline in extracting and reorganizing globally relevant information into concise and factually consistent summaries that also display sound syntax and coherent semantics. Our observations and rationale might explain why the MLLTs-trained models score higher on three ROUGE metrics with CNNDM but only higher on R-L with XSum compared to the baseline. Meanwhile, endophoric reference problems are much less frequent, but some generated summaries might have shown evidence that our approach could help mitigate endophoric reference errors, for instance, the third example in Table 4.9.

Table 4.9: Generative summary assessment on CNNDM. We use ellipses to omit long content not relevant or critical to our illustration. We underline the related facts in both source documents and the generated summaries with straight blue lines. We also highlight the factual issues in the generated summaries with red wave symbols.

No.	Source	Text
1	Article	Matthew Hall, ... in Manchester's fashionable Northern Quarter district ... after scaling the walls of trendy apartment blocks where ...
	BART-base	Matthew Hall, ... scaled the walls of Manchester's fashionable Northern Quarter district. ...
	BART-base*/MLLTs	Matthew Hall, ... after scaling walls of trendy apartment blocks. ...
2	Article	... Tim Sherwood and Chris Ramsey know each other inside out. ... The pair worked together at Spurs with Les Ferdinand (second right), now QPR's director of football. ...
	BART-base	... Pair worked together at Spurs, now QPR's director of football.
	BART-base*/MLLTs	Tim Sherwood and Chris Ramsey know each other inside out. ... The pair worked together at Spurs with Les Ferdinand, now QPR's director of football. ...
3	Article	... Bikram Choudhury built an empire. ... after the guru was accused of rape or sexual assault by six of his former students. ... Sarah Baughn, a former student who now accuses Choudhury of sexual assault. ... He said he's guided by a deep calling to help others. ...
	BART-base	Bikram Choudhury says he wants to set the record straight. ... Sarah Baughn says he's guided by a deep calling to help others.
	BART-base*/MLLTs	Bikram Choudhury says he never sexually assaulted anyone. ... He says he feels sorry for his accusers, claiming they've been manipulated to lie.

Continued on next page

No.	Source	Text
4	Article	... In 2011, al Qaeda took Warren Weinstein hostage. ... his family paid money to his captors, ... the captors ... began demanding prisoners be released in exchange for Weinstein, ...
	BART-base	Al Qaeda took Warren Weinstein hostage in 2011, then paid money to his captors. ...
	BART-base*/MLLTs	After al Qaeda took Warren Weinstein hostage in 2011, his captors began demanding prisoners be released in exchange for Weinstein. ...

Table 4.10: Generative summary assessment on XSum. We use the same illustration approach as with Table 4.9.

No.	Source	Text
1	Article	... But Mr Farage, ... "We must be completely mad, as a country, to be giving people from Eastern Europe in-work benefits," he told BBC News. ...
	BART-base	UKIP Leader Nigel Farage has said the government should be "completely mad" to cut immigration from Eastern Europe by claiming in-work benefits.
	BART-base*/MLLTs	The UK must be "completely mad" to be giving migrants from Eastern Europe in-work benefits, former UKIP leader Nigel Farage has said.
2	Article	The test investigates whether people can detect if they are talking to machines or humans. ... The 65-year-old Turing Test is successfully passed if a computer is mistaken for a human more than 30% of the time during a series of five-minute keyboard conversations. On 7 June Eugene convinced 33% of the judges at the Royal Society in London that it was human. ... The event was organised by Reading University's School of Systems Engineering in partnership with RoboLaw, ...

Continued on next page

No.	Source	Text
	BART-base	An artificial intelligence system won a competition to see if it was man or woman during a series of five-minute keyboard conversations.
	BART-base*/MLLTs	An artificial intelligence system called Eugene has passed a Turing test, which was conducted at the University of Reading.
3	Article	In an interview marking five years as first minister, ... "If we see the Tory plans, ... " I do not know what the NHS in Wales would look like by 2020, or the education services, if we see those levels of cuts. ...
	BART-base	The first minister has said there will be no further cuts to the NHS until 2020 if the Conservatives win the general election.
	BART-base*/MLLTs	Wales could have no NHS by 2020 if Conservative plans for further cuts go ahead, the first minister has said.
4	Article	The closure of the tunnel near Linlithgow in West Lothian will mean rail services between Edinburgh and Glasgow will be severely hit. ... Engineers said the tunnel closure was necessary ... Train operator ScotRail has a dedicated website to advise on the disruption, which will mainly hit travel between Glasgow - or Stirling/Dunblane - and Edinburgh ... programme director of EGIP for Network Rail, said the tunnel work over the next six weeks ... He said: "We can't avoid the work in Winchburgh tunnel. ... Work on the ... M74 Motorway Improvements Project in Lanarkshire is already leading to disruption as traffic restrictions are in force. ...
	BART-base	Work on the main Edinburgh to Glasgow railway line is due to begin next month with the closure of the M74 motorway tunnel.
	BART-base*/MLLTs	Work on the Winchburgh railway tunnel is expected to cause major disruption to Glasgow-Edinburgh rail services over the next six weeks.

4.4 Summary

This chapter addresses the distorted sub-phrasal hallucinations and the endophoric reference errors. A syntactic structure-aware semantic similarity learning has been developed to tackle distorted sub-phrasal hallucinations. A margin ranking learning has been formulated to mitigate endophoric reference errors. A structure label classification has also been added to further regularize summary generation.

The experimental results have shown that the combined methods can improve model performance and factuality. Specifically, auto-metric evaluations have shown the model trained with our methods performs better over the baseline on the dataset of characteristically intrinsic reference summaries, while human evaluation has shown a big margin gain in faithfulness over the baseline on the dataset of reference summaries characteristically extrinsic. The discrepancy between auto-metric and human evaluations indicates that the model-based metrics may be deficient in extrinsic knowledge encoded in the metric models due to their training dataset mainly being intrinsic. The qualitative assessment has further demonstrated that the models trained with our combined methods mitigate distorted sub-phrasal hallucinations compared to the baselines. On the other hand, the qualitative assessment has not been conclusive on endophoric reference error reduction as endophoric reference problems are much less frequent. Our future work may further investigate the endophoric reference error reduction by broadening the sample search.

CHAPTER 5

ADAPTIVE MARGIN RANKING LOSS ENHANCED ENTITY ALIGNMENT LEARNING FOR MITIGATING INTRINSIC NAMED ENTITY-RELATED HALLUCINATIONS

5.1 Research Problem

Our study has found that summarization of long text sequences (e.g., CNNDM documents) incurs high occurrences of named entity-related hallucinations[1] (NERHs) compared to other types of factual problems (e.g., negation-related). NERHs can be divided into intrinsic and extrinsic. The former includes the cases where a named entity hallucinated in a summary is mentioned in the source document, while the latter occurs when a summary introduces a named entity novel to the source document. We are interested in the intrinsic NERHs as the main source of NERHs. Among the intrinsic NERHs, entity-entity hallucinations are often observed. For example, given the following elliptically-shortened source document from the CNNDM test set,

> *"... Since civil war ..., 310,000 people have been killed, the Syrian Observatory for Human Rights said Thursday. ... estimate by the U.N. of at least 220,000 dead. ..."*,

a pre-trained model fine-tuned with the CNNDM training data may generate the summary having the following segment,

> *"U.N.: More than 310,000 people have been killed in Syria ..."*.

[1] This investigation focuses on CR-based named entities.

The summary mistakes "the Syrian Observatory for Human Rights" for "U.N.". We believe that the cause is rooted in the misalignment of entities with their contexts in terms of sentences in the model latent space.

The other form of intrinsic NERHs is entity-reference hallucinations. The previously observed endophoric reference errors in Chapter 4 are the cases of entity-reference hallucinations. For example, given the following elliptically-shortened source document,

> *"... with an eight-month-old baby, ... Savannah Guthrie ... help her precious baby girl Vale drift off. When ... mother-of-one discussed an Australian father's tip for getting his baby to sleep ..."*,

the fine-tuned model generates the summary containing the segment as follows,

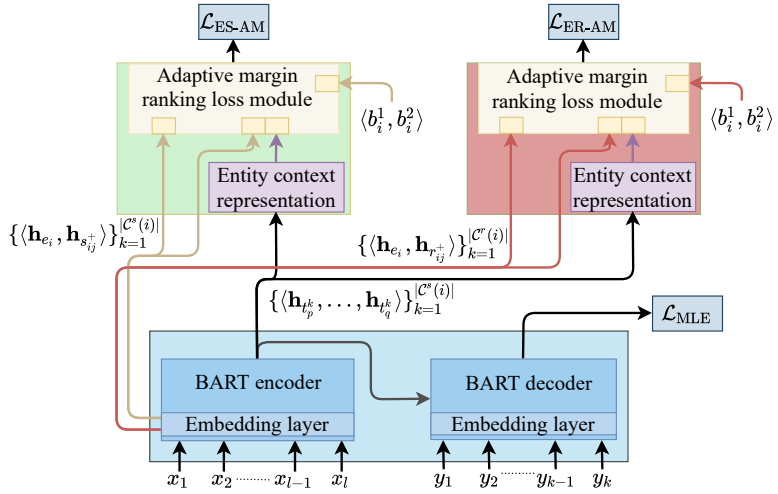
> *"The ... mother-of-one discussed ... tips for getting his eight-month-old daughter to sleep ..."*.

It is seen that "her" is mistaken for "his" in the summary. We think that the cause could lie in the reference similarity in the model latent space in addition to the CE limitations discussed in Chapter 4.

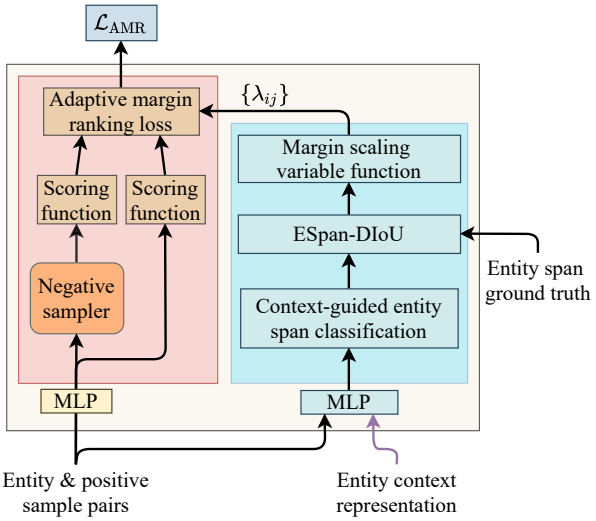
The illustrated examples suggest that ATS models could fail to learn from data to distinguish positive candidates (e.g., the Syrian Observatory for Human Rights) from negative ones (e.g., U.N.) in the latent space. Margin ranking loss is well-known for tackling such problems. Our previous work has used margin ranking loss to tackle endophoric reference errors and shown some evidence of mitigating the problem.

The classical margin ranking loss treats all samples identically or equally conditioned, but various forms of intrinsic NERHs are implicit in data with complicated linguistic structures and might interplay at both lexical and contextual levels. Data samples thus pose varying learning conditions (easy or difficult) for a model. Yet, it is difficult to define and categorize them in modeling. It is desirable for methods adaptive to the variant learning conditions at different granularity. Thus, we propose an adaptive margin ranking loss to bridge the gap in tackling our hypothesized causes of the illustrated examples. In doing so, we utilize the adaptive loss to facilitate two entity alignment methods to mitigate the intrinsic NERHs jointly.

In developing our adaptive methods with the knowledge gained in our previous work, we desire a real-valued margin-scaling variable function to produce adaptive margins with three properties. That is, they are proportional to the (easy or difficult) learning conditions implicit in samples; they are real values within a well-defined scalar space permissible for our learning purpose; and the base margin is preserved at the low bound of the scalar space.



(a) Extended BART encoder-decoder.



(b) Adaptive margin ranking loss module.

Figure 5.1: (a) Architecture of a BART encoder-decoder (blue-ish block) extended with an entity-sentence alignment method (green-ish block) and an entity-reference alignment method (carmine block). The two alignment methods internally utilize an adaptive margin ranking loss module (milky block) and an entity context representation module (purple-ish block). (b) Adaptive margin ranking loss module consists of two key submodules: the margin ranking loss with adaptive capacity (pinkish block) and the margin scaling variable submodule (sky-blue block). MLPs are used to reduce dimensionality.

5.2 Our Methods

Our methods are illustrated in Figure 5.1. The architecture, as shown in Figure 5.1a, consists of the backbone BART encoder-decoder, the entity-sentence alignment method (E-Sent AM), and the entity-reference alignment method (E-Ref AM). Both alignment methods utilize adaptive margin ranking loss that has the same modular structure as shown in Figure 5.1b. The two methods also share an entity context

Table 5.1: Key notation summary. We use $\cdot$ in place of item (or sample) identities for simplicity without loss of generality.

Notation	Description
$\|\cdot\|$	The total number of items or samples.
$\{\cdot\}$	An ordered set of items or samples.
$\{\cdot\}_{k=1}^{\|\cdot\|}$	An ordered set indexed by an indexing variable k up to the total number $\|\cdot\|$.
$\langle\cdot,\cdot\rangle/\langle\cdot,\cdot,\cdot\rangle$	A pair/triplet notation.
$\langle\cdot,...,\cdot\rangle$	A sequence notation.
$[\cdot,\cdot]$	A dimensional concatenation operator of latent representations.
$\langle t_p, t_q\rangle$	A sentence bounding (i.e. beginning and ending) token indices concerning a document.
e_i	An entity i.
s_{ij} (or s_{ij}^+)	A sentence j in a document where an entity i appears and/or is referenced (as a positive sentence sample).
$s_{ij'}^-$	A negative sentence sample related to an entity i, that is, a sentence j' in a document where an entity i does not appear or is referenced.
r_{ij} (or r_{ij}^+)	A reference j in the reference cluster of an entity i (as a positive reference sample).
$r_{ij'}^-$	A negative reference sample linked to an entity i, that is, a reference j' refers to a different entity from the entity i.
λ_{ij}	A margin scaling variable for the sample j of the entity i.
$\mathbb{C}^s(i)$	A cluster of sentences where an entity i appears and/or is referenced.
$\mathbb{C}^s$	A set of all entity-sentence clusters.
$\mathbb{C}^r(i)$	A cluster of references to an entity i.
$\mathbb{C}^r$	A set of all entity-reference clusters.
$\|S\|$	The total number of sentences in a document.
$\mathbf{h}_{\cdot}$	A (latent) representation of an item or sample.
$\mathbf{h}_{e_i}^c$	An entity i's (latent) context representation.
$sc_{i\cdot}^+/sc_{i\cdot}^-$	Margin ranking scoring function of positive/negative sample with respect to an anchor entity i.
$\mathscr{L}_{\cdot}$	A learning objective or loss notation.

representation module. We first discuss the generic aspects of the architecture before proceeding to the specifics of our two alignment methods.

5.2.1 Adaptive Margin Ranking Loss

Recalling margin ranking loss from Chapter 4, it can be defined as:

$$\mathscr{L}_{\text{MR}} = \max(0, sc_k^+ - sc_k^- + m) \tag{5.1}$$

where sc_k^+ and sc_k^- are positive and negative sample scores concerning an anchored entity of the k^{th} sample triplet (i.e. entity, positive and negative samples), and m is a margin hyperparameter. The function aims at distancing negative samples from positive ones concerning an anchored entity in model latent space. However, the constant scalar margin assumes that samples pose a uniform learning condition. The assumption is hardly suitable for many complex problems, including intrinsic NERHs. This gives rise to our adaptive margin ranking loss, defined as follows:

$$\mathscr{L}_{\text{AMR}} = \max(0, sc_k^+ - sc_k^- + \lambda_k \cdot m) \tag{5.2}$$

where λ_k is a scaling variable for the k^{th} sample triplet.

Before detailing its applications in our two alignment methods, we first summarize key notations used in our methods in Table 5.1 given that the two methods have a noticeable amount of similar notations.

5.2.2 Entity-Sentence Alignment Method

An entity has contextual sentences where the entity occurs and/or is referenced in a document. We can express the contextual sentences of an entity as an entity cluster as follows:

$$\langle e_i, \{s_{ij}\}_{k=1}^{|\mathbb{C}^s(i)|} \rangle, \quad s_{ij} \in \mathbb{C}^s(i),\ \mathbb{C}^s(i) \in \mathbb{C}^s \text{ and } j \in |S| \tag{5.3}$$

where e_i is the entity i, $|\mathbb{C}^s(i)|$ is the number of contextual sentences in the entity cluster $\mathbb{C}^s(i)$, $\mathbb{C}^s$ is entity cluster set, $\{s_{ij}\}_{k=1}^{|\mathbb{C}^s(i)|}$ is the ordered set of the contextual sentences, j is the sentence number in the document, and $|S|$ is the total number of sentences in the document. The contextual sentences may not be adjacent. Each sentence s_{ij} is further expressed by its bounding (i.e. beginning and ending) token index pair in the document as follows:

$$s_{ij} \doteq \langle t_p, t_q \rangle. \tag{5.4}$$

Positive Samples We construct the entity-sentence positive samples by transforming the entity and sentences from Equation (5.3) into one-to-one pairs as follows:

$$\langle e_i, \{s_{ij}\}_{k=1}^{|\mathbb{C}^s(i)|} \rangle \Rightarrow \{\langle e_i, s_{ij}^+ \rangle\}_{k=1}^{|\mathbb{C}^s(i)|} \tag{5.5}$$

where the superscription + denotes positive samples.

Negative Samples Using the positive samples defined in Equation (5.5), we construct negative sample pairs by randomly drawing the samples from the positive samples of the other entity clusters:

$$\{\langle e_i, s_{i'j'}^- \rangle\}_{k=1}^{|\mathbb{C}^s(i)|}, \quad s_{i'j'} \in \mathbb{C}^s(i'),\ \mathbb{C}^s(i') \in \mathbb{C}^s,\ i' \neq i \text{ and } j' \in |S|. \tag{5.6}$$

The superscription $-$ denotes negative samples. We thus obtain the sample triplets from Equation (5.5) and Equation (5.6):

$$\{\langle e_i, s^+_{ij}, s^-_{i'j'}\rangle\}_{k=1}^{|\mathbb{C}^s(i)|}. \tag{5.7}$$

Sample Representations We may simply use an entity's key token latent state from the encoder as its representation. To learn a representation of a sentence sample, we implement a multi-filter multi-kernel multi-scale convolutional neural network (M3-CNN) described in Appendix C.1. The resulting representations of the triplets in Equation (5.7) can then be expressed as:

$$\{\langle \mathbf{h}_{e_i}, \mathbf{h}_{s^+_{ij}}, \mathbf{h}_{s^-_{i'j'}}\rangle\}_{k=1}^{|\mathbb{C}^s(i)|} \tag{5.8}$$

where $\mathbf{h}_{e_i}$, $\mathbf{h}_{s^+_{ij}}$ and $\mathbf{h}_{s^-_{i'j'}}$ are the entity, positive and negative sentence representations respectively.

Entity Context Representation Different entities may have their contexts sharing a subset of sentences. This presents ambiguous learning conditions to ATS models. One option is to mutually exclude such sharing to train our model, but it may reduce the sufficient number of training samples required to generalize the model better. Hence, we instead take an entity context-guided learning approach discussed later to take into account the impact. This context guidance approach also allows the model to learn differences from similarity, closely analogous to human abstraction. We first learn the context representation. Using the contextual sentences in Equation (5.3) and each sentence's bounding token index pair in Equation (5.4), we can gather the token latent states of the contextual sentences from the encoder:

$$\mathbf{h}^c_{e_i,\langle . \rangle} \doteq \{\langle \mathbf{h}_{t^k_p}, ..., \mathbf{h}_{t^k_q}\rangle\}_{k=1}^{|\mathbb{C}^s(i)|} \tag{5.9}$$

where $\langle \mathbf{h}_{t^k_p}, ..., \mathbf{h}_{t^k_q}\rangle$ is the sequential token representations of the k^{th} contextual sentence. To learn the context representation of the sequence, we apply an M3-CNN model, and thus get the resulting representation as follows:

$$\mathbf{h}^c_{e_i} = \text{M3-CNN}(\mathbf{h}^c_{e_i,\langle . \rangle}). \tag{5.10}$$

With sample data ready, we can now follow Figure 5.1b to instantiate adaptive margin ranking loss. We start with a measure adaptive to learning conditions.

Context-Guided Entity Span Classification Kryscinski et al. (2020), in their FactCCX method, have developed a claim (text) span classification to identify supports and mistakes predicted by their evaluation model. Such text span classification may be regarded as a test method for learning conditions. That is, easy learning conditions are implied if a model converges the classification well. It is difficult

to learn otherwise. We extend the idea by integrating context guidance. To do so, we concatenate the entity, positive sentence representations from Equation (5.8) and the context representation from Equation (5.10), followed by a clamped linear transformation to get the entity span prediction logits as follows:

$$\text{logits}_{e_{ij}} = \min(\max(0, W[\mathbf{h}_{e_i}, \mathbf{h}_{s_{ij}^+}, \mathbf{h}_{e_i}^c]), l_{max}) \tag{5.11}$$

where W is the learning parameters, [,] is the concatenation operator, l_{max} is the maximum token sequence length that the encoder-decoder permits, and $\text{logits}_{e_{ij}} \in \mathbb{R}^{2\times 1}$ is the predictions of bounding token index pair concerning the entity.

Entity Span Distance-IoU We now need a metric on the entity span classification to give rise to a scaling variable function that satisfies the aforementioned adaptive margin properties. Intersection over union (IoU) is a well-known metric for object detection in computer vision. It measures the overlapping ratio between a predicted object and the ground truth bounding box. It is scale-invariant and bound to $[0.0, 1.0]$. Entity span in 1D shares a similar overlapping characteristic to the bounding box in 2D. We adopt a Distance-IoU by Zheng et al. (2020b) for fast learning convergence, and derive an entity span-based Distance-IoU (ESpan-DIoU) as follows.

Let the predicted $\text{logits}_{e_{ij}}$ for the entity i be split into index pair $(\hat{b}_i^1, \hat{b}_i^2)$. With the ground truth $\left(b_i^1, b_i^2\right)$, an entity span IoU is defined as:

$$\begin{aligned} f_\cap &= \max(0,\ \min(\hat{b}_i^2, b_i^2) - \max(\hat{b}_i^1, b_i^1)), \\ f_\cup &= (\hat{b}_i^2 - \hat{b}_i^1) + (b_i^2 - b_i^1) - f_\cap + \epsilon, \\ \text{IoU}_i &= \frac{f_\cap}{f_\cup} \end{aligned} \tag{5.12}$$

where $f_\cap$ and $f_\cup$ are the intersection and union of the two text spans respectively, and ϵ is a small residue to avert possible divide-by-zero. A DIoU is then derived as:

$$\begin{aligned} f_\omega &= \max(\hat{b}_i^2, b_i^2) - \min(\hat{b}_i^1, b_i^1), \\ f_\rho &= ((b_i^1 + b_i^2) - (\hat{b}_i^1 + \hat{b}_i^2))^2/4, \\ \text{DIoU}_i &= \text{IoU}_i - \frac{f_\rho}{(f_\omega^2 + \epsilon)} \end{aligned} \tag{5.13}$$

where f_ω is the smallest width enclosing the two spans, f_ρ is the squared distance of the two span centers, and f_ω^2 is the squared f_ω.

Given $\text{IoU}_i \in [0,1]$ and $\frac{f_\rho}{(f_\omega^2+\epsilon)} \in [0,1]$, $\frac{f_\rho}{(f_\omega^2+\epsilon)}$ moves toward one when IoU_i approaches zero. Vice versa. Therefore, $\text{DIoU}_i \in [-1,1]$.

Margin Scaling Variable Function We first convert the ESpan-DIoU to a loss bound by $[0.0, 2.0]$ such that it incurs higher cost when the DIoU_i moves toward -1, defined as:

$$\mathscr{L}_{\text{DIoU}_i} = 1.0 - \text{DIoU}_i. \tag{5.14}$$

To preserve a base margin at the low bound, we transform the loss to be bound by [1.0,3.0]. Therefore, the scaling variable function for the sample j of the entity i is defined as:

$$\lambda_{ij} = 1.0 + \mathscr{L}_{\mathrm{DIoU}_i}. \tag{5.15}$$

Scoring Function The aim is to measure the relevance of the sentences to the anchored entity. We consider this as a similarity measure using the cosine function. A positive sample score is thus defined as:

$$sc^{+}_{ij} = \frac{\mathbf{h}_{e_i} \cdot \mathbf{h}_{s^{+}_{ij}}}{||\mathbf{h}_{e_i}||_2 \cdot ||\mathbf{h}_{s^{+}_{ij}}||_2}. \tag{5.16}$$

By the same formula, a negative sample score $sc^{-}_{ij'}$ can be acquired with the negative sample representations $\mathbf{h}_{s^{-}_{i'j'}}$ instead.

Entity-Sentence Alignment Learning Objective Given $\{\langle sc^{+}_{ij}, sc^{-}_{ij'}, \lambda_{ij}\rangle\}_{k=1}^{|\mathbb{C}^s(i)|}$, the learning objective is computed as:

$$\mathscr{L}_{\text{ES-AM}} = \frac{1}{|\mathbb{C}^s(i)|} \sum_{k=1}^{|\mathbb{C}^s(i)|} \max(0, sc^{+}_{ij} - sc^{-}_{ij'} + \lambda_{ij} \cdot m). \tag{5.17}$$

5.2.3 Entity-Reference Alignment Method

This alignment method follows the same process as Section 5.2.2 but with different data and scoring function. We focus our discussion on the difference. Note that this method is considered a generalization of margin ranking discussed in Section 4.2.2 of Chapter 4.

Positive Samples Positive samples here consist of the annotated coreferences as in Section 4.2.2. Therefore, an entity cluster is expressed as follows:

$$\{\langle e_i, r^{+}_{ij}\rangle\}_{k=1}^{|\mathbb{C}^r(i)|}, \quad j \in |\mathbb{C}^r(i)| \text{ and } \mathbb{C}^r(i) \in \mathbb{C}^r \tag{5.18}$$

where e_i is an entity i, r^{+}_{ij} is a reference j (e.g., a pronoun), $|\mathbb{C}^r(i)|$ is the number of references in the entity cluster $\mathbb{C}^r(i)$, and $\mathbb{C}^r$ is entity cluster set.

Negative Samples We construct negative sample pairs by randomly drawing the positive reference samples from the other entity clusters:

$$\{\langle e_i, r^{-}_{i'j'}\rangle\}_{k=1}^{|\mathbb{C}^r(i)|}, \quad j' \in |\mathbb{C}^r(i')|, \mathbb{C}^r(i') \in \mathbb{C}^r \text{ and } i' \neq i. \tag{5.19}$$

The sample triplets are then given as:

$$\{\langle e_i, r^{+}_{ij}, r^{-}_{i'j'}\rangle\}_{k=1}^{|\mathbb{C}^r(i)|}. \tag{5.20}$$

Sample Representations As with the approach taken in Section 5.2.2, we may use the key token latent states from the encoder for entity referent and reference representations, respectively. Hence, the representations of the triples are expressed as:

$$\{\langle \mathbf{h}_{e_i}, \mathbf{h}_{r^+_{ij}}, \mathbf{h}_{r^-_{i'j'}} \rangle\}_{k=1}^{|\mathbb{C}^r(i)|}. \tag{5.21}$$

Context-Guided Entity Span Classification The classification may share the same representation of context guidance as in Section 5.2.2 but concatenates it with the entity and positive reference representation pairs from Equation (5.21) as follows:

$$\text{logits}_{e_{ij}} = \min(\max(0, W'[\mathbf{h}_{e_i}, \mathbf{h}_{r^+_{ij}}, \mathbf{h}^c_{e_i}]), l_{max}) \tag{5.22}$$

where W' is the learning parameters.

Scoring Function Following our early work discussed in Chapter 4, the scoring function is defined in the complex embedding space:

$$sc = \text{Re}(E_s \overline{E_o}^{\text{T}}) \tag{5.23}$$

where E_s is the referent representations, and $\overline{E_o}$ is the complex conjugate of the reference representations E_o.

Entity-Reference Alignment Learning Objective Given $\{\langle sc^+_{ij}, sc^-_{ij'}, \lambda_{ij} \rangle\}_{k=1}^{|\mathbb{C}^r(i)|}$, the learning objective is computed as follows:

$$\mathscr{L}_{\text{ER-AM}} = \frac{1}{|\mathbb{C}^r(i)|} \sum_{k=1}^{|\mathbb{C}^r(i)|} \max(0, sc^+_{ij} - sc^-_{ij'} + \lambda_{ij} \cdot m). \tag{5.24}$$

5.2.4 Total Learning Objective

The total learning objective consists of the standard MLE objective as Equation (3.1), the entity-sentence alignment objective as Equation (5.17), and the entity-reference alignment objective as Equation (5.24):

$$\mathscr{L} = \mathscr{L}_{\text{MLE}} + (\mathscr{L}_{\text{ES-AM}} + \mathscr{L}_{\text{ER-AM}}). \tag{5.25}$$

5.3 Experimental Results and Analysis

5.3.1 Dataset

As the adaptive margin ranking loss is a generalization of classical margin ranking loss and the entity-reference alignment method unifies the solution for the endophoric reference errors in Chapter 4, we continue our experiments with CNNDM and XSum, and prepare our datasets based on CR-based named entity annotations the same way as described in Section 4.3.1 of Chapter 4.

5.3.2 Implementation

Our methods adopt the pre-trained BART-base encoder-decoder as in our previous work. Our source code is accessible on GitHub[2]. The key implementation is detailed as follows.

5.3.2.1 Adaptive Margin Scaling Variables and Base Margin

To determine a base margin, we are interested in the dynamics of the margin scaling variables. Hence, we add trace logic in the source code to log the maximum and minimum values of the scaling variables per epoch during training. Our trial runs show that the value range starts from about $[2.25, 2.99]$ and converges toward the range of about $[1.99, 2.24]$ when the runs are finished on an early stop criterion. Based on the observation with the consideration of the experimented margin configurations in Chapter 4, we set our base margin at 25.

5.3.2.2 Alignment Methods

Referents and references may consist of multiple words. To simplify the learning methods without loss of representational discriminative power, we take the same super node representation learning approach as described in Chapter 4.

In addition to refactoring several functions of the knowledge graph learning framework from the package PyKeen as in Chapter 4, we also add a similarity interaction interface to the framework for the entity-sentence alignment method.

5.3.2.3 Entity Span Classification

The classification in Equation (5.11) includes sentence feature $\mathbf{h}_{s_{ij}}$. Given that the entity context vector $\mathbf{h}^c_{e_i}$ may have encapsulated the salient feature of the individual contextual sentence, we thus simplify the equation by omitting the sentence feature in the concatenated feature vector as $[\mathbf{h}_{e_i}, \mathbf{h}^c_{e_i}]$ to alleviate the constraints imposed by the shared computational resources, particularly GPU memory.

5.3.2.4 Training on Multi-GPUs

Following the training setting described in Section 4.3.2 of Chapter 4, we use AdamW optimizer and set the learning rate to $5e^{-5}$ with a linear decay, and the weight decay to $1e^{-6}$. Two shared GPU cards are used. Each of them is NVIDIA A100/80GB[3]. A configuration of Dual AMs has a model size of 581.460MB. A fine-tuning session of the Dual AMs with CNNDM on an early-stop configuration takes about 94 hours. For XSum, we first fine-tune the model without CR-derived data (i.e., without using our alignment methods). It takes roughly 26 hours on an early-stop configuration. We

[2] `https://github.com/13114386/adaptive_mrl`.

[3] As the used GPUs are shared resources, we are unable to fully utilize them for the computation of our methods on a larger pre-trained backbone model.

Table 5.2: ROUGE Evaluation (CNNDM and XSum). 1. The number of test samples from our annotation preprocessing is 11483 (out of 11490 samples) for CNNDM and 11328 (out of 11334 samples) for XSum. 2. Our summary-level R-L (equivalent to the R-Lsum ROUGE metric this research uses). 3. No results on CNNDM were available. Therefore, we use authors-published source code (`https://github.com/meetdavidwan/factpegasus`) to train and test a model following their settings, except that the maximum source length and target length are changed to 1024 and 142, respectively, to match the pre-trained BART-base configuration. 4. Entity-reference alignment method. 5. Entity-sentence alignment method. 6. Dual AMs consists of both entity-reference and entity-sentence alignment methods.

Model	**CNNDM Test Set**[1]			**XSum Test Set**[1]		
	R-1	R-2	R-L[2]	R-1	R-2	R-L[2]
QA-Span/BertSumExtAbs (Dong et al., 2020)	41.75	19.27	38.81	36.86	14.82	29.70
ERPGN/BART-Base (Lyu et al., 2022)	42.28	19.64	38.93	39.60	16.90	31.74
FactPEGASUS(Zero-shot) /BART-base (Wan and Bansal, 2022)[3]	40.98	18.97	28.90	32.97	11.42	25.41
BART-base	42.81	19.52	39.72	41.80	18.99	33.89
BART-base/E-Ref AM[4]	**43.10**	**19.82**	**40.05**	41.74	18.84	33.70
BART-base/E-Sent AM[5]	42.88	19.56	39.84	**41.99**	**19.06**	**34.00**
BART-base/Dual AMs[6]	42.81	19.50	39.74	41.87	18.91	33.78

then further fine-tune the model using the alignment methods with the CR-derived data on the same early-stop configuration for about 3 hours.

5.3.2.5 Inference Setting

We use the same inference settings as in Table 4.3 of Chapter 4.

5.3.3 ROUGE Evaluation

Table 5.2 shows the ROUGE scores for both CNNDM and XSum test sets. Separated by double lines, the top section of the table lists several recent ATS factuality research using the same or similar sized backbone models, followed by our experimental results (the ROUGE scores on high confidence interval). Our experiments have evaluated the fully configured dual alignment methods (Dual AMs) and two ablations: the entity-reference alignment method (E-Ref AM) and the entity-sentence alignment method (E-Sent AM). We have also fine-tuned the BART-base as a baseline for comparison.

Compared to prior works, the models trained on our annotated datasets have outperformed most scores on both CNNDM and XSum.

Among our experiments, the model trained with the E-Ref AM produces better

Table 5.3: SummaC score statistics over 100 randomly sampled generated summaries from CNNDM and XSum test sets, respectively.

Model	**CNNDM** (100 samples)				**XSum** (100 samples)			
	SummaC$_{ZS}$		**SummaC$_{Conv}$**		**SummaC$_{ZS}$**		**SummaC$_{Conv}$**	
	μ(%)	σ	μ(%)	σ	μ(%)	σ	μ(%)	σ
BART-base	68.3	0.255	62.5	0.230	**10.2**	0.190	23.8	0.036
BART-base/E-Ref AM	71.6	0.222	65.0	0.206	8.9	0.154	23.5	0.026
BART-base/E-Sent AM	68.4	0.239	64.5	0.201	7.0	0.123	23.5	0.029
BART-base/Dual AMs	**71.8**	0.231	**66.8**	0.200	9.9	0.185	**23.9**	0.035
Reference	48.5	0.243	45.6	0.188	6.4	0.108	23.3	0.028

scores on CNNDM, while the one trained with the E-Sent AM has the edge on XSum. Would the results also suggest that the alignment methods individually outperform the combined Dual AMs in reducing intrinsic NERHs and improving the overall factuality? To answer the question, we first conduct an automatic factuality consistency evaluation as follows.

5.3.4 Automatic Factuality Consistency Evaluation

Following our previous work, we use the SummaC for the evaluation and include SummaC$_{ZS}$ in addition to SummaC$_{Conv}$.

100 Random Samples We randomly sample 100 generated summaries, then use the metrics to score each summary in relation to its source document, followed by computing their score statistics (i.e. mean and standard deviation) as shown in Table 5.3 for both CNNDM and XSum, respectively.

It is seen that the combined Dual AMs method scores higher mean values than the alignment ablations on both CNNDM and XSum, even though the Dual AMs may have had lower ROUGE scores respectively.

Compared to the baseline BART-base, our methods achieve better scores with CNNDM. But, with XSum, the baseline has an advantage on SummaC$_{ZS}$ scores while the Dual AMs edges ahead on the SummaC$_{Conv}$. Along with the standard deviations, the results for XSum suggest that the baseline produces some summaries with higher probabilities while the Dual AMs generates more summaries in high entailment probability bins.

We also score the reference summaries for analysis. We see that the reference summaries have the lowest scores for both datasets. Recalling our observation and rationale on the SummaC$_{Conv}$ score disparity between CNNDM and XSum in Chapter 4, this low score on the reference summaries could also suggest that the human-annotated reference summaries contain a significant amount of extrinsic knowledge from the metric model perspective.

Table 5.4: SummaC score statistics over our preprocessed test sets.

Model	**CNNDM**				**XSum**			
	SummaC$_{ZS}$		**SummaC$_{Conv}$**		**SummaC$_{ZS}$**		**SummaC$_{Conv}$**	
	μ(%)	σ	μ(%)	σ	μ(%)	σ	μ(%)	σ
BART-base	72.5	0.233	68.8	0.214	9.6	0.177	23.7	0.048
BART-base/Dual AMs	**73.1**	0.231	**69.3**	0.210	**9.8**	0.178	23.7	0.048

Table 5.5: Statistical significance (by paired t-test) on SummaC scores.

Model	**CNNDM**		**XSum**	
	SummaC$_{ZS}$	**SummaC$_{Conv}$**	**SummaC$_{ZS}$**	**SummaC$_{Conv}$**
BART-base vs BART-base/Dual AMs	0.007 (<0.05)	0.003 (<0.05)	0.304 (>0.05)	0.912 (>0.05)

Statistical Significance Assessment As devised to reduce intrinsic entity hallucinations, our methods have different objectives from n-gram matching seen in pre-training language modeling. Therefore, we expect that our methods improve scores slightly on n-gram overlapping-based metrics (e.g. ROUGEs) by matching the de-hallucinated entities and references in summaries. Since entities and references are sparse and small sets in long documents, the improvement in overall automatic factuality scores is expected to be moderate. However, we would nonetheless like to evaluate SummaC on the test sets and examine the statistical significance of SummaC scores between the BART-base backbone (trained with the Dual AMs) and the BART-base baseline. Their score statistics for both CNNDM and XSum are computed and shown in Table 5.4. Using paired t-test[4], we further compute statistical significance on SummaC scores between the BART-base backbone and the BART-base baseline, as shown in Table 5.5. Our null hypothesis is that there is no significant difference between scores on the generated summaries from the two models (p-value < 0.05).

Table 5.5 shows that the results on CNNDM reject the null hypothesis and thus indicate that the scores on the generated summaries from the Dual AMs-trained backbone are significantly different from those from the baseline. Given that the backbone achieves better factuality scores than the baseline on CNNDM shown in Table 5.4, this confirms our confidence that the summaries generated from the Dual AMs-trained backbone achieve significantly better SummaC scores than those from the baseline. On the other hand, the results on XSum, by accepting the null hypothesis, indicate that there is no significant difference in scores on the generated summaries between the two models. We think that this statistical insignificance may be partly due to the one-sentence-alike conciseness of XSum summaries in addition to the relatively smaller CR-annotated training dataset. Nonetheless, it agrees with

[4]We use ttest_rel API from Python's scipy.stats package.

Table 5.6: Human evaluation of factuality on the 100 randomly sampled generated summaries for CNNDM and XSum, respectively. 1. Our fine-tuned baseline BART-base.

Error Type	**Model** (CNNDM/100 samples)				**Model** (XSum/100 samples)			
	BART-base[1]	Dual AMs	E-Ref AM	E-Sent AM	BART-base[1]	Dual AMs	E-Ref AM	E-Sent AM
Entity intrinsic	17	7	10	11	23	12	16	14
Entity extrinsic	8	7	9	6	1	1	2	0
Subtotal	*25*	***14***	*19*	*17*	*24*	***13***	*18*	*14*
Modifier	0	0	2	0	3	3	12	15
Event	3	2	4	2	19	9	8	7
Event-time	5	2	5	1	9	8	6	8
Location	0	1	0	0	10	10	11	13
Negation	0	2	0	0	2	0	1	1
Number	1	5	3	3	13	12	18	13
Misspelling	2	5	3	1	0	0	0	0
Subtotal	*11*	*17*	*17*	***7***	*56*	***42***	*56*	*57*
Total	*36*	*31*	*36*	***24***	*80*	***55***	*74*	*71*

the results on XSum in Table 5.4.

5.3.5 Human Evaluation

We further assess various aspects of factual issues that may give rise to automatic metric scores. We evaluate how well our methods reduce the NERHs of interest. We also assess a range of commonly observed syntactic agreement errors that are often the causes of hallucinations, covering event, event time, location, number, modifier, and negation. Misspelling errors are also included. Appendix C.2 details the defining rules for counting the hallucinations and errors.

The same 100 sampled summaries evaluated by SummaC are used. The erroneous occurrences for both CNNDM and XSum are shown in Table 5.6. Separated by double lines, the table contains the counts of the NERHs, followed by the syntactic agreement issues and misspellings, and then the sums of all counts.

Our alignment methods, as shown, consistently reduce the intrinsic NERHs compared to the baseline for both CNNDM and XSum: the Dual AMs reduces them considerably. Noticed that the models trained with XSum have resulted in much fewer extrinsic entity hallucinations than those trained with CNNDM. We think that the one-sentence-alike conciseness of XSum summarization might have limited exposure to the extrinsic entity hallucinations.

Meanwhile, the error counts on the CNNDM summaries show that the Dual AMs

and E-Ref AM methods introduce more syntactic agreement and misspelling errors than the baseline, while the E-Sent AM method can reduce them. The Dual AMs method has more errors on location, negation, numbers, and misspellings while the baseline results in more errors on event and event time. On XSum, the Dual AMs method results in the least syntactic agreement errors, while the baseline has much higher error counts on events. This count divergence on error types might occur because the models attend to the respective features more often.

Additionally, the models trained with XSum incur much more syntactic agreement errors than those trained with CNNDM. It is worth noting that the sampled summaries from all models have few other factual errors outside our categories. Therefore, we think that the abstractive and extrinsic nature of XSum summaries could have contributed to more errors in events, event time, locations, and numbers.

5.3.6 Extended Factuality Comparison to FactPEGASUS

We extend our factuality assessment to compare with FactPEGASUS/Zero-shot[5] on both automatic metric evaluations and manual assessment as part of the study. The extended assessment leads us to some useful findings and discussions.

5.3.6.1 Automatic Factuality Consistency Evaluation

We use SummaC to evaluate the summaries generated by the FactPEGASUS on the same 100 random samples used in Section 5.3.4. The evaluation results are $\mathbf{SummaC_{zs}}(\mu\text{-}85.0\%, \sigma\text{-}0.199)$ and $\mathbf{SummaC_{Conv}}(\mu\text{-}75.9\%, \sigma\text{-}0.193)$ on CNNDM, while $\mathbf{SummaC_{zs}}(\mu\text{-}18.9\%, \sigma\text{-}0.309)$ and $\mathbf{SummaC_{Conv}}(\mu\text{-}27.4\%, \sigma\text{-}0.150)$ on XSum. The FactPEGASUS has better SummaC scores than ours while our models present better ROUGE scores shown earlier. To better understand what underlies such a disconnected correlation between the ROUGE and SummaC scores, we further conduct our human evaluation on the same 100 summaries as follows.

5.3.6.2 Human Evaluation

The evaluation has noticed that the FactPEGASUS-generated summaries from CNNDM contain many broken sentences. That is, the sentences are suddenly clipped in a way that they finish with words such as prepositions, pronouns, or determiners dangling at the end with missing information. We consider that the phenomenon is important because the clipped endings might avert various hallucination occurrences otherwise but can also miss out on important facts and leave the summaries incomplete and less comprehensible. We thus include the assessment of the phenomenon. The results are shown in Table 5.7.

[5]It is worth noting that FactPEGASUS/Zero-shot uses FactCC to score important pseudo-summary sentence selection for masked pre-training and FactCC was trained with CNNDM-derived data. As we use the authors' published source code and dataset to fine-tune and evaluate FactPEGASUS zero-shot on CNNDM, the zero-shot requirement is thus relaxed.

Table 5.7: Human evaluation and comparison of FactPEGASUS factuality on the 100 random samples.

Error	**CNNDM**		**XSum**	
Type	FactPEGASUS	Dual AMs	FactPEGASUS	Dual AMs
Entity intrinsic	13	7	21	12
Entity extrinsic	0	7	0	1
Subtotal	***13***	*14*	*21*	***13***
Modifier	2	0	14	3
Event	0	2	5	9
Event-time	0	2	2	8
Location	0	1	7	10
Negation	0	2	1	0
Number	0	5	4	12
Misspelling	0	5	0	0
Broken sentence	44	0	2	0
Subtotal	*46*	***17***	***35***	*42*
Total	*59*	***31***	*56*	***55***

Compared to the FactPEGASUS on CNNDM, the Dual AMs method performs better on the intrinsic named entity-related hallucinations (NERHs), while the FactPEGASUS has much fewer extrinsic entity hallucinations, syntactic agreement, and misspelling errors. But the FactPEGASUS incurs a significant number of broken sentences. This could explain its lower ROUGE scores but higher SummaC scores on factuality consistency in that the clipping sentence phenomenon may degenerate the n-gram overlapping matches. On the other hand, it could benefit the factual entailment assessment in latent semantic space because of less noise and fewer outliers, even though it might lose many facts at the same time. This might also partially explain its few extrinsic entity hallucinations.

Comparing the models on XSum, we see that the Dual AMs also performs much better on the reduction of intrinsic NERHs than the FactPEGASUS, while the FactPEGASUS results in fewer syntactic agreement errors. Meanwhile, the FactPEGASUS has much higher errors in the modifier category. Given that XSum has concise summaries, the FactPEGASUS creates far fewer broken sentences. Overall, the Dual AMs method shows better performance in terms of factuality consistency, particularly the intrinsic NERHs.

5.3.6.3 Discussion

The analysis of the ROUGE results, SummaC scores, and human evaluation has indicated that there is a correlational disparity between ROUGE scores and factuality consistency results. The comparison has provided us further insights into how various factual and syntactic errors might underpin the ROUGE and SummaC scores. We see how the generated summaries with the clipping phenomenon might

result in low n-gram overlap-based metric scores but high factual entailment metric results. It indicates to us that the automatic metrics can be a 'bias' in the model's generative capacity by not knowing the underlying conditions and phenomena of the generated summaries. Going forward, we believe that the ATS factuality research needs a systemic approach to understand the correlations of various quantitative metric scores concerning a wide range of factual and syntactic phenomena. Gaining such a holistic understanding of the correlations is going to open us up to new approaches to factuality modeling and evaluation metric development.

5.4 Summary

This chapter addresses named entity-related hallucinations in model-generated summaries. An adaptive margin ranking loss method has been developed to deal with entity-related hallucinations in complicated global contexts where variant learning conditions are implicit in data. This adaptive loss has facilitated two entity alignment methods to tackle entity-entity and entity-reference hallucinations.

The experimental results have shown the efficacy of our approach in mitigating entity-related hallucinations. Specifically, automatic factuality consistency metric evaluation has shown that our entity alignment methods empowered by the adaptive margin ranking loss noticeably improve the model on the dataset of characteristically intrinsic reference summaries. Human evaluation on categorized syntactic agreement errors has further shown that our methods improve the trained models to reduce the total number of entity-related hallucinations compared to the baselines on both datasets of intrinsic and extrinsic reference summaries.

CHAPTER 6

INFORMATIVE ATTENTION GUIDED BY NAMED ENTITY SALIENCE FOR IMPROVING INFORMATIVE FACTUALITY OF ABSTRACTIVE TEXT SUMMARIZATION

6.1 Research Problem

Existing ATS modeling concerns generating summaries that are fluent, coherent, relevant, and consistent (e.g., Kryscinski et al., 2019) as commonly accepted criteria for ATS. Among these criteria, the relevance concerns a model-generated summary relevant to its source document. This is achieved primarily attributed to cross-attention mechanisms devised in encoder-decoder modeling. For example, in the prevalent Transformer-based ATS encoder-decoders, the cross-attention mechanisms intuitively assign each decoded token latent state with a distribution over the source document sequence token latent states. This teaches models to generate a summary correlated to the source document. Although it is essential, this form of learning relevance may not always be optimal for informativeness from a reference summary viewpoint because it could ignore focal information in reference summaries that are informative but less statistically correlated with the source document otherwise, for example, extrinsic knowledge. This inadequacy is evidenced in generated summaries that often contain less informative content, and it is not uncommon to miss out on substantial information. For example, given the following elliptically-shortened source document,

"... But it is not impossible that our decision to leave the European Union

> *could end up being judged in the European Court in Luxembourg. ...could, in theory, fight to keep us in their clutches. ...”*,

and its reference summary,

> *“Irony of ironies, is it possible that the European Court could block us from leaving the European Union?”*,

a pre-trained model may generate a summary on the same source document as follows,

> *“If you are an ardent Brexiteer, stop before you crack open the champagne.”*.

Clearly, the model-generated summary is much less informative compared to the reference summary. We think that this is an overlooked aspect of ATS factuality because the less informative summaries could deplete the crucial facts and could even be misinformative at worst.

Prior work for improving the informativeness of ATS typically integrates topic-modeling approaches from statistical topic extraction methods, such as Latent Dirichlet Allocation (e.g., Narayan et al., 2018; Huang et al., 2020) and Poisson Factor Analysis (e.g., Wang et al., 2020), to neural topic models, such as variational autoencoder (VAE)-based models (e.g., Fu et al., 2020; Ma et al., 2022) and language model-based methods (e.g., Zheng et al., 2020a). However, topic modeling commonly imposes a fixed number of topics. This presumption makes topic modeling-based ATS sensitive to the number of topics and affects the applicability of the learned ATS models.

This chapter presents a very different learning approach, consisting of an optimal transport-based informative attention method and an accumulative joint entropy reduction method on named entities. Our approach has the following advantages over topic modeling-based prior work. We integrate our methods into ATS modeling in end-to-end training seamlessly, whereas ATS modeling integrated with topic modeling is a two-stage process. Our approach is free of the aforementioned limitations imposed by topic modeling in ATS modeling. Our learning approach reconciles the reference summary perspective complementary to source document relevance in training, while prior work focuses on topics in the source documents.

6.2 Our Methods

Figure 6.1 illustrates our learning approach that extends an encoder-decoder with our optimal transport-based informative attention and accumulative joint entropy reduction methods.

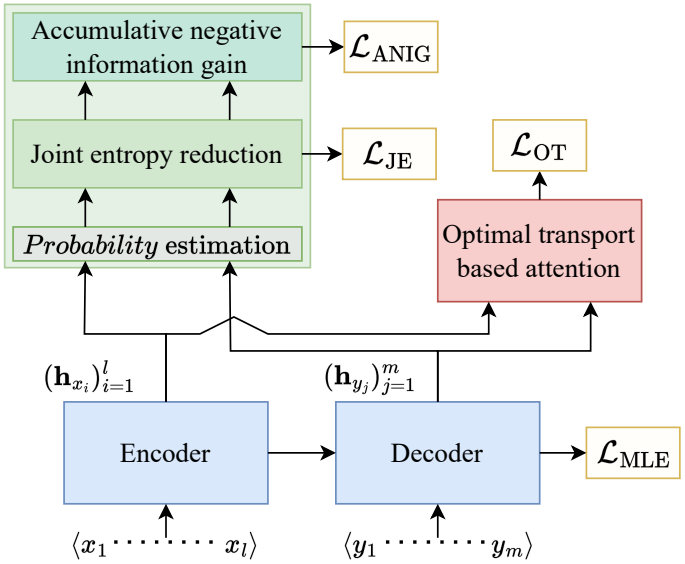


Figure 6.1: Illustration of an encoder-decoder with our methods, including the optimal transport-based informative attention (carmine block) and the accumulative joint entropy reduction (tealish block).

6.2.1 Optimal Transport-Based Informative Attention

Learning summary generation should be not only relevant to source documents but also informative concerning the reference summaries. This prompts the idea of reverse cross-attention. Expanding on the gained knowledge and promising experimental results of Wasserstein distance-based semantic similarity learning in Chapter 4, we devise an informativeness regularization as the optimal transport (OT) problem using Kantorovich's formulation (e.g., Kolouri et al., 2017) that aims at finding an optimal transport plan. The metric for the transport plans is Kantorovich distance (or Wasserstein distance). In the context of improving informative summaries, it is about moving semantic information in such a way that the source document content only 'interested' by the summary distribution is retained for summary generation. This is the reverse direction of moving information described in Section 4.2.1 of Chapter 4.

For an l-length source document and the corresponding m-length summary, the discrete optimal transport problem, as a minimizer, can be expressed as follows:

$$K(x,y) = \min_{\pi} \sum_{i=1}^{l} \sum_{j=1}^{m} \pi(x_i, y_j) c(x_i, y_j),$$

(6.1) subject to

$$\sum_{j=1}^{m} \pi(x_i, y_j) = \pi(x_i), \sum_{i=1}^{l} \pi(x_i, y_j) = \pi(y_j), \pi(x_i, y_j) \geq 0$$

where $\pi(\cdot,\cdot)$ is the joint distribution, and $c(\cdot,\cdot)$ is the cost function. The joint distributions achieving infimum are called optimal transport plans.

To realize an optimal transport, we first formulate the cost function as follows. Given the latent states (or representations) of the source document $\mathbf{h}_x$ and the corresponding summary $\mathbf{h}_y$, the cost function is defined as a pairwise L2 norm:

$$c_{(x_i,y_j)} = ||\mathbf{h}_{x_i} - \mathbf{h}_{y_j}||_2. \tag{6.2}$$

For each source document token $\mathbf{h}_{x_i}$, its distances to the summary tokens $\{\mathbf{h}_{y_j}\}_{j=1}^{m}$ are computed.

Formulating the joint distribution (the plan) using a coupling method is critical to learning an optimal transport that maximizes information concerning short reference summaries. The bilinear formulation used in our early work has the orthogonal property. In this context, it means that a source token representation similar to the summary token representation(s) would become salient, whereas it would be suppressed. This provides a simple yet effective coupling method to learn the focal information from the source document relevant to the summary. Therefore, we formulate the joint distribution based on the bilinear transformation of (x, y) in representational expression as:

$$\mathrm{T}_{i,j} = \mathbf{h}_{x_i} W_{i,j} \mathbf{h}_{y_j} \tag{6.3}$$

where $W_{i,j} \in W$ is the learning weight of the coupling pair. The joint distribution is then defined as:

$$\pi_{(x_i,y_j)} = \frac{e^{z_{i,j}}}{\sum_{j'} e^{z_{i,j'}}},\ z_{i,j} \in \mathrm{T}_i \tag{6.4}$$

where a probability distribution over the summary token representations is computed for each source document token representation. Moving information in this direction is not symmetric to the one in the opposite direction as in Chapter 4, and learns different focal information. Substituting the terms in Equation 6.1 with Equation 6.2 and Equation 6.4, we derive the informative attention learning objective as an average of the transport plan minimizer:

$$\mathscr{L}_{\mathrm{OT}} = \frac{1}{l \times m} K(x, y). \tag{6.5}$$

6.2.2 Accumulative Joint Entropy Reduction

The method gives rise to informative guidance via named entity latent salience to facilitate informative attention learning. Named entities[1] provide the focal points since document content is commonly laid out through them. Increasing the salience of named entities implies reducing their uncertainty from an information theory viewpoint.

[1]We follow the classification of named entity recognition.

A named entity often consists of multiple words. Furthermore, token encoding schemes like byte-pair encoding (Sennrich et al., 2016) employed in ATS modeling may transform a word into multiple tokens. This means that our method has to reduce the uncertainty of the named entity tokens collectively as well as the uncertainty in their sequential structure. Our method tackles both uncertainties holistically, detailed as follows.

Uncertainty is measured by (Shannon) entropy in information theory. The higher entropy, the higher uncertainty[2]. The information theory (e.g., Polyanskiy and Wu, 2022) also states that conditioning reduces entropy (or uncertainty). Furthermore, under any stationary process $X = (X_1, ...X_n)$, it satisfies:

$$H(x_n|X_{1:n-1}) \leq H(x_{n-1}|X_{1:n-2}). \tag{6.6}$$

Named entities follow a stationary process and are conditional. This suggests that each conditional named entity token prediction should be no greater uncertain than that at the previous step. The empirical study by Xu et al. (2020) also supports the theory in the context of ATS. This permits the formulation of our uncertainty reduction for named entities as follows.

Let us express a named entity token sequence as $x_{1:n}$, and assume the sequence is a Markov chain for notation simplicity. The conditional entropy of a named entity token can be expressed as:

$$\begin{cases} H(x_i|x_{i-1}), & i > 1 \\ H(x_i), & i = 1 \end{cases}. \tag{6.7}$$

To simplify our discussion without loss of generality, we introduce notations $H(x_0) = 0$ and $H(x_1|x_0) = H(x_1)$. We can thus express a conditional uncertainty inequality as:

$$H(x_{i+1}|x_i) \leq H(x_i|x_{i-1}), \quad 1 \leq i < n. \tag{6.8}$$

After rearranging, we obtain an adjacent step entropy difference as follows:

$$\triangle H(x_i, x_{i+1}) = H(x_{i+1}|x_i) - H(x_i|x_{i-1}). \tag{6.9}$$

We call this entropy reduction an adjacent information gain. Meanwhile, following the chain rule for entropy, the joint entropy of the sequence can be defined as:

$$H(x_{1:n}) = \sum_{i=1}^{n} H(x_i|x_{i-1}). \tag{6.10}$$

From Equation 6.9 and Equation 6.10, we derive a joint information gain:

$$\triangle H(x_{2:n}) = \sum_{i=2}^{n} (H(x_i|x_{i-1}) - H(x_{i-1}|x_{i-2})). \tag{6.11}$$

[2]We use entropy and uncertainty exchangeably when there is no ambiguity.

Maximizing this quantity reduces the relative uncertainty in the conditional prediction of the generative sequence. This maximization quantity is translated into a negative joint information gain for minimization training:

$$\mathscr{L}_{\text{NIG}} = -\triangle H(x_{2:n}). \tag{6.12}$$

Following the conditioning entropy theory, we expect the relative uncertainty to be higher at the beginning of an entity token sequence and lower toward the end of the sequence. It would be reasonable to apply hyperparameters to weight each term in the summation to regularize the uncertainty reduction for efficient learning. However, searching for proper hyperparameters could be time-consuming. Instead, we extend Equation 6.12 to an accumulative formulation, named accumulative negative information gain:

$$\mathscr{L}_{\text{ANIG}} = -\frac{1}{n}\sum_{i=2}^{n}\sum_{j=2}^{i}\triangle H(x_{2:j}). \tag{6.13}$$

This has the effect of penalizing more in the beginning and then weighting the penalty less gradually. Equation (6.13) encourages the learning model conditionally to choose the next token that is less uncertain than the current token. It thus reduces the uncertainty in token sequential structure. However, it may not guarantee to be efficient in reducing the absolute uncertainty of choosing a named entity. This can be amended by using the joint entropy as a regularization term too, defined as:

$$\mathscr{L}_{\text{JE}} = \frac{1}{n}\sum_{i=1}^{n} H(x_i|x_{i-1}). \tag{6.14}$$

Equation 6.13 and Equation 6.14 form two components of the accumulative joint entropy reduction (AJER). The double-dose formulation safeguards the integrity of named entity estimations. The AJER can be applied to named entities in both source documents and summaries due to the conditionality inherent in the entities. The autoregressive nature of the decoder further makes the conditionality model-explicit on summary entities.

6.2.3 Total Learning Objective

Composing of Equation 3.1, Equation 6.5, Equation 6.13, and Equation 6.14, we arrive at the total learning objective:

$$\mathscr{L} = \mathscr{L}_{\text{MLE}} + (\alpha_{\text{OT}}\mathscr{L}_{\text{OT}} + \alpha_{\text{ANIG}}\mathscr{L}_{\text{ANIG}} + \alpha_{\text{JE}}\mathscr{L}_{\text{JE}}) \tag{6.15}$$

where α_{OT}, α_{ANIG}, and α_{JE} are hyperparameters to control objective influences.

Table 6.1: Preprocessed dataset sizes.

Dataset	**CNNDM**	**XSum**
Train	196735	186589
Validation	13367	11316
Test	11490	11334

6.3 Experimental Results and Analysis

6.3.1 Dataset

We use the named entity recognition (NER) parsing tool from Stanford CoreNLP (v4.4.0) to acquire named entity annotations from the CNNDM and XSum training datasets respectively.

The named entity annotation data includes entity names and their word positions with respect to their sentences and documents. The annotated NER entities used for this research include TITLE, PERSON, ORGANIZATION, NATIONALITY, RELIGION, IDEOLOGY, DEGREE, DATE, TIME, DURATION, LOCATION, CITY, STATE_OR_PROVINCE, COUNTRY, NUMBER, MONEY, PERCENT, ORDINAL, CAUSE_OF_DEATH, CRIMINAL_CHARGE. To minimize the possible use of erroneous annotations, we only include entities whose length is not greater than ten after analyzing the entities. We thereafter transform them into our training datasets of both CNNDM and XSum respectively.

As mentioned in our early works, the Stanford CoreNLP uses Penn Treebank, which may segment words, to produce the parsing tree for NER annotation. Therefore, our training source documents and reference summaries are built from the annotated word sequences. We also preprocess the token encodings[3] of the annotated training documents and summaries for runtime efficiency as discussed in the early chapters. To map the word-level annotated named entities to the model-encoded tokens during fine-tuning, a word-token map is created for each annotated source document and summary.

The token encodings of validation and test datasets (CNNDM and XSum) are also prepared without annotation. Table 6.1 lists the dataset sizes.

6.3.2 Implementation

As before, we have adopted BART as our backbone encoder-decoder. However, we are able to run the BART implementation with the pre-trained BART-large weight profiles[4] as our methods demand relatively less computational memory resources,

[3] `https://huggingface.co/transformers/v4.9.2/model_doc/bart.html#transformers.BartTokenizer`.

[4] `https://huggingface.co/facebook/bart-large-cnn` for CNNDM, and `https://huggingface.co/facebook/bart-large-xsum` for XSum.

particularly GPU memory. The models for both CNNDM and XSum weight profiles have 12 layers, 16 attention heads, a filter size of 4096, and hidden state dimensions of 1024. Note that the model evaluation is conducted solely on the trained backbones since our methods are only used to train the backbone models. The pre-trained BART-large models are also used as baselines for evaluation comparisons. Our source code is accessible on GitHub[5]. The key implementation details are given as follows.

6.3.2.1 Initialization of Extended Modules

Optimal Transport Plan The bilinear transformation weight matrix is initialized to 1.0.

Entity Probability Estimation for AJER For named entities in summaries, we directly use the decoder's output logits in formulating AJER. For named entities in source documents, we clone the BART decoder's logits head (a linear transformation layer) and initialize it using the BART implementation's weight initialization function. Note that we do not share the logits linear layer of the decoder for the source document entity probability estimations because it has been found that the sharing is detrimental to summary generation.

Learning Objective Weights All learning objective weights α_{OT}, α_{ANIG}, and α_{JE} are default to 1.0 in our experiments.

6.3.2.2 Index Mapping from Named Entity Words to Encoded Tokens

As discussed in our early works, the annotated word-level named entities need to be mapped to the model-encoded entity tokens. Therefore, similar to our early works, we map entity words to entity tokens during fine-tuning by utilizing the word-token map as prepared in Section 6.3.1.

6.3.2.3 Dropout

Dropout is a widely adopted technique to mitigate overfitting. The pre-trained large BART models on both CNNDM and XSum weight profiles have the activation dropouts turned off. We thus turn on the model activation dropouts by setting them to 0.1 for fine-tuning the models with our methods, following the dropout configuration of the pre-trained BART-base used in our early works.

6.3.2.4 Single Token Entity Case for Negative Information Gain

In such cases, we skip them with no incurring costs.

6.3.2.5 Early Stop Fine-Tuning Criterion

Fine-tuning sessions on BART-large backbones are time-consuming and subject to shared computational resource scarcity. However, we would like to take model

[5] https://github.com/74808917/ozizAxRminf6Zyv.

Table 6.2: ROUGE evaluation. CNNDM: 11490 samples. XSum: 11334 samples. 1. Our summary-level R-L (equivalent to the R-Lsum ROUGE metric this research uses). 2. T-BERTSUM(ExtAbs) for CNNDM, and T-BERTSUM(Abs) for XSum.

Model	**CNNDM Test Set**			**XSum Test Set**		
	R-1	R-2	R-L[1]	R-1	R-2	R-L[1]
BertSUM+TA (Wang et al., 2020)	43.06	20.58	39.67	39.77	17.39	32.39
T-BERTSUM(ExtAbs/Abs) (Ma et al., 2022)[2]	43.06	19.76	39.43	39.90	17.48	32.18
GATSum (Jiang et al., 2022)	44.46	21.32	39.84	44.60	21.53	36.66
TAS (Zheng et al., 2021)	44.38	21.19	41.33	44.63	21.62	36.77
KTOPAS (Khanam et al., 2021)	42.10	20.01	38.45	N/A	N/A	N/A
PEGFAME (Aralikatte et al., 2021)	42.95	20.79	39.90	45.31	22.75	37.46
BART-large (Lewis et al., 2020)	44.16	21.28	40.90	45.14	22.27	37.25
BART-large/OT	44.67	21.16	41.59	45.16	21.69	36.54
BART-large/AJER	44.67	21.31	41.58	44.83	21.51	36.17
BART-large*/OT+AJER	**44.75**	**21.54**	**41.69**	45.08	21.58	36.22

checkpoints that give us good indications on the performance of test inference later while avoiding the run-out-of computational time budget. To this end, we use ROUGE metrics to evaluate models on the validation datasets each epoch during fine-tuning as in our early works, but we take model checkpoints on the highest validation ROUGE scores if the following three or four consecutive validation runs do not produce ROUGE scores coming out on top.

6.3.2.6 Fine-Tuning on Multi-GPUs

We continue the utilization of the multi-GPU running procedure developed in our early works, including the optimizer settings where the AdamW optimizer, the learning rate $5e^{-5}$ with a linear decay, and the weight decay $1e^{-6}$ are used.

The models are trained with our methods on two-GPU parallelism. Each GPU card is NVIDIA RTX8000/48GB. We set the total mini-batch size to 64 with 4×8 (mini-batch×gradient accumulation step)/per GPU. A configuration of OT+AJER has a model size of 1750.418MB. As models are trained on shared computational resources, the training time may vary considerably due to GPU utilization loads. At light GPU load, a training iteration may take about one second.

6.3.2.7 Inference Setting

We use the same inference settings as in Table 4.3 of Chapter 4.

6.3.3 ROUGE Evaluation

We first show ROUGE results in Table 6.2. The table includes prior work for comparison, starting with several recent topic-aware ATS research outcomes that have achieved high ROUGE scores, followed by the published BART-large ATS results.

Table 6.3: QuestEval mean score statistic over the summaries generated from the CNNDM and XSum test sets (measured against reference summaries).

Model	**QuestEval**	
	CNNDM μ(%)	XSum μ(%)
BART-large	46.5	45.1
BART-large*/OT+AJER	**47.0**	44.7

Then, separated by double lines are our experiments, which include the ablation study results of OT and AJER. It is seen that our OT+AJER-trained backbone achieves the best results over prior work on CNNDM, while the OT+AJER-trained backbone on XSum also has competitive results. We think that the characteristics of the XSum dataset may have an impact on the performance of our methods. Later, our human evaluation explores the impact and gives rise to plausible reasons behind the results.

Ablation Study The ablation studies train our backbone models with OT and AJER, respectively. Comparing the ablation results shown in Table 6.2, we see that the OT+AJER-trained model achieves better results on CNNDM, while the OT-trained model edges ahead on XSum. We will also explore the plausible reasons for the observed results here in our human evaluation.

6.3.4 Automatic Factuality Consistency Evaluation

Factuality consistency metrics are possibly the most relevant choice for measuring informativeness semantically. Several automatic factuality consistency metrics have been developed over the years (e.g., Kryscinski et al., 2020; Durmus et al., 2020; Scialom et al., 2021). Given our methods reconciling with the reference summaries, question-answering (QA) model-based metrics are proper for the purpose. Among them, we choose QuestEval[6] (Scialom et al., 2021) configured with its summarization task in that it is more robust for reference-less assessment and measures important information. This is important since we evaluate model-generated summaries against the golden references as sources. We compare our OT+AJER-trained backbone with the BART-large baseline given the pre-trained baseline implementation is accessible and used as our backbone model. The mean scores are shown in Table 6.3. The results show that the OT+AJER-trained model performs better than the baseline on CNNDM while the baseline edges ahead on XSum. The results are in agreement with the ROUGE evaluation in terms of relative performances between the two models on the two datasets respectively.

[6] https://github.com/ThomasScialom/QuestEval.

Table 6.4: Human evaluation of better informativeness on randomly drawn samples of generated summaries. The evaluation randomly draws 60 samples on CNNDM and XSum, respectively.

	CNNDM (60 samples)		**XSum** (60 samples)	
Model	No.	%	No.	%
BART-large	11/60	18.33	9/60	15.00
BART-large*/OT+AJER	**18**/60	**30.00**	**19**/60	**31.66**

6.3.5 Human Evaluation

To further ascertain auto-metric evaluation results and grasp a better understanding of plausible underlying reasons, we[7] conduct our human evaluations on both informativeness and factuality.

Informativeness Evaluation This is a challenging evaluation in that model-generated summaries may contain extrinsic information not evidenced in the articles and reference summaries. They may contain factual errors too. Some of the errors may obscure informativeness more than others. It is also challenging because reference summaries could occasionally be uninformative or erroneous. Therefore, we set out a few key rules to make the evaluation more objective, including taking into consideration factual error impact on the informative explication, and consolidating reference summaries with the information their source articles convey instead of relying on reference summaries solely. Additionally, the evaluation excludes the consideration of the extrinsic information in model-generated summaries that are not evidenced in both their source documents and reference summaries. The extrinsic information will be assessed in the human evaluation of factuality shortly. The evaluation guidelines are detailed in Appendix D.1.1. Note that we 'normalize' whitespaces in model-generated summaries and detail the reasons in Appendix D.1.2.

To facilitate the evaluation, we have developed a user interface-based evaluation tool as illustrated in Appendix D.1.3. Following the similar approach discussed in Section 4.3.6, the main user page of the tool presents an article, a reference summary, and the model-generated summaries, followed by informativeness choice buttons. An annotator can either select the better informative summary or rank the summaries as 'tie'. 60 random samples from the respective CNNDM and XSum test sets are used. A total of 120 samples are shown to an annotator page-by-page. The tool presents generated summaries randomly shuffled to ensure the evaluation is as fair as possible. The user interface also does not indicate which dataset a sample is drawn from. The tool records an annotator's choices in a back-end database from which the queried analysis is conducted upon completion of the evaluation.

[7]The author conducts the evaluation.

Table 6.5: Human evaluation of factuality on the same 60 randomly sampled generated summaries (CNNDM and XSum respectively). 1. The entity extrinsic covers extrinsic person names, events and locations. 2. BART-large baseline. 3. BART-large*/OT+AJER. 4. five are factual, two are erroneous, and one is inconclusive.

Error Type	**CNNDM** (60 samples)		**XSum** (60 samples)	
	Baseline[2]	Ours[3]	Baseline[2]	Ours[3]
Entity extrinsic[1]	1	1	2	8 (5+2+1)[4]
Name	2	1	6	5
Event	0	0	1	1
Event-time	0	0	1	1
Location	0	1	3	2
Number	0	0	10	8
Other	3	3	11	5
Total	*6*	*6*	*34*	***30 (25)***

The results are shown in Table 6.4. It is clear that the OT+AJER-trained models generate more informative summaries than the baselines on both CNNDM (by ~11%) and XSum (by ~16%), respectively. The resulting disparities between our OT+AJER-trained backbones and the pre-trained baselines could be a good indication of sampled token distributional shifts.

Factuality Evaluation As we expect that our methods could alter the generative distributions considerably, we are curious about what it actually means to ATS factuality. We thus conduct the human evaluation on factuality to answer the question and further obtain some insights supporting the answers. This is non-trivial as factual issues or hallucinations can have various forms (e.g., Maynez et al., 2020; Tang et al., 2023a). Our early work in Chapter 5 has demonstrated that the human evaluation of factuality using syntactic agreement categorization gives a better understanding of hallucination forms and plausible underlying causes. We follow the same approach with tailoring to our NER-related focus, and assess the same 120 samples used in the informativeness evaluation.

The results are shown in Table 6.5. It is noticeable that our OT+AJER-trained backbone incurs a considerable number of extrinsic entity issues on XSum that are not seen in either corresponding source documents or reference summaries. As extrinsic entities could be factual (e.g., Cao et al., 2022a), we are interested in knowing how many of these extrinsic issues are factual. After investigating using external resources (e.g., Google search), it turns out that five of them are factual and informative, two are erroneous, and one is inconclusive. Taking into account the five factual extrinsic entities, we see the overall factual error counts reduced to nine fewer errors than the baseline on XSum.

The finding suggests that our methods may improve the model of extrinsic data

mining within a dataset, whereas extrinsic data mining typically resorts to external knowledge bases. We hypothesize that the salience boosted by the AJER method helps the OT method to effectively 'build' correlational linkage between entities in source documents and extrinsic entities in reference summaries in a global context, both document-wide and corpus-wide. The hypothesis would be more affirmative if the summaries generated by the OT-trained model had fewer factual extrinsic entities. To this end, we analyze the extrinsic entities in the summaries generated by the OT-trained model on the same numbered random samples. We find five extrinsic entity issues, among which two are factual and three are erroneous. This result supports our hypothesis. These findings might partially explain the ROUGE results on XSum in Table 6.2 where the scores from the OT+AJER-trained model are lower than those of the OT-trained model and some prior works. They could similarly reason the results on XSum in Table 6.3. We think that the model's improved ability to utilize globally learned extrinsic knowledge in composing summaries gives rise to a capacity closely analogous to human intelligence in writing summaries using extrinsic knowledge.

Additionally, the summaries generated from our trained model incur fewer categorized syntactic agreement errors on XSum than those by the baseline, including person name, location, number, and 'Other' categories. We think this error reduction across categories may also be attributable to the integrated AJER method that improves the salience of named entities in their contexts in the model latent space and thus reduces the uncertainty in their conditional probability estimations.

Meanwhile, Table 6.5 shows that the OT+AJER-trained model also fends off hallucinations well on CNNDM, on par with the baseline. The rationale given to the result analysis on XSum above is also applicable to the results on CNNDM. The difference is only that the entities in both source documents and reference summaries of CNNDM tend to be from the same distribution. Thus, incurring much less extrinsic information may explain the better scores resulting from the OT+AJER-trained model on CNNDM in Table 6.2 and Table 6.3 because of more gram overlapping lexically and being similar semantically in the model latent space between the OT+AJER-trained model-generated summaries and the reference summaries respectively.

We also see that the models trained on CNNDM have produced much fewer errors than the models trained on XSum. This indicates that pre-trained ATS models like BART-large are more effective at learning from the extractive and intrinsic dataset than from the abstractive and extrinsic dataset. Our approach may have provided a data-efficient way to reduce the gap, given the OT+AJER-trained model is able to reduce the number of errors on XSum compared to the baseline.

We provide the five extrinsic but factual entity-related summaries found in the human evaluation of factuality on XSum in Table 6.6.

Table 6.6: Extrinsic but factual examples (from the 60 model-generated random samples on XSum). We use ellipses to omit the long content that is irrelevant to the discussion. The extrinsic but factual entities are highlighted with blue-colored underlines. For legibility, we replace the Unicode '\u00a3' with £, and '\n' with the Latex's newline format command.

Source	**Text**
Article	In only his second season with Porsche, Webber and his two teammates, German Timo Bernhard and New Zealander Brendon Hartley, need a fourth-place finish or better in the 6 Hours of Bahrain on Saturday to complete a remarkable run of success for car number 17. ... Webber, 39, cannot remember such a hot streak of form beyond his days in junior categories and admits he never expected to be in contention so quickly at this stage of his return to sportscars. "I'm very surprised we're in this position," he said. "We had a few tough races at the start of the championship and now here we are in Bahrain ready to close the championship off hopefully. "It started in an incredible June where we got two cars home at Le Mans and had a 1-2 against Audi. "Since then we have won every race and I don't think we envisaged that - the aggressive mentality of the F1 approach." ...
Reference	Almost exactly five years after Mark Webber lost his best chance of becoming Formula 1 world champion, the Australian is on the brink of fulfilling his title dream in the World Endurance Championship this weekend in Bahrain.
BART-large	Former Red Bull driver Mark Webber says he is "surprised" his Porsche team are in such a strong position to win the world championship.
BART-large* /OT+AJER	Former Red Bull driver Mark Webber says he is surprised how good a run of form his Porsche team have been in this year's World Endurance Championship is as he aims to close the gap on championship leader Nico Rosberg.

Continued on next page

Source	Text
Article	Levein, the club's director of football, described finishing fifth in the Premiership, as "frustrating and a tad disappointing". "I've got high hopes for him [Cathro]," Levein said. "This is a tough place to manage, as I know myself. "The thing about here is the players like him." Levein points to mistakes in the January transfer window which saw Hearts sign nine players - six of whom have since left the club - as a chief reason for the underwhelming league campaign. ... "Rushed is probably a fair assessment," Levein explained. ... "For him to be a real success here we need to give him time and let him understand what this is all about," Levein explained. ...
Reference	Craig Levein retains "high hopes" for Ian Cathro's Hearts tenure but says the club will seek to "repair the damage" of recent transfer windows.
BART-large	Hearts head coach Ian Cathro has been "rushed" into taking charge of the club, according to director of football Craig Levein.
BART-large* /OT+AJER	Hearts director of football Craig Levein says head coach Ian Cathro has been "rushed" into taking charge at Tynecastle.
Article	Killie moved eight points clear of bottom side Dundee United with a 3-0 win over St Johnstone on Saturday. ..."You want to be part of that and it gives me the chance to attract the kind of players I would like to make sure the club isn't in this situation again. "The club has been dicing with survival for many seasons and that needs to change." ...
Reference	Manager Lee Clark is urging Kilmarnock to build on the first win of his tenure and make sure they are part of an exciting top flight next season.
BART-large	Hibernian manager Lee Clark hopes his side can regain the "wow factor" of the Scottish Premiership.
BART-large* /OT+AJER	Manager Lee Clark hopes Kilmarnock can help restore the "wow factor" to the Scottish Premiership as they bid to avoid relegation.

Continued on next page

Source	Text
Article	Property developers Michael and John Taggart are offering a settlement where they would repay less than 1p for every pound they owe. That arrangement, known as an Individual Voluntary Arrangement (IVA), would allow them to avoid bankruptcy. ...
Reference	Two County Londonderry brothers facing bankruptcy owe their creditors up to £213m, the High Court has been told.
BART-large	Creditors of the Taggart family are set to vote on a proposal that would allow them to avoid bankruptcy.
BART-large* /OT+AJER	Two of Northern Ireland's biggest businessmen are seeking to avoid bankruptcy by offering a voluntary repayment plan to their creditors.
Article	Jack Beales, 93, of Rhyl, carried out the attacks while she was a young girl, mainly when he was in his 70s. ... The court heard the abuse began when she was aged six and continued for about 10 years. ... But she said it had not and she was still on strong medication for depression and anxiety, and was awaiting counselling. ... Det Insp William Jones, of North Wales Police, said: "John (Jack) Beales was a manipulative and depraved sexual predator who committed multiple offences over a sustained period of time. He is now thankfully behind bars.
Reference	The victim of a pensioner jailed for 13 years for horrific sex attacks has told of the abuse that ruined her life.
BART-large	A "manipulative and depraved sexual predator" who raped and sexually abused a woman over a 10-year period has been jailed for 12 years.
BART-large* /OT+AJER	A "depraved" Denbighshire pensioner who raped and sexually abused his victim as a child has been jailed for 12 years.

6.4 Summary

This chapter addresses factual informativeness concerning reference summaries. To encode the model with informative facts of reference summaries more effectively, we have developed an optimal transport-based informative attention method guided by an accumulative joint entropy reduction on named entities.

The experimental results have shown that the models trained with our methods can improve the overall informativeness of the model-generated summaries. Specifically, automatic factuality consistency evaluation has shown that our trained model performs better than the baseline on the dataset of reference summaries characteristically intrinsic, whereas human evaluation of informativeness has shown that the trained models have big margin gains over the baselines on both datasets of intrinsic and extrinsic reference summaries. Further human evaluation of factuality on the entity-related categorized syntactic agreement errors has shown the overall number of factual error reductions over the baseline on the dataset of characteristically extrinsic reference summaries while maintaining the same error level on the dataset of characteristically intrinsic reference summaries. Following the observation from human evaluation of factuality, we have made a qualitative assessment. The results have revealed that our trained model captures more extrinsic but factual entities from the dataset of extrinsic reference summaries than the baseline.

CHAPTER 7

CONCLUSION

This research studies abstractive text summarization (ATS), an important deep-learning task of natural language processing in the Big Data era that requires advanced methods to turn voluminous and often long text data into concise, informative and factual summaries or abstracts for efficient human consumption. The research reported in this thesis has focused on one of the most challenging aspects of ATS, namely factuality. Consequently, this thesis presents four novel ways to improve the factuality of ATS.

Pioneering research over the recent decade has paved the way for further ATS research. In particular, the advancement in Transformer-based language modeling has enabled efficient transfer learning for many downstream tasks, including ATS. Numerous innovative fine-tuning methods have propelled the progress of ATS in terms of fluency, relevance, coherence, and consistency. While such holistic improvements have greatly elevated the faithfulness of ATS, some factual issues or hallucinations remain as challenges to it. It is crucial to address these issues because they interfere with the adaptation of ATS to many real-world applications. Additionally, the advanced solutions yet to be developed to tackle the challenging factual issues will also contribute to other research in natural language processing because pursuing factuality is universally applicable to real-world machine learning applications.

The importance of ATS factuality motivated us to take up the challenge. Through our investigations with existing ATS models, we identified several factual issues that impact the summarization at different levels of granularity and can be improved: word-level arrangements, sentence-level structures, document-level contexts, and sparse relations. We tackled these observed factual issues progressively in the following ways.

First, we identified the undesirable word repeat issue. We have prescribed a determinantal point process-based sampling method coupled with a reinforcement-enhanced learning method to deal with this problem. Our methods directly address the key underlying cause - the distributional mode concentration - by holistically diversifying the word sampling.

Second, after tackling the distorted sub-phrasal hallucinations and the endophoric reference errors observed in our investigation, we presented a syntactic structure-aware semantic similarity learning that engaged with the distorted sub-phrasal hallucinations and a margin ranking learning task to mitigate endophoric reference errors, further enhanced with a structure label classification on generated summaries. By infusing syntactic structure knowledge into model latent space, we complement the prevalent correlational learning with our relational learning to improve the related ATS factuality.

Third, through our investigations and research experiments, we observed additionally more prevalent entity-related hallucinations in model-generated summaries. The problems prompted us to develop even more generic and adaptive solutions to tackle them. Building on promising experimental results from our previous work, we developed an adaptive margin ranking loss that aims at adapting the classic margin ranking loss to variant learning conditions implicit in data. We applied the adaptive loss to facilitate entity alignment learning to address the observed intrinsic entity-related hallucinations, that is, entity-entity hallucinations and entity-reference hallucinations.

Fourth, we addressed an overlooked aspect of factuality in ATS, namely, the factual informativeness concerning reference summaries. This oversight stems from the success of the prevalent cross-attention mechanisms in existing ATS modeling that have achieved model-generated summaries relevant to their source documents. This form of learning relevance is essential, but it can miss out on learning focal information in reference summaries. This inadequacy of learning informative knowledge from reference summaries may deplete models from informative facts in the summary generation. To improve factual informativeness in this regard, we devised an optimal transport-based informative attention method that pivots with an accumulative joint entropy reduction on named entities. Our learning approach reconciles the reference summary perspective with the source document relevance of the existing ATS modeling.

Throughout the research reported on in this thesis, we conducted extensive experiments, evaluations, and analyses to verify and demonstrate the efficacy of our proposed solutions for the factual issues of interest. This research has generated important insights into the plausible reasons underlying the experimental results

and evaluations and shed light on the strengths and weaknesses of various aspects of earlier ATS research, such as ATS modeling and evaluation metrics.

The factuality of ATS is a challenging research field. Many aspects have not been fully explored and exploited. The knowledge we have gained continues to inspire and direct our research pursuits in the field.

Automatic evaluation metrics studied and/or used in this research, such as lexical n-gram overlapping-based statistical metrics like ROUGEs and model-based factuality consistency metrics like SummaC and QuestEval, have shown their strengths and weaknesses in evaluating summaries generated on different datasets depending on reference summaries characteristically being intrinsic or extrinsic. Developing automatic evaluation metrics robust to intrinsic and extrinsic datasets, particularly concerning factuality consistency, is critical to evaluating abstractive text summarization consistently and reliably. Methods proposed in this research have been examined on intrinsic and extrinsic datasets, and have shown capacity in encoding models with intrinsic and/or extrinsic facts from the datasets. One future research will explore the proposed methods for improving metric evaluation models.

Works including this research are constrained to the document length of thousand(s). In the real world, various documents often run into tens of thousands of words if not millions. Adaptation of the factuality research to such documents poses even greater challenges. The recent advancement of large language models (LLMs) has shown great promise in various research fields and real-world applications. Prompt and instruction-engineered learning approaches have demonstrated the impressive factual improvement of LLM-generated text in question-answering applications. It gives rise to the potential for abstractive text summarization of large documents. But, training LLMs for adaptations demands a large amount of data and computational resources. This presents an opportunity in future research to develop efficient and effective modeling methods to adapt LLMs to abstractive text summarization of large documents with facts intact.

References

Griffin Adams, Han-Chin Shing, Qing Sun, Christopher Winestock, Kathleen McKeown, and Noémie Elhadad. 2022. Learning to Revise References for Faithful Summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4009–4027, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Armen Aghajanyan, Akshat Shrivastava, Anchit Gupta, Naman Goyal, Luke Zettlemoyer, and Sonal Gupta. 2020. Better Fine-Tuning by Reducing Representational Collapse.

Laura Aina, Kristina Gulordava, and Gemma Boleda. 2019. Putting Words in Context: LSTM Language Models and Lexical Ambiguity. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3342–3348, Florence, Italy. Association for Computational Linguistics.

Rahul Aralikatte, Shashi Narayan, Joshua Maynez, Sascha Rothe, and Ryan McDonald. 2021. Focus Attention: Promoting Faithfulness and Diversity in Summarization. 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 6078–6095, Online. Association for 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.

Qiwei Bi, Haoyuan Li, Kun Lu, and Hanfang Yang. 2021. Augmented Abstractive Summarization with Document-Level Semantic Graph. *Journal of Data Science*, 19(3):450–464. Publisher: School of Statistics, Renmin University of China.

Alexei Borodin and Eric M. Rains. 2005. Eynard-Mehta Theorem, Schur Process, and their Pfaffian Analogs. *J Stat Phys*, 121(3):291–317.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, pages 1877–1901, Red Hook, NY, USA. Curran Associates Inc.

Jose Camacho-Collados and Mohammad Taher Pilehvar. 2018. From Word To Sense Embeddings: A Survey on Vector Representations of Meaning. *Journal of Artificial Intelligence Research*, 63:743–788.

Meng Cao, Yue Dong, and Jackie Cheung. 2022a. Hallucinated but Factual! Inspecting the Factuality of Hallucinations in Abstractive Summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3340–3354, Dublin, Ireland. Association for Computational Linguistics.

Meng Cao, Yue Dong, Jingyi He, and Jackie Chi Kit Cheung. 2022b. Learning with Rejection for Abstractive Text Summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9768–9780, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6633–6649, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. 2018. Deep Communicating Agents for Abstractive Summarization. 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 1662–1675, New Orleans, Louisiana. Association for Computational Linguistics.

Kyubyung Chae, Jaepill Choi, Yohan Jo, and Taesup Kim. 2024. Mitigating Hallucination in Abstractive Summarization with Domain-Conditional Mutual Information. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1809–1820, Mexico City, Mexico. Association for Computational Linguistics.

Chen Chen, Wei Emma Zhang, Alireza Seyed Shakeri, and Makhmoor Fiza. 2023. The Exploration of Knowledge-Preserving Prompts for Document Summarisation. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. ISSN: 2161-4407.

Sihao Chen, Fan Zhang, Kazoo Sone, and Dan Roth. 2021. Improving Faithfulness in Abstractive Summarization with Contrast Candidate Generation and Selection. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5935–5941, Online. Association for Computational Linguistics.

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.

Sangwoo Cho, Chen Li, Dong Yu, Hassan Foroosh, and Fei Liu. 2019. Multi-document summarization with determinantal point processes and contextualized representations. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 98–103, Hong Kong, China. Association for Computational Linguistics.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What Does BERT Look at? An Analysis of BERT's Attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Gabriele Corso, Luca Cavalleri, Dominique Beaini, Pietro Liò, and Petar Veličković. 2020. Principal Neighbourhood Aggregation for Graph Nets. In *Advances in Neural Information Processing Systems*, volume 33, pages 13260–13271. Curran Associates, Inc.

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.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 1170, pages 13063–13075. Curran Associates Inc., Red Hook, NY, USA.

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. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1067–1082, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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.

Adam Ek and Jean-Philippe Bernardy. 2020. Composing Byte-Pair Encodings for Morphological Sequence Classification. In *Proceedings of the Fourth Workshop on Universal Dependencies (UDW 2020)*, pages 76–86, Barcelona, Spain (Online). Association for Computational Linguistics.

Kawin Ethayarajh. 2019. How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings. 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 55–65, Hong Kong, China. Association for Computational Linguistics.

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.

Patrick Fernandes, Miltiadis Allamanis, and Marc Brockschmidt. 2019. Structured neural summarization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Wikimedia Foundation. Wikimedia downloads.

Xiyan Fu, Jun Wang, Jinghan Zhang, Jinmao Wei, and Zhenglu Yang. 2020. Document Summarization with VHTM: Variational Hierarchical Topic-Aware Mechanism. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7740–7747. Number: 05.

Saadia Gabriel, Antoine Bosselut, Jeff Da, Ari Holtzman, Jan Buys, Kyle Lo, Asli Celikyilmaz, and Yejin Choi. 2021. Discourse Understanding and Factual Consistency in Abstractive Summarization. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 435–447, Online. Association for Computational Linguistics.

Mingqi Gao, Jie Ruan, Renliang Sun, Xunjian Yin, Shiping Yang, and Xiaojun Wan. 2023. Human-like Summarization Evaluation with ChatGPT. ArXiv:2304.02554 [cs] version: 1.

Zhiguang Gao, Feng Jiang, Xiaomin Chu, and Peifeng Li. 2022. Employing Internal and External Knowledge to Factuality-Oriented Abstractive Summarization. In *Natural Language Processing and Chinese Computing*, pages 797–809. Springer, Cham. ISSN: 1611-3349.

Marjan Ghazvininejad, Vladimir Karpukhin, Vera Gor, and Asli Celikyilmaz. 2022. Discourse-Aware Soft Prompting for Text Generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4570–4589, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

José Ángel González, Annie Louis, and Jackie Chi Kit Cheung. 2022. Source-summary Entity Aggregation in Abstractive Summarization. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6019–6034, Gyeongju, Republic of Korea. International Committee on 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.

David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2003. English gigaword. *Linguistic Data Consortium, Philadelphia*, 4(1):34.

Chulaka Gunasekara, Guy Feigenblat, Benjamin Sznajder, Ranit Aharonov, and Sachindra Joshi. 2021. Using Question Answering Rewards to Improve Abstractive Summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 518–526, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Beliz Gunel, Chenguang Zhu, Michael Zeng, and Xuedong Huang. 2020. Mind The Facts: Knowledge-Boosted Coherent Abstractive Text Summarization. *arXiv:2006.15435 [cs]*. ArXiv: 2006.15435.

Michael Hahn. 2020. Theoretical Limitations of Self-Attention in Neural Sequence Models. *Transactions of the Association for Computational Linguistics*, 8:156–171. Place: Cambridge, MA Publisher: MIT Press.

Benjamin Heinzerling and Michael Strube. 2019. Sequence Tagging with Contextual and Non-Contextual Subword Representations: A Multilingual Evaluation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 273–291, Florence, Italy. Association for Computational Linguistics.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The Curious Case of Neural Text Degeneration.

Yuxin Huang, Zhengtao Yu, Junjun Guo, Zhiqiang Yu, and Yantuan Xian. 2020. Legal public opinion news abstractive summarization by incorporating topic information. *International Journal of Machine Learning and Cybernetics*, 11(9):2039–2050.

Fakultit Informatik, Y. Bengio, Paolo Frasconi, and Jfirgen Schmidhuber. 2003. Gradient Flow in Recurrent Nets: the Difficulty of Learning Long-Term Dependencies. *A Field Guide to Dynamical Recurrent Neural Networks*.

Heewon Jang and Wooju Kim. 2021. Reinforced Abstractive Text Summarization With Semantic Added Reward. *IEEE Access*, 9:103804–103810. Conference Name: IEEE Access.

Haozhe Ji, Pei Ke, Zhipeng Hu, Rongsheng Zhang, and Minlie Huang. 2022. Tailoring Language Generation Models under Total Variation Distance.

Ming Jiang, Yifan Zou, Jian Xu, and Min Zhang. 2022. GATSum: Graph-Based Topic-Aware Abstract Text Summarization. *Information Technology and Control*, 51(2):345–355. Number: 2.

Daniel Kang and Tatsunori B. Hashimoto. 2020. Improved Natural Language Generation via Loss Truncation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 718–731, Online. Association for Computational Linguistics.

Shirin Akther Khanam, Fei Liu, and Yi-Ping Phoebe Chen. 2021. Joint knowledge-powered topic level attention for a convolutional text summarization model. *Knowledge-Based Systems*, 228:107273.

Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

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. In *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 555–571, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Diederik P. Kingma and Max Welling. 2014. Auto-Encoding Variational Bayes. ArXiv:1312.6114 [cs, stat] version: 10.

A. C. Koivunen and A. B. Kostinski. 1999. The feasibility of data whitening to improve performance of weather radar. *Journal of Applied Meteorology*, 38(6):741–749.

Keshav Kolluru, Vaibhav Adlakha, Samarth Aggarwal, Mausam, and Soumen Chakrabarti. 2020. OpenIE6: Iterative Grid Labeling and Coordination Analysis for Open Information Extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3748–3761, Online. Association for Computational Linguistics.

Soheil Kolouri, Se Rim Park, Matthew Thorpe, Dejan Slepcev, and Gustavo K. Rohde. 2017. Optimal Mass Transport: Signal processing and machine-learning applications. *IEEE Signal Processing Magazine*, 34(4):43–59. Conference Name: IEEE Signal Processing Magazine.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the Dark Secrets of BERT. 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 4365–4374, Hong Kong, China. Association for Computational Linguistics.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2017. ImageNet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):84–90.

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. 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.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Alex Kulesza and Ben Taskar. 2012a. Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning*, 5(2-3):123–286. ArXiv: 1207.6083.

Alex Kulesza and Ben Taskar. 2012b. Learning determinantal point processes. ArXiv:1202.3738 [cs, stat] version: 1.

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.

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. In *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 439–444, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022b. Deduplicating Training Data Makes Language Models Better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8424–8445, Dublin, Ireland. Association for Computational Linguistics.

Seanie Lee, Dong Bok Lee, and Sung Ju Hwang. 2021. *Contrastive Learning with Adversarial Perturbations for Conditional Text Generation*. International Conference on Learning Representations.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The Power of Scale for Parameter-Efficient Prompt Tuning. In *Proceedings of the 2021 Conference on*

Empirical Methods in Natural Language Processing, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Jicun Li, Xingjian Li, Tianyang Wang, Shi Wang, Yanan Cao, Chengzhong Xu, and Dejing Dou. 2023. Improving Bert Fine-Tuning via Stabilizing Cross-Layer Mutual Information. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Lei Li, Wei Liu, Marina Litvak, Natalia Vanetik, and Zuying Huang. 2019. In conclusion not repetition: Comprehensive abstractive summarization with diversified attention based on determinantal point processes. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 822–832, Hong Kong, China. Association for Computational Linguistics.

Piji Li, Wai Lam, Lidong Bing, and Zihao Wang. 2017. Deep Recurrent Generative Decoder for Abstractive Text Summarization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2091–2100, Copenhagen, Denmark. Association for Computational Linguistics.

Wei Li, Xinyan Xiao, Yajuan Lyu, and Yuanzhuo Wang. 2018. Improving Neural Abstractive Document Summarization with Explicit Information Selection Modeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1787–1796, Brussels, Belgium. Association for Computational Linguistics.

Wei Li and Hai Zhuge. 2021. Abstractive Multi-Document Summarization Based on Semantic Link Network. *IEEE Transactions on Knowledge and Data Engineering*, 33(1):43–54. Conference Name: IEEE Transactions on Knowledge and Data Engineering.

Xiang Lisa Li and Percy Liang. 2021. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online. Association for Computational Linguistics.

Kexin Liao, Logan Lebanoff, and Fei Liu. 2018. Abstract Meaning Representation for Multi-Document Summarization. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1178–1190, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

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

Sen Liu, Libin Yang, and Xiaoyan Cai. 2022a. SEASum: Syntax-Enriched Abstractive Summarization. *Expert Systems with Applications*, 199:116819.

Yixin Liu and Pengfei Liu. 2021. SimCLS: A Simple Framework for Contrastive Learning of Abstractive Summarization. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 1065–1072, Online. Association for Computational Linguistics.

Yixin Liu, Pengfei Liu, Dragomir Radev, and Graham Neubig. 2022b. BRIO: Bringing Order to Abstractive Summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2890–2903, Dublin, Ireland. Association for Computational Linguistics.

Zhenzhen Liu, Chao Wan, Varsha Kishore, Jin Peng Zhou, Minmin Chen, and Kilian Q. Weinberger. 2024. Correction with Backtracking Reduces Hallucination in Summarization. ArXiv:2310.16176 [cs] version: 3.

Yao Lu, Yue Dong, and Laurent Charlin. 2020. Multi-XScience: A Large-scale Dataset for Extreme Multi-document Summarization of Scientific Articles. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8068–8074, Online. Association for Computational Linguistics.

Michal Lukasik, Himanshu Jain, Aditya Menon, Seungyeon Kim, Srinadh Bhojanapalli, Felix Yu, and Sanjiv Kumar. 2020. Semantic Label Smoothing for Sequence to Sequence Problems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4992–4998, Online. Association for Computational Linguistics.

Yuanjie Lyu, Chen Zhu, Tong Xu, Zikai Yin, and Enhong Chen. 2022. Faithful Abstractive Summarization via Fact-aware Consistency-constrained Transformer. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, CIKM '22, pages 1410–1419, New York, NY, USA. Association for Computing Machinery.

Tinghuai Ma, Qian Pan, Huan Rong, Yurong Qian, Yuan Tian, and Najla Al-Nabhan. 2022. T-BERTSum: Topic-Aware Text Summarization Based on BERT. *IEEE Transactions on Computational Social Systems*, 9(3):879–890. Conference Name: IEEE Transactions on Computational Social Systems.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.

Kazuki Matsumaru, Sho Takase, and Naoaki Okazaki. 2020. Improving Truthfulness of Headline Generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1335–1346, Online. Association for Computational Linguistics.

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.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. Scottsdale, Arizona, USA. arXiv. ArXiv:1301.3781 [cs] version: 3.

Feng Nan, Ramesh Nallapati, Zhiguo Wang, Cicero Nogueira dos Santos, Henghui Zhu, Dejiao Zhang, Kathleen McKeown, and Bing Xiang. 2021. Entity-level Factual Consistency of Abstractive Text Summarization. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2727–2733, Online. Association for Computational Linguistics.

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. Place: Cambridge, MA Publisher: MIT Press.

Juri Opitz, Angel Daza, and Anette Frank. 2021. Weisfeiler-Leman in the Bamboo: Novel AMR Graph Metrics and a Benchmark for AMR Graph Similarity. *Transactions of the Association for Computational Linguistics*, 9:1425–1441.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Laura Perez-Beltrachini and Mirella Lapata. 2021. Multi-document summarization with determinantal point process attention. *Journal of Artificial Intelligence Research*, 71:371–399.

Diogo Pernes, Afonso Mendes, and André F. T. Martins. 2022. Improving abstractive summarization with energy-based re-ranking. In *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 1–17, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Yury Polyanskiy and Yihong Wu. 2022. *Information Theory: From Coding to Learning*, first edition. Cambridge University Press.

Peng Qi, Timothy Dozat, Yuhao Zhang, and Christopher D. Manning. 2018. Universal Dependency Parsing from Scratch. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 160–170, Brussels, Belgium. Association for Computational Linguistics.

Weizhen Qi, Yu Yan, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou. 2020. ProphetNet: Predicting Future N-gram for Sequence-to-SequencePre-training. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2401–2410, Online. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving Language Understanding by Generative Pre-Training. *OpenAI*, page 12.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language Models are Unsupervised Multitask Learners. *OpenAI*, 1:9.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2016. Sequence Level Training with Recurrent Neural Networks. ArXiv:1511.06732 [cs] version: 7.

Mathieu Ravaut, Hailin Chen, Ruochen Zhao, Chengwei Qin, Shafiq Joty, and Nancy Chen. 2023. PromptSum: Parameter-Efficient Controllable Abstractive Summarization.

Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-Critical Sequence Training for Image Captioning. pages 7008–7024.

Paul Roit, Johan Ferret, Lior Shani, Roee Aharoni, Geoffrey Cideron, Robert Dadashi, Matthieu Geist, Sertan Girgin, Leonard Hussenot, Orgad Keller, Nikola Momchev, Sabela Ramos Garea, Piotr Stanczyk, Nino Vieillard, Olivier Bachem, Gal Elidan, Avinatan Hassidim, Olivier Pietquin, and Idan Szpektor. 2023. Factually Consistent Summarization via Reinforcement Learning with Textual Entailment Feedback. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6252–6272, Toronto, Canada. Association for Computational Linguistics.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. FaceNet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823. ISSN: 1063-6919.

M. Schuster and K.K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681. Conference Name: IEEE Transactions on Signal Processing.

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, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Kaiqiang Song, Logan Lebanoff, Qipeng Guo, Xipeng Qiu, Xiangyang Xue, Chen Li, Dong Yu, and Fei Liu. 2020. Joint Parsing and Generation for Abstractive Summarization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8894–8901.

Kaiqiang Song, Lin Zhao, and Fei Liu. 2018. Structure-Infused Copy Mechanisms for Abstractive Summarization. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1717–1729, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Arvind Krishna Sridhar and Erik Visser. 2022. Improved Beam Search for Hallucination Mitigation in Abstractive Summarization. ArXiv:2212.02712 [cs] version: 1.

Rhea Sukthanker, Soujanya Poria, Erik Cambria, and Ramkumar Thirunavukarasu. 2020. Anaphora and coreference resolution: A review. *Information Fusion*, 59:139–162.

Shichao Sun and Wenjie Li. 2021. Alleviating Exposure Bias via Contrastive Learning for Abstractive Text Summarization. ArXiv:2108.11846 [cs] version: 1.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Liyan Tang, Tanya Goyal, Alex Fabbri, Philippe Laban, Jiacheng Xu, Semih Yavuz, Wojciech Kryscinski, Justin Rousseau, and Greg Durrett. 2023a. 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, Toronto, Canada. Association for Computational Linguistics.

Xiangru Tang, Arman Cohan, and Mark Gerstein. 2023b. Aligning Factual Consistency for Clinical Studies Summarization through Reinforcement Learning. In *Proceedings of the 5th Clinical Natural Language Processing Workshop*, pages 48–58, Toronto, Canada. Association for Computational Linguistics.

Jency Thomas, Amrutha Sreeraj, Ayswarya Sreeraj, Megha Mary Varghese, and Thomas Kuriakose. 2022. Automatic text summarization using deep learning and reinforcement learning. In *Sentimental Analysis and Deep Learning*, Advances in Intelligent Systems and Computing, pages 769–778, Singapore. Springer.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Eric Gaussier, and Guillaume Bouchard. 2016. Complex Embeddings for Simple Link Prediction. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 2071–2080. PMLR. ISSN: 1938-7228.

Akim Tsvigun, Ivan Lysenko, Danila Sedashov, Ivan Lazichny, Eldar Damirov, Vladimir Karlov, Artemy Belousov, Leonid Sanochkin, Maxim Panov, Alexander Panchenko, Mikhail Burtsev, and Artem Shelmanov. 2022. Active Learning for Abstractive Text Summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5128–5152, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Liam van der Poel, Ryan Cotterell, and Clara Meister. 2022. Mutual Information Alleviates Hallucinations in Abstractive Summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5956–5965, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing Multi-Head Self-Attention: Specialized Heads Do the Heavy Lifting, the Rest Can Be Pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5797–5808, Florence, Italy. Association for Computational Linguistics.

David Wan and Mohit Bansal. 2022. FactPEGASUS: Factuality-Aware Pre-training and Fine-tuning for Abstractive Summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1010–1028, Seattle, United States. Association for Computational Linguistics.

Tianshu Wang, Faisal Ladhak, Esin Durmus, and He He. 2022. Improving Faithfulness by Augmenting Negative Summaries from Fake Documents. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11913–11921, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Zhengjue Wang, Zhibin Duan, Hao Zhang, Chaojie Wang, Long Tian, Bo Chen, and Mingyuan Zhou. 2020. Friendly Topic Assistant for Transformer Based Abstractive Summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 485–497, Online. Association for Computational Linguistics.

Ronald J. Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3):229–256.

Ronald J. Williams and David Zipser. 1989. A Learning Algorithm for Continually Running Fully Recurrent Neural Networks. *Neural Computation*, 1(2):270–280.

Wenhao Wu, Wei Li, Jiachen Liu, Xinyan Xiao, Ziqiang Cao, Sujian Li, and Hua Wu. 2022. FRSUM: Towards Faithful Abstractive Summarization via Enhancing Factual Robustness. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3640–3654, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Wenhao Wu, Wei Li, Xinyan Xiao, Jiachen Liu, Ziqiang Cao, Sujian Li, Hua Wu, and Haifeng Wang. 2021. BASS: Boosting Abstractive Summarization with Unified Semantic Graph. 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 6052–6067, Online. Association for Computational Linguistics.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. ArXiv:1609.08144 [cs] version: 2.

Yu Xia, Xu Liu, Tong Yu, Sungchul Kim, Ryan Rossi, Anup Rao, Tung Mai, and Shuai Li. 2024. Hallucination Diversity-Aware Active Learning for Text Summarization. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8665–8677, Mexico City, Mexico. Association for Computational Linguistics.

Liqiang Xiao, Hao He, and Yaohui Jin. 2022. FusionSum: Abstractive summarization with sentence fusion and cooperative reinforcement learning. *Knowledge-Based Systems*, 243:108483.

Wen Xiao and Giuseppe Carenini. 2023. Entity-based SpanCopy for Abstractive Summarization to Improve the Factual Consistency. In *Proceedings of the 4th Workshop on Computational Approaches to Discourse (CODI 2023)*, pages 70–81, Toronto, Canada. Association for Computational Linguistics.

Jiacheng Xu, Shrey Desai, and Greg Durrett. 2020. Understanding Neural Abstractive Summarization Models via Uncertainty. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6275–6281, Online. Association for Computational Linguistics.

Jiaxuan You, Rex Ying, and Jure Leskovec. 2021. Design Space for Graph Neural Networks. *arXiv:2011.08843 [cs]*. ArXiv: 2011.08843.

Jiaxuan You, Zhitao Ying, and Jure Leskovec. 2020. Design Space for Graph Neural Networks. In *Advances in Neural Information Processing Systems*, volume 33, pages 17009–17021. Curran Associates, Inc.

Haopeng Zhang, Semih Yavuz, Wojciech Kryscinski, Kazuma Hashimoto, and Yingbo Zhou. 2022a. Improving the Faithfulness of Abstractive Summarization via Entity Coverage Control. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 528–535, Seattle, United States. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020a. PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. In *Proceedings of the 37th International Conference on Machine Learning*, pages 11328–11339. PMLR. ISSN: 2640-3498.

Shengqiang Zhang, Xingxing Zhang, Hangbo Bao, and Furu Wei. 2022b. Attention Temperature Matters in Abstractive Summarization Distillation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 127–141, Dublin, Ireland. Association for Computational Linguistics.

Yuhao Zhang, Derek Merck, Emily Tsai, Christopher D. Manning, and Curtis Langlotz. 2020b. Optimizing the Factual Correctness of a Summary: A Study of Summarizing Radiology Reports. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5108–5120, Online. Association for Computational Linguistics.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced Language Representation with Informative Entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

Zheng Zhao, Shay B. Cohen, and Bonnie Webber. 2020. Reducing Quantity Hallucinations in Abstractive Summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2237–2249, Online. Association for Computational Linguistics.

Changmeng Zheng, Yi Cai, Guanjie Zhang, and Qing Li. 2020a. Controllable Abstractive Sentence Summarization with Guiding Entities. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5668–5678, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Chujie Zheng, Kunpeng Zhang, Harry Jiannan Wang, Ling Fan, and Zhe Wang. 2021. Topic-Guided Abstractive Text Summarization: a Joint Learning Approach. ArXiv:2010.10323 [cs] version: 2.

Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren. 2020b. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(07):12993–13000. Number: 07.

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2020. Extractive Summarization as Text Matching. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6197–6208, Online. Association for Computational Linguistics.

Ming Zhong, Pengfei Liu, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2019. Searching for Effective Neural Extractive Summarization: What Works and What's Next. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1049–1058, Florence, Italy. Association for Computational Linguistics.

Chunting Zhou, Graham Neubig, Jiatao Gu, Mona Diab, Francisco Guzmán, Luke Zettlemoyer, and Marjan Ghazvininejad. 2021a. Detecting Hallucinated Content in Conditional Neural Sequence Generation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1393–1404, Online. Association for Computational Linguistics.

Yinghua Zhou, Fang Cao, Yangyang Cao, Ning Yang, and Zhen Li. 2021b. A Grammar-Aware Pointer Network for Abstractive Summarization. In *Modern Industrial IoT, Big Data and Supply Chain*, Smart Innovation, Systems and Technologies, pages 207–216, Singapore. Springer.

APPENDIX A

APPENDIX FOR CHAPTER 3

A.1 Encoder-Decoder

Encoder-decoders are formulated for correlational learning of text sequences.

Conditional Probability Encoder-decoders share autoregressive formulation as a conditional probability estimation problem:

$$p(\tilde{y}_t|\theta) = p(\tilde{y}_t|\tilde{y}_1,...,\tilde{y}_{t-1},\mathbf{x};\theta) \tag{A.1}$$

where a parameterized model θ generates a word/token $\tilde{y}_t$ at time step t probabilistically conditional on the past generated sequence $\langle\tilde{y}_1,...,\tilde{y}_{t-1}\rangle$ and the observation input sequence $\mathbf{x}$.

Cross-Attention An important construct of encoder-decoders is cross-attention, which learns word correlations between the target (gold reference) sequence and source sequence in the latent space. In a simple form without loss of generality, a cross-attention can be formulated of three equations as follows:

$$\mathbf{h}_{ij} = f(\mathbf{h}_{y_i},\mathbf{h}_{x_j}), \quad \mathbf{h}_{y_i},\mathbf{h}_{x_j} \in \mathbb{R}^d \tag{A.2}$$

where it first correlates pairwise the i^{th} target word latent state $\mathbf{h}_{y_i}$ and the j^{th} source word latent state $\mathbf{h}_{x_j}$, f is a pairwise correlation operation, a common choice is either dot product or addition. It then computes the attention (correlational probability) of $\mathbf{h}_{ij}$ over the all source correlations for the target $\mathbf{h}_{y_i}$, usually by Softmax function, defined as:

$$a_{ij} = \frac{e^{\mathbf{h}_{ij}}}{\sum_{j'=1}^{|s|} e^{\mathbf{h}_{ij'}}} \tag{A.3}$$

where $|s|$ is the source sequence length. It finally updates the target state $\mathbf{h}_{y_i}$ as a sum of weighted correlation states as:

$$\mathbf{h}_{y_i} = \sum_{j=1}^{|s|} a_{ij}\mathbf{h}_{ij}. \tag{A.4}$$

Generative Sampling To make predictions, ATS commonly uses sampling on maximum likelihood estimation (MLE), formulated with the latent states outputted from the last layer of the decoder. This can also be simplified into three steps. It first linearly transforms a latent state to logits (unnormalized distribution over vocabulary), in a simple form as:

$$g_i^{|V|} = W \cdot \mathbf{h}_{y_i} + \mathbf{b}, \quad W \in \mathbb{R}^{|V| \times d}, \mathbf{b} \in \mathbb{R}^{|V|} \tag{A.5}$$

$|V|$ is the vocabulary size, $\{W, \mathbf{b}\}$ are the linear transformation parameters, and $g_i^{|V|}$ is the vector of logits. The logits are then normalized into a probability distribution commonly by Softmax function as:

$$p_i^k = \frac{e^{g_i^k}}{\sum_{k'=1}^{|V|} e^{g_i^{k'}}} \tag{A.6}$$

where p_i^k is the probability of the k^{th} word of the vocabulary V. Finally, sampling on the MLE finds the word (by its vocabulary index) of the highest probability:

$$\begin{aligned} k^\star &= \underset{k}{\operatorname{argmax}}(\{p_i^k | k \in |V|\}) \\ &\Rightarrow \tilde{y}_i = V[k^\star] \end{aligned} \tag{A.7}$$

where the prediction $\tilde{y}_i$ is selected by indexing the vocabulary V using $k^\star$.

Learning Objective The standard learning objective of MLE uses the cross-entropy (CE) loss function:

$$\mathscr{L}_{\text{MLE}} = -\frac{1}{|g|} \sum_{i=1}^{|g|} y_i \log \tilde{y}_i \tag{A.8}$$

where $\tilde{y}_i$ is the i^{th} word prediction, and y_i is the corresponding ground truth. In practice, Equation (A.8) is relaxed to the estimated conditional probability distribution in the form of Equation (A.1). With the commonly used teacher-forcing learning strategy (Williams and Zipser, 1989) for training, the learning objective can be expressed as:

$$\mathscr{L}_{\text{MLE}} = -\frac{1}{|g|} \sum_{i=1}^{|g|} y_i \log p(\tilde{y}_i | y_{1:i-1}, \mathbf{x}) \tag{A.9}$$

where the ground truth or gold reference sequence $y_{1:i-1}$ replaces the corresponding past generated sequence $\tilde{y}_{1:i-1}$ otherwise.

APPENDIX B

APPENDIX FOR CHAPTER 4

B.1 Graph Neural Networks

To help grasp GNN essentials and the reason behind GNN model choice in our work, this section presents the generic working mechanism and important properties of GNNs illustrated by a simple GNN graph of four nodes, shown in Figure B.1.

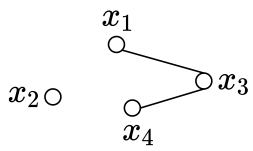


Figure B.1: A simple graph example.

For simplicity without loss of generality, a node is represented as a scalar value. Thus, the nodes form a vector:

$$\boldsymbol{x}^0 = [x_1, x_2, x_3, x_4]^{\mathrm{T}} \tag{B.1}$$

where $\boldsymbol{x}^0$ represents the initial states. Furthermore, an adjacent node connection matrix A is given as follows:

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}. \tag{B.2}$$

For simplicity, the adjacent matrix represents a single relational and undirected graph. A two-layer GNN computes the graph node representations as follows. Also,

for simplicity, the learnable parameters are omitted, and only linear operations are considered for the illustration. At the first layer, the latent states are computed as:

$$\boldsymbol{x}^1 = A \cdot \boldsymbol{x}^0 = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_1 + x_3 \\ x_2 \\ x_1 + x_3 + x_4 \\ x_3 + x_4 \end{bmatrix}. \tag{B.3}$$

Following the second layer of the GNN, the latent states are further updated as:

$$\boldsymbol{x}^2 = A \cdot \boldsymbol{x}^1 = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 + x_3 \\ x_2 \\ x_1 + x_3 + x_4 \\ x_3 + x_4 \end{bmatrix} = \begin{bmatrix} 2x_1 + 2x_3 + x_4 \\ x_2 \\ 2x_1 + 3x_3 + 2x_4 \\ x_1 + 2x_3 + 2x_4 \end{bmatrix}. \tag{B.4}$$

It is noticed that

- When the state updates progress to higher layers, the GNN aggregates every node with the indirectly connected nodes further away. It may thus result in an over-smoothing phenomenon. That is, all nodes have similar latent states numerically insignificant to distinguish for any learning tasks (e.g. classification).
- Without changing connectivity, the mere change of geometric positioning does not alter the computed result.
- Nodes have their feature information multiplied by their degrees (the number of connections to the other nodes). Thus, nodes with a large degree number may overwhelm the other node features.

Provided the observation above, it is clear that any capable GNN should possess the properties of

- Discriminative (to over-smoothing). In other words, the power of representative expressiveness.
- Geometric invariant (to node position changes).
- Scaling balance (to degree imbalance).

Meanwhile, the illustrated GNN operations are generic to a wide range of GNN modeling tasks (e.g., You et al., 2021), and can be summarised as

1. Message passing. A node receives the information (a.k.a. messages) from its connected neighboring nodes.

2. Message aggregation. The received messages are merged into a feature representation by some algorithm.

3. Node state update. The receiving node updates its state with the aggregated message feature by some algorithm.

B.2 Super Token Representation Learning

Given the tokens of a word, we construct a token graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{D})$. $\mathcal{V} = \{v_0, ..., v_m\}$ is the token node set. $\mathcal{E} = \{(v_0, v_j) | j \in [0, ..., m]; \langle v_0, ..., v_m \rangle\}$ is the edge set between the leading token v_0 and each following tokens. $\mathcal{D} = \{0, ..., m\}$ is the distance set of the subsequent tokens to the leading token. Note that we add a self-loop edge to the leading token to deal with a single-token word case. Similar to the idea of positional embeddings in Transformer-based models, we build the distance embeddings of vocabulary size $|\mathcal{D}|$ but use them as edge features.

We adopt a generic GNN model proposed by You et al. (2020) for learning the representations. Given the designed graph, a single-layer GNN is sufficient to learn super token representations. In a simple form, the message passing is defined as:

$$\mathbf{h}_u'^{\mathcal{V}} = (W_g^{\mathcal{V}} \mathbf{h}_u^{\mathcal{V}} + \mathbf{b}_g^{\mathcal{V}}) + (W_g^{\mathcal{D}} \mathbf{h}_u^{\mathcal{D}} + \mathbf{b}_g^{\mathcal{D}}), \quad u \in \mathcal{N}(v_0) \tag{B.5}$$

where $\{W_g^{\mathcal{V}}, \mathbf{b}_g^{\mathcal{V}}\}$ and $\{W_g^{\mathcal{D}}, \mathbf{b}_g^{\mathcal{D}}\}$ are the linear transformation parameters of the token embedding $\mathbf{h}_u^{\mathcal{V}}$ and the distance embedding $\mathbf{h}_u^{\mathcal{D}}$ respectively. The message aggregation is followed as:

$$\mathbf{h}_{\{u\}}^{\mathcal{V}} = \sum_{u \in \mathcal{N}(v_0)} \mathbf{h}_u'^{\mathcal{V}}. \tag{B.6}$$

The update is then computed as:

$$\mathbf{h}_{v_0}'^{\mathcal{V}} = \mathbf{h}_{v_0}^{\mathcal{V}} + \mathbf{h}_{\{u\}}^{\mathcal{V}}. \tag{B.7}$$

During training, we apply the GNN to aggregate the tokens of each word to a super token node at the leading token position of each word. The word-level structures are applied to the super tokens thereafter.

The GNN model is adopted from PyTorch Geometric package[1] with our configurations.

B.3 Human Evaluation

B.3.1 Human Evaluation Criteria and Guidelines

There are a range of human evaluation criteria seen in prior works. To relate to our work, we choose faithfulness and fluency. But fluency has a close link to coherence

[1] `https://pytorch-geometric.readthedocs.io`.

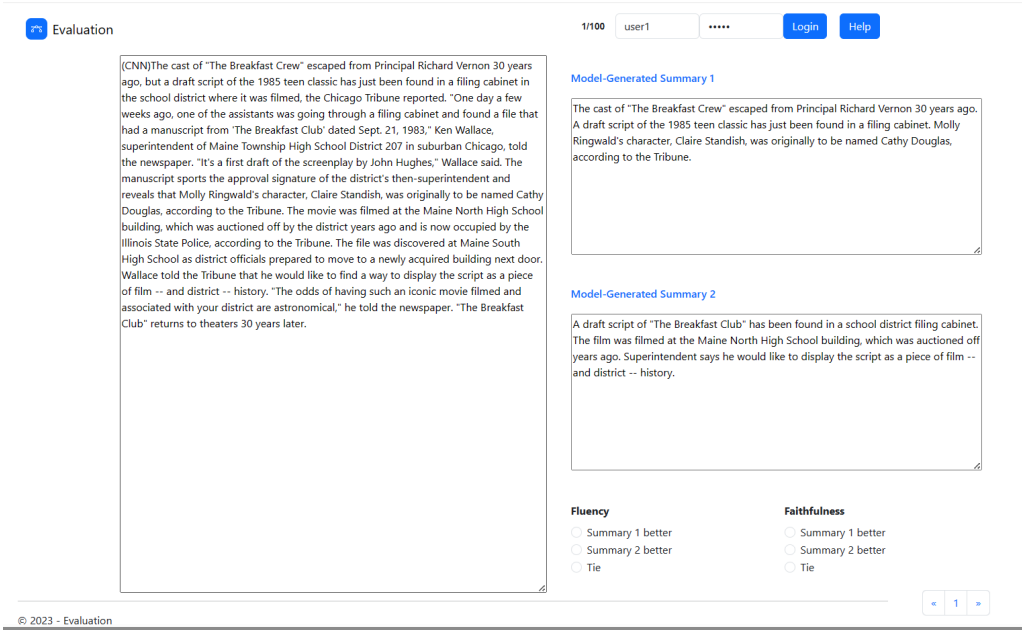


Figure B.2: An illustration of human evaluation user interface.

while faithfulness is inevitably associated with relevance. So, we define the following guidelines for the evaluation:

1. The grammatical or factual errors propagated from articles are discounted in deciding the respective choices.
2. If information in a summary is not evidenced in the article but can be reasonably deduced from the article, they are considered factual, not an error.
3. If both summaries of an article have no apparent factual errors, key information summarized from articles may be considered in deciding faithfulness.
4. If both summaries of an article contain factual errors, errors in key information have more significance in deciding faithfulness.
5. Missing punctuation marks may be less concern if they do not obscure comprehension or alter the facts from articles.
6. If unsure of which summary is better, 'tie' is the option to choose.
7. An annotator (evaluator) has the freedom to decide what are the key information that an article intends to convey.

B.3.2 Human Evaluation User Interface

Figure B.2 illustrates the main user interface of the human evaluation tool. The model-generated summaries shown in "Summary 1" and "Summary 2" are randomly

shuffled such that the numbers “1” and “2” in the captions have no fixed binding to the models throughout the evaluation.

APPENDIX C

APPENDIX FOR CHAPTER 5

C.1 M3 Convolutional Neural Network

Convolutional neural networks (CNNs) have long been used in deep learning research. The CNNs with multi-filters and multi-kernels are computationally more economical than Transformer-based models for relatively short sequences while still capturing salient features well. They have been used to encode a representation of short text sequences (e.g., Kim, 2014; Chen and Bansal, 2018) with a single-scale max-over-time pooling. We adopt a CNN for encoding a representation of multi-sentential entity context. We think the choice of CNN is sensible because the inputs to the CNN are the information-infused latent states from the BART encoder-decoder. The lengths of our multi-sentential contexts are somewhat in between full-length documents and the short-text sequences seen in the aforementioned prior works. So, we extend single-scale pooling to multi-scale pooling by borrowing the idea from deep learning in computer vision. Given a padded text sequence x to the pre-defined max length and a down-sampling scale factor, we can compute the number of down-sampling scales. We can also compute input and output size along the sequence length dimension at each scale. So, each convolutional block consists of a 1D convolution, a ReLU, and an adaptive 1D max-pool. A fully connected linear layer is applied to the output of the last down-sampling scale.

C.2 Defining Rules for Entity-Related Hallucination Analysis

Named entity-related hallucinations (NERHs) can have various forms. Therefore, it is necessary to define rules to identify those hallucinations for our statistical analysis. Although the focus of this paper is the intrinsic NERHs, the identifying

rules also cover the extrinsic NERHs. So, we categorize our assessment rules for a NERH as follows,

1. It is an intrinsic NERH if a named entity is mistaken for the other named entity within the context of the source document.

2. It is an intrinsic NERH if a named entity is mistaken concerning the context of the source document where there may not be any other named entity present in the source document.

3. A named entity has its name incorrect, for example, surname and given name. If the full name is wrong, we categorize it as an extrinsic NERH. If the partial name (either surname or given name) is wrong, we categorize it as an intrinsic NERH. However, if there are one or two wrong characters in a partial name, we categorize it as a misspelling instead.

4. It is an intrinsic NERH if a named entity's reference is mistaken for another entity's reference, for example, mistaking 'she' for 'he'.

5. It is an intrinsic NERH if a named entity is missing altogether when it is necessary in its place. For example, no named entity is present when a reference (e.g., he or she) is mentioned.

We do not mark hallucinations for entity aggregations if their logical containment relation with the aggregated named entity is correct given their context.

As the pre-trained BART model has encoded a large amount of prior knowledge of general information, there are considerable amounts of extrinsic hallucinations that may be factual. Thus, we do not consider the cases as hallucinations or errors if we can verify them using external knowledge bases such as Wikipedia, online news, and/or Google Maps even though they cannot be deduced from the source document. We do not mark cases as errors if they can be reasonably deduced from the source documents. We extend the negation category to cover different forms of meaning contradiction at the phrasal level.

APPENDIX D

APPENDIX FOR CHAPTER 6

D.1 Human Evaluation of Summary Informativeness

D.1.1 Human Evaluation Guidelines

We define the following guidelines for the evaluation:

1. Evaluation mainly considers the coverage of focal information (explication rather than word-by-word match).

2. Reference summaries are unnecessarily ideal. Sometimes, they could be uninformative or erroneous. So, evaluation should be based on reference summaries consolidated with the key information the articles convey instead of relying on reference summaries alone.

3. Generated summaries may contain factual errors that are inconsistent with or contradict the facts in articles. Some errors may obscure informativeness more than others. An annotator is free to decide on that.

4. Generated summaries may contain extrinsic knowledge that is not evidenced in the articles and reference summaries. Such extrinsic knowledge may or may not be factual. An annotator can ignore them.

5. Information in a model-generated summary may not be evidenced directly in the article sometimes, but it may be deduced, induced, or derivable from the article.

6. Long summaries may be truncated by modeling standards. So, evaluation assesses summaries up to the point.

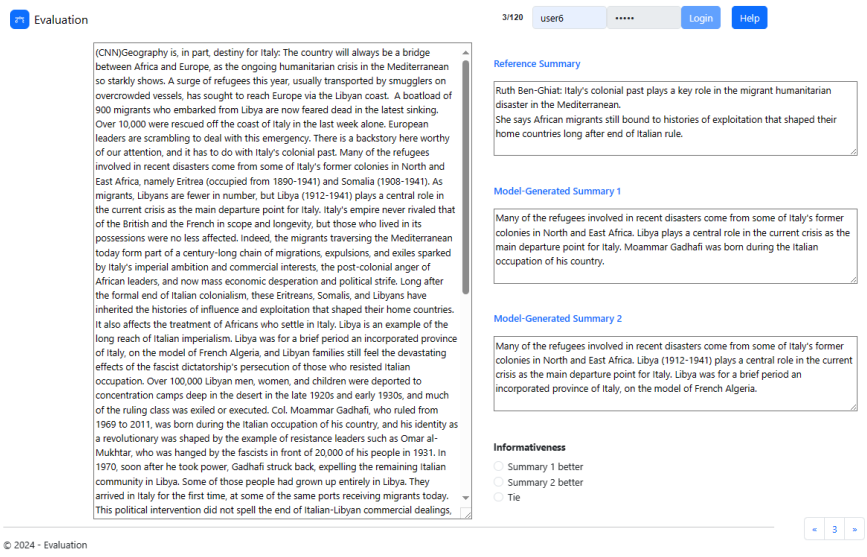


Figure D.1: Main user interface of informativeness evaluation.

7. Summaries may not be in grammatically perfect format, such as whitespace, punctuation, indefinite article, and plurality. They are less concerned if they do not alter the meaning.

8. If unsure of which model-generated summary is better, please choose 'tie'.

9. External resources, such as dictionaries, Google Search and Maps, can be used to assist the evaluation.

D.1.2 Normalize Whitespace in Model-Generated Summaries for Informativeness Evaluation

As discussed in Section 6.3.1, our NER-annotated training dataset is built from Penn Treebank segmented words. We have observed that the Penn Treebank segmented words fit into the pre-trained BART model tokenization vocabulary well. However, the model tokenizer has an encoding strategy that whitespace proceeding a word is converted into a special character and merged with the word before applying byte-pair encoding to generate the token sequences. As a result, the model tokenizer's decoding process in inference restores both whitespace and word. This leaves extra whitespaces in our trained model-generated summaries, such as around hyphens and inside round brackets. These extra whitespaces do not alter comprehension but may lead to discerning summary-model binding due to such spacing patterns during informativeness evaluation and may thus affect the evaluation fairness. Therefore, we 'normalize' (remove) whitespaces in the samples used in the evaluation. Additionally, we have also observed that whitespace after an apostrophe may occasionally be absent in both our model-generated summaries and summaries from the baseline.

For a similar reason, we also ‘normalize’ (insert) whitespace after an apostrophe when it is proper. We do so for the sample summaries of both ours and the baseline.

D.1.3 Human Evaluation User Interface

Figure D.1 shows the main user interface of the informativeness evaluation tool, where articles and summaries are populated for evaluation. Note that the model-generated summaries shown in “Summary 1" and “Summary 2" are randomly shuffled such that the numbers “1" and “2" in the captions have no fixed binding to the models throughout the evaluation.