Exploring the Limits of Knowledge-Graph Augmented Abstractive Summarization for Improving Factuality

Anton Abilov
Department of Computer Science
Cornell Tech, Cornell University
aa2776@cornell.edu

Lucas Davis
Department of Computer Science
Cornell Tech, Cornell University
lld58@cornell.edu

Lars Kouwenhoven
Department of Computer Science
Cornell Tech, Cornell University
lk545@cornell.edu

Ze Yuan Li
Department of Computer Science
Cornell Tech, Cornell University
zl344@cornell.edu

Aman Prasad
Department of Computer Science
Cornell Tech, Cornell University
ap798@cornell.edu

Abstract

Factual summarization remains an open problem. While state of the art models such as PEGASUS generate fluent summaries, they make frequent mistakes in terms of factuality. In this project, we review the capability of knowledge-graph augmented summarization models to alleviate this issue. We reproduce the results by ASGARD, evaluate the summaries in terms of factuality and fluency on the CNN/Daily Mail dataset, and conduct a thorough error analysis inspired by the FRANK benchmark. We find that the knowledge-graph augmented summarization model generates more factual summaries, but they are less fluent.

1 Introduction

Recent breakthroughs in deep learning research have significantly improved our ability to automatically summarize text [21]. However, despite transformer models' ability to excel at abstractive summarization in terms of fluency, they are prone to generate text that is not factual and faithful to the source document [17]. An example is shown in Table 1.

Source document
The former chairman of Bradford City was linked to eight other fires before the Valley Parade blaze that killed 56, a new book has claimed. Author Martin Fletcher claims the devastating fire was not an accident and has revealed a sequence of other blazes at businesses owned by or associated with Stafford Heginbotham [...]
Generated summary
Author Martin Fletcher claims the devastating fire was not started deliberately and was caused by a discarded cigarette. [...]

Table 1: A summary generated by a transformer model which completely misrepresents the facts in the source document.

Preprint. Under review.

As pre-trained language models increase their capability and become more widely available, it is crucial to solve the problem of factuality to make the models more applicable in a wider set of use-cases that could have great societal impacts, such as medical note or legal document summarization.

In this work, we seek to explore the problem of factual document summarization by studying the methods proposed by Huang et al. [9] which leverages a knowledge-graph to generate more factual summaries. They incorporate a graph attention network which attend to facts extracted from the source document using OpenIE knowledge extraction. Their results indicate that attending to extracted facts leads to marginal improvements in factuality metrics, but the authors do not thoroughly examine the factuality errors that remain. We seek to better understand their method and its limits in terms of generating factual summaries through reproducing their results and conducting a thorough error analysis. [1] We hypothesize that attending to a graph representation of the document can reduce the model's propensity to hallucinate by leveraging the structured representation of the source document and increasing the saliency of key parts of the source document.

Our **main contribution** in this paper is a qualitative analysis of whether incorporating a knowledge graph improves the factuality of generated summaries. We evaluate this by comparing the graph-augmented summarization model with a transformer baseline. We compute two automated factuality metrics - FactCC [10] and FEQA [5], and conduct a manual annotation. Using our manually annotated summaries, we explore whether automated factuality metrics (FEQA and FactCC) accurately capture whether a summary is factual. Finally, we closely examine the errors made by the graph model by assessing a smaller set of generated summaries to inspire directions for future work.

2 Previous Works

Previous years have seen an increased pervasiveness of language model pretraining for a wide variety of downstream tasks. Examples of the development of such models include GPT [18], XLM [11] and BERT [4]. One application of interest is text summarization, as explored by Liu and Lapata [13]. Specifically, they show how BERT can be used to generative summaries based on abstractive and extractive approaches. In both settings, the models they develop achieve state-of-the-art results.

Maynez et al. [15] then explored the performance of such summarization tasks, both in terms of faithfulness (does the content in the summary truly occur in the original document) and factuality (are the claims in the summary accurate and truthful?). They break down factuality errors into two categories - extrinsic and intrinsic hallucinations. Extrinsic hallucinations are parts of the generated summary that are not supported by the document, which might be introduced by the model due to associations it learns during pretraining. Intrinsic hallucinations are reasoning errors which are caused by the model misrepresenting knowledge entailed within the source text. The authors uncover that abstractive summarization models frequently perform poorly in terms of factuality and faithfulness, and show that traditional metrics such as ROUGE score are inadequate for measuring these types of error. Pagnoni et al. [17] perform a further study of factuality errors and develop a typology which we leverage in this work.

Many recent works have attempted to address factuality errors of summarization models through either post-correction [3], constrained generation [14] or framing new learning objectives [16], yet factual summary generation remains an open problem. One potential method to improve factuality and faithfulness of summarization is to attend to extracted facts from the source document using graph neural networks. Huang et al. [9] propose ASGARD, which gives a summarizing semantic interpretation over the input to generate more informative summaries, and conclude their method improves the faithfulness of abstractive summary generation. A further effort by Zhu et al. [23] utilizes graph attention to generate factual summaries. Additionally, their method is able to improve the factuality of existent summaries. Finally, Gunel et al. [8] relies on the Wikidata knowledge graph, incorporated into a encoder-decoder architecture to render their abstractive summary model more fact-aware. Their results also show an improvement compared to baseline transformers.

Inconsistency between different factuality metrics remains a problem that inhibits further research in this field. Additionally, it is known that automatic metrics do not always correspond to human evaluations. Ferreira et al. [7] posit an early exploration of consistency between automated and

[1] Our code is available in two repositories on Github: `https://github.com/anton164/graph-factuality`, `https://github.com/anton164/GraphAugmentedSum`

manual scoring metrics for text summarization. Instead of having human subjects rate the generated summaries, they instead ask the subjects to highlight sentences from an article that should be included in a summary. They subsequently analyze the overlap between the machine-generated and human-highlighted summaries and find that summarization methods leading to high ROUGE scores generally have a high overlap with the human-highlighted summaries. Fabbri et al. [6] perform a comparison, employing both experts and crowd-sourced annotations, and find BertScore and ROUGE-3 to be most strongly correlated with factuality, and low correlation with ROUGE-L.

3 Methods

3.1 Dataset

We used the CNN/Daily Mail dataset (CNN/DM) [19].

Dataset	Training	Validation	Testing	Year of Data
CNN/Daily Mail	281,113	13,368	11,490	2007 - 2015

Table 2: Dataset Statistics

3.2 Open IE Extraction

Open information extraction (Open IE) refers to the process of extracting relation tuples from sentences. The implementation used in ASGARD and this report was from Stanford Natural Language Processing Group [1]. A relation is a tuple of a subject, verb, and direct object. Each piece of the tuple may contain extra descriptive terms such as adjectives and prepositions. For example "Lewis Hamilton suggested Mercedes will be back in control for this weekend's Chinese Grand Prix." has the relation (Lewis Hamilton, suggested, Mercedes) extracted. In a summarization model, these relation tuples help separate the facts in the reference text avoiding situations where the model combines multiple thoughts into an incorrect statement. In practice the relation tuples help guide models by being the input to attention layers.

3.3 Models

Baseline

In order to evaluate the performance of the graph-augmented summarization model, we use a pre-trained RoBERTa transformer as our baseline summary generator. This is effectively the graph model without a graph.

ASGARD

We replicate the *ASGARD: Abstractive Summarization with Graph Augmentation and Reward* [9] model. This framework uses and builds up on a typical encoder-decoder framework: the encoder that will encode the source document, and a decoder that will generate the abstractive summary for the source document. Along with this, the framework utilizes an open information extraction system (OpenIE) in order to generate a global document information graph. The global document information graph is used in conjunction with the outputs of a bidirectional-LSTM in the document encoder in order to generate the initial embedding for a Graph Attention Network (GAT).

This model is trained in two phases. First, the model is trained on the maximum likelihood objective, which has two parts. The first objective is $\mathcal{L}_{seq} = -\frac{1}{|D|} \sum_{(\mathbf{x},\mathbf{y}) \in D} \log p(\mathbf{y}|\mathbf{x};\theta)$, which is simply the negative of the log likelihood that the predicted summary, $\mathbf{y}$, is the correct summary given the input text, $\mathbf{x}$. The second objective is $\mathcal{L}_{mask} = -\frac{1}{N_v} \sum_{v_i \in D} m_i \log \hat{m}_i + (1 - m_i) \log(1 - \hat{m}_i)$, which acts as the objective to label node salience, or whether the generated nodes appear in the reference summaries of a particular document. In this case, $\hat{m}_i = sigmoid(\mathbf{u}_2 \mathbf{v}_i)$ for each node $\mathbf{v}_i$ and represents the computed mask for that particular node. m_i represents the true mask for the node $\mathbf{v}_i$. The two objectives are combined to form $\mathcal{L}_{ml} = \mathcal{L}_{seq} + \mathcal{L}_{mask}$.

After optimizing the model based on the previous objective, the model is then trained using a Deep Reinforcement Learning method. In essence, the model is trained to perform well on a multiple choice test, the method more formally known as a multiple choice cloze reward. This method involves generating two summaries. First, a summary is generated by sampling from the distribution of tokens generated by the model ($\mathbf{y}^s$). The next summary is generated by simply greedily choosing the most likely tokens in the generated distribution, also used as a baseline ($\hat{\mathbf{y}}$). The objective function in this scenario is the following: $\mathcal{L}_{rl} = -\frac{1}{|D|} \sum_{(\mathbf{y}^s, \mathbf{x}) \in D} (R(\mathbf{y}^s) - R(\hat{\mathbf{y}})) \log p(\mathbf{y}^s | \mathbf{x}; \theta)$ where R is the reward function defined for this reinforcement learning problem. $R(\mathbf{y})$ is calculated as the sum of the ROGUE score (discussed below) and the multiple choice cloze score obtained for a particular prediction.

The multiple choice cloze score is implemented as a multiple choice test, where the model is evaluated on how well it can predict an argument pair (e.g. in the phrase "dogs drink water", the words "dogs" and "water"), along with the predicate that comes in between an argument pair (e.g. the word "drink" in the previous example). Tying everything together, the model is optimized both for predicting the highest likelihood summary from a source document, while also being trained using reinforcement learning to produce factual summaries via the attention mechanism. We use the ASGARD model and compare the generated results with the baseline as described below.

3.4 Evaluation

3.4.1 Fluency: ROUGE, BERTScore

In order to evaluate sequence to sequence summarization models, multiple metrics have been devised, which evaluates how similar the summary and original article are on a semantic level. The most popular metric is ROUGE [12] or Recall-Oriented Understudy for Gisting Evaluation. This metric compares the number of n-grams between the input article and output summary. For extractive summaries this is a great measure of model fluency because it compares exact sequences of words. Despite its popularity there are problems with this metric, namely that there must be an exact match between n-grams, which makes abstractive summaries score poorly. BERTScore [22] computes similarity over predicted and actual tokens rather than detecting exact matches. The similarity score more accurately captures the meaning behind sentences, which is fairer when scoring abstractive summaries fluency.

3.4.2 Factuality

As proposed by Pagnoni et al. [17], it is appropriate to evaluate factuality through multiple approaches. The authors propose a linguistically motivated typology of factual errors. Factual errors can be divided the following three categories:

1. Semantic Frame Errors - semantic frame is a schematic representation of an event, relation, or state which consists of a predicate and list of participants called the frame elements [2]
2. Discourse Errors - factual errors that arose from erroneous links between discourse segments, these kinds of errors cannot be captured by the semantic frame
3. Content Verifiability - content cannot be linked to source article

Based on their benchmark, we leverage two varied automated factuality metrics to evaluate the baseline model and ASGARD - FEQA and FactCC. Following their approach, we evaluate the usefulness of these metrics by computing correlation between the metrics and human annotation of factuality.

3.4.3 FactCC

The first factuality metric we used is FactCC [10]. FactCC evaluates factuality by comparing individual sentences The FactCC model was taken from a pre-trained Transformer-based model - uncased, base BERT architecture and later fine-tuned by Kryscinski et al. [10]. This model outputted whether a summary is consistent or inconsistent with the document. We utilized the final trained model, and did not apply training of our own.

We generated test data in JSON format from graph model and the baseline model. Each row of the testing data consists of the original article, and one sentence of the summary. This was fed into the FactCC model and the accuracy was calculated.

3.4.4 FEQA

Another factuality metric that we considered is FEQA [5], a faithfulness metric based on a question-answering (QA) framework. FEQA quantifies "faithfulness" by first using a QA model to generate questions and answers from source documents. The authors of the original FEQA paper use a fine-tuned BART language model to turn sentences such as "Joe was born in Brooklyn" into questions such as "Where was Joe born?". These questions are generated on a sentence by sentence basis from the source document.

A separate model is used to answer the generated questions using information from the generated summary. In order to answer the generated questions, the authors use a fine-tuned BERT-base model. If the answers from the generated summary either don't exist or are incorrect, the model marks it as "unfaithful". A f1 score is computed for each of the document-summary pairs, with higher scores indicating higher faithfulness and subsequently stronger summary factuality. We used FEQA as a metric as it was one of the metrics listed in the FRANK paper, although it was listed as a metric that is less correlated with human judgement when compared to FactCC.

3.4.5 Manual Evaluation

In addition to assessment through quantitative, automated metrics, we additionally employed an experiment of more qualitative, human nature. We recruited five graduate students for this experiment. Each graduate student was assigned to ten different articles from the CNN/Daily Mail dataset, for a total of fifty articles. In addition to the original article, we also provided the subjects with two summaries: one generated by by Baseline as described in section 3.3, and the other generated by ASGARD. After instructing the subjects to read these three pieces of text, we presented them with the following questions for each summary:

1. Is the generated summary factually correct? I.e. for each claim in the summary, can you find a part of the original article confirming this claim?
2. If not, please comment on the parts of the summary that are not factual.
3. Is the generated summary fluent? I.e. are the sentences grammatically correct and complete, and could one reasonably expect a human to have written this text?

We recorded binary answers for questions 1 and 3, and a free-form text response for question 2, that we qualitatively evaluated.

Figure 1 displays the interface[2] that we provided for this experiment, and which the subjects used to read the original articles and the summaries.

3.5 Model Implementation Details

The baseline model for our experiments is a pre-trained RoBERTa transformer fine-tuned to summarize CNN/Daily Mail.[3] We compare this baseline to the architecture presented by ASGARD [9]. The ASGARD model is trained with document graph encoding for 70k steps with gradient clipping of 2, learning of 0.001 and a batch size of 32. Summaries are generated by both models with beam search size 5.

4 Results

4.1 Manual evaluation

The results for the manual evaluation experiment are shown in Table 4. The score refers to the percentage of articles that were marked factual or fluent, respectively, by the annotators.

[2]A live interface is available here: `https://share.streamlit.io/anton164/graph-factuality/main/ui/compare_summary.py`

[3]We leverage the patrickvonplaten/roberta2roberta-cnn_dailymail-fp16 HuggingFace checkpoint.

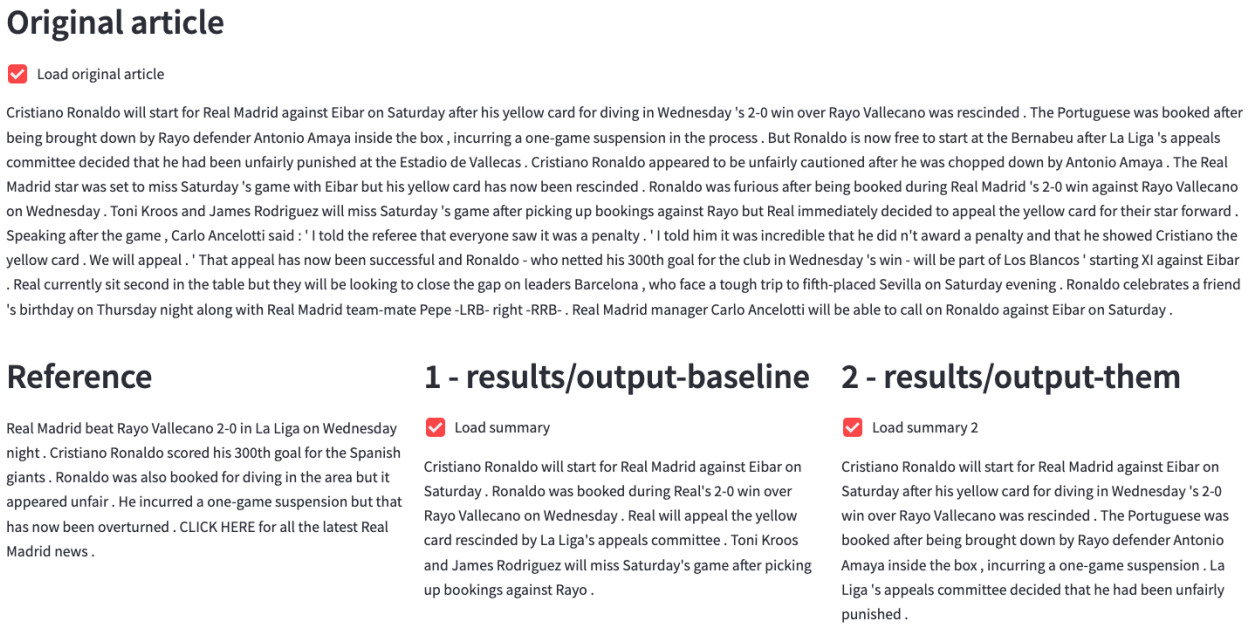

Figure 1: Summary evaluation interface

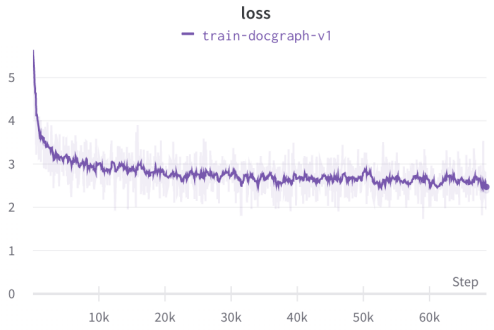

Table 3: Training loss for ASGARD model

Metric	Score
Factuality (baseline)	72%
Factuality (graph)	84%
Fluency (baseline)	76%
Fluency (graph)	56%

Table 4: Result of manual evaluation

As one may expect, the factuality score for the graph-based model is significantly higher than that for the baseline. At the same time, the graph-based model produces summaries that were marked as less fluent, when compared to the baseline model. There appears to be a trade-off between both metrics.

We will now present a set of samples that illustrate these points. Table 5 displays the results related to summary 16 from the CNN/Daily Mail dataset. We note that the baseline summary makes two non-factual claims: (1) that the fire was not deliberate, according to Martin Fletcher, and (2) that it occurred during half-time of Bradford's Valley Parade (which is a football club, not a game). All claims presented by the graph-based summary are correct. However, two sentences lack fluency. These have been marked in blue. Turning to document 40, we similarly find a factual error in the baseline summary: it claims the prison housed criminals in 1996, when it was in fact closed. This fact is even highlighted in the sentence before. The graph-based summary makes no such non-factual claims, but still introduces a pair of sentences lacking fluency, again marked in blue.

Source document
The former chairman of Bradford City was linked to eight other fires before the Valley Parade blaze that killed 56, a new book has claimed. Author Martin Fletcher claims the devastating fire was not an accident and has revealed a sequence of other blazes at businesses owned by or associated with Stafford Heginbotham, the club's chairman at the time. West Yorkshire police today said in a statement that they would review any fresh evidence surrounding the tragedy. [...] Fletcher, a survivor of the blaze, spent 15 years researching the disaster in which his brother Andrew, 11, his father John, 34, his uncle Peter, 32, and his grandfather Eddie, 63, were all killed. [...] The fire broke out near half-time of Bradford's game against Lincoln on May 11, 1985, and was thought to have been started by a spark from a match or a cigarette dropped through a gap in the wooden frame of the main stand on to piles of rubbish which had been collecting for years.
Baseline summary
Author Martin Fletcher claims the devastating fire was not started deliberately and was caused by a discarded cigarette. He spent 15 years researching the tragedy in which his brother Andrew, 11, his father John, 34, and grandfather Eddie, 63, were all killed. The fire broke out near half-time of Bradford's Valley Parade on May 11, 1985. It was thought to have been started by a cigarette dropped through a gap in the wooden frame of the main stand on to piles of rubbish which had been collecting for years.
Graph-based summary
The former chairman of Bradford City was linked to eight other fires before the Valley Parade blaze that killed 56, a new book has. Author Martin Fletcher claims the devastating fire was not an accident and has revealed a sequence of other blazes at businesses owned by a discarded cigarette. West Yorkshire police say they would review evidence.

Table 5: Document 16

Source document
Holmesburg Prison's 101-year history was filled with violent riots, bloody beatings and most shocking of all - the cruel chemical experimentation on its inmates. Since its construction in 1895, its high walls housed some of the country's most dangerous criminals whose uprisings ended in death and mass-injury. And haunting images published in 2011 proved that government doctors used the jail in Philadelphia, Pennsylvania to test chemical substances on inmates and disabled American citizens. Nearly two decades after its closure in 1995, all that remains of one of the most notorious and violent prisons in American history is cracked paint, crumbling hallways and empty cells which once housed its human guinea pigs. [...] in one incident in 1952, an inmate found dead in a bathtub with an autopsy concluding that he was beaten to death. [...] Since its closure in 1996, one of the few things that remain in tact in Holmesburg Prison are the old books read by prisoners who were once used as human guinea pigs.
Baseline summary
Holmesburg Prison in Philadelphia, Pennsylvania, was closed in 1995. In 1996, its walls housed some of the country's most dangerous criminals. Since then, nearly two decades after its closure, all that remains is crumbling. Pictures emerged in 2011 showing the eerie silence and decay of the prison. One inmate was found dead in a bathtub with an autopsy concluding that he was beaten to death.
Graph-based summary
Holmesburg Prison 's 101-year history was filled with violent riots, bloody beatings and the cruel chemical experimentation on its inmates. Since its construction in 1895, its high walls housed some of the country's most dangerous criminals. And that government doctors used the jail in Philadelphia, Pennsylvania to test chemical substances on inmates and disabled American citizens. Holmesburg Prison are the old books read by prisoners who were used as human guinea pigs.

Table 6: Document 40

4.2 Fluency and Factuality Evaluation

	Fluency				Factuality	
Model	Bert-Score	Rouge-1	Rouge-2	Rouge-L	FEQA (F-1)	FactCC Accuracy(%)
Baseline	81.84	41.30	18.83	28.57	38.4	60.9
ASGARD	82.54	43.84	20.24	29.60	**38.6**	**81.6**

Table 7: Evaluation Metrics Calculated on Baseline Model and ASGARD

When considering the fluency and factuality metrics on the baseline and ASGARD outputs, the table above shows that the ASGARD outputs performed significantly better than the baseline model in almost every metric. For fluency in particular, the metrics were able to pick up that the ASGARD model generated more fluent metrics overall, however these metrics were unable to pick up on some of the nuances that the manual evaluators were able to observe. One such nuance was that the ASGARD model often produced nonsensical outputs towards the end of the generated summaries, while the first few sentences were on point. The Rouge scores specifically were significantly higher for the ASGARD generated summaries, indicating that the metric found the ASGARD summaries to be more fluent overall. This indicates that the chosen fluency metrics may not be well correlated with human judgements.

The factuality metrics showed similar empirical results as the fluency metrics. The ASGARD model performed better than the baseline, which is also corroborated by our manual findings. In almost every case, we found the ASGARD model to produce factual summaries, whether or not they were always fluent. However, between the two metrics, FactCC was the preferred metric. FEQA produced results that were in line with what we had expected, however, it took much longer to produce results. Using the computation setup that we had, it took over 14 hours to generate 40 scores for the ASGARD model and 20 summaries for the baseline model (1 for each of the manually judged summaries). With over 11,000 samples in total, generating a FEQA score for each sample proved to be intractable. The FactCC method took only a few minutes in order to generate scores for all 11,000 documents, and was also shown to be more correlated with human judgement.

4.3 Open IE evaluation

The relation sets generated by Stanford Open IE were sometimes helpful for understanding how ASGARD avoided errors. In many cases where the baseline model was not factual but ASGARD was, the Open IE relations did not capture any information related to the erroneous statement. In the output summary the ASGARD model would select different subjects to use in the summary. However it also sometimes directly corrected an error the baseline model made even though the relation sets did not capture additional information. This is likely due to the graph attention network giving additional guidance away from some poor answers, making it stick closer to the original text, as opposed to giving it deeper insights.

4.4 Correlation

We calculated the correlation coefficient on human reviewer's judgment of factuality with the FactCC and FEQA scores. We find that FactCC correlates better with human judgments, however, the overall correlation is still fairly low.

	FactCC	FEQA
Pearson's Correlation Coefficient	0.12	-0.02

Table 8: Pearson's Correlation

5 Discussion

Our experimental results reveal two main insights related to metrics of factuality. Based on the automatic and manual evaluation of generated summaries, we conclude that the graph-augmented

summarization model improves factuality at the expense of fluency. It is worth exploring whether such a trade-off is truly *inherent* to summarization, or simply a result of the current methods used. In comparing two different models, we do not have sufficient information to generalize this.

Second, few of the factuality metrics prevalent in summarization research correspond to factuality as rated manually, by human annotators. It appears that, the manner in which humans evaluate factuality is not fully captured in contemporary factuality metrics. When analyzing summaries marked by humans as non-factual, it was revealed that these truly contained factual inconsistencies. This suggests that the lack of correlation between human judgment and automated factuality metrics is the result of a limitation of the factuality metrics, rather than human judgment. Note that the factuality metrics employed, FactCC and FEQA, evaluate the factuality of individual sentences in the summaries. As soon as one sentence in a summary was non-factual, we marked the entire summary as such. Therefore, the longer the summary, the lower the chance that it was found to be factual. This may not be reflective of the type of factuality we intend to measure. Finally, the computation of factuality scores was further hindered by the computational complexity of the methods used, specifically FEQA.

6 Future Work

In future work, we would first try to understand and mitigate the fluency issues of the graph-augmented summarization model through attribution [20]. Next, we would evaluate the model on different datasets which are more abstractive, such as XSum. We would also like to improve the model's ability to reason about "out-of-article" knowledge and alleviate extrinsic hallucinations by encoding knowledge from external knowledge graphs inspired by methods such as Gunel et al. [8].

7 Individual Contributions

All authors contributed equally to writing the paper. A.A. conceptualized the study, adapted and trained the summarization models, and analyzed the results. Z.L. recruited team members, performed manual review and executed all computations related to FactCC. L.D. created the test harness for Rouge and Bertscore, performed manual reviews, and analyzed open IE relations. L.K. led the manual evaluation process and aided in producing the baseline summaries. A.P. participated in the manual review process and calculated all FEQA scores.

References

[1] G. Angeli, M. J. Johnson Premkumar, and C. D. Manning. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 344–354, Beijing, China, July 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-1034. URL https://aclanthology.org/P15-1034.

[2] C. F. Baker, C. J. Fillmore, and J. B. Lowe. The Berkeley FrameNet project. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 86–90, Montreal, Quebec, Canada, Aug. 1998. Association for Computational Linguistics. doi: 10.3115/980845.980860. URL https://aclanthology.org/P98-1013.

[3] M. Cao, Y. Dong, J. Wu, and J. C. K. Cheung. Factual error correction for abstractive summarization models, 2020. URL https://arxiv.org/abs/2010.08712.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[5] E. Durmus, H. He, and M. Diab. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. *CoRR*, abs/2005.03754, 2020. URL https://arxiv.org/abs/2005.03754.

[6] A. R. Fabbri, W. Kryściński, B. McCann, C. Xiong, R. Socher, and D. Radev. Summeval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409, 2021.

[7] R. Ferreira, L. de Souza Cabral, R. D. Lins, G. P. e Silva, F. Freitas, G. D. Cavalcanti, R. Lima, S. J. Simske, and L. Favaro. Assessing sentence scoring techniques for extractive text summarization. *Expert systems with applications*, 40(14):5755–5764, 2013.

[8] B. Gunel, C. Zhu, M. Zeng, and X. Huang. Mind the facts: Knowledge-boosted coherent abstractive text summarization. *arXiv preprint arXiv:2006.15435*, 2020.

[9] L. Huang, L. Wu, and L. Wang. Knowledge graph-augmented abstractive summarization with semantic-driven cloze reward, 2020. URL https://arxiv.org/abs/2005.01159.

[10] W. Kryscinski, B. McCann, C. Xiong, and R. Socher. 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, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.750. URL https://aclanthology.org/2020.emnlp-main.750.

[11] G. Lample and A. Conneau. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*, 2019.

[12] C.-Y. Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W04-1013.

[13] Y. Liu and M. Lapata. Text summarization with pretrained encoders. *arXiv preprint arXiv:1908.08345*, 2019.

[14] Y. Mao, X. Ren, H. Ji, and J. Han. Constrained abstractive summarization: Preserving factual consistency with constrained generation, 2020. URL https://arxiv.org/abs/2010.12723.

[15] J. Maynez, S. Narayan, B. Bohnet, and R. T. McDonald. On faithfulness and factuality in abstractive summarization. *CoRR*, abs/2005.00661, 2020. URL https://arxiv.org/abs/2005.00661.

[16] F. Nan, C. N. d. Santos, H. Zhu, P. Ng, K. McKeown, R. Nallapati, D. Zhang, Z. Wang, A. O. Arnold, and B. Xiang. Improving factual consistency of abstractive summarization via question answering, 2021. URL https://arxiv.org/abs/2105.04623.

[17] A. Pagnoni, V. Balachandran, and Y. Tsvetkov. Understanding factuality in abstractive summarization with frank: A benchmark for factuality metrics, 2021. URL https://arxiv.org/abs/2104.13346.

[18] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding with unsupervised learning. 2018.

[19] A. See, P. J. Liu, and C. D. Manning. Get to the point: Summarization with pointer-generator networks. *CoRR*, abs/1704.04368, 2017. URL http://arxiv.org/abs/1704.04368.

[20] J. Xu and G. Durrett. Dissecting generation modes for abstractive summarization models via ablation and attribution, 2021. URL https://arxiv.org/abs/2106.01518.

[21] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization, 2019. URL https://arxiv.org/abs/1912.08777.

[22] T. Zhang*, V. Kishore*, F. Wu*, K. Q. Weinberger, and Y. Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SkeHuCVFDr.

[23] C. Zhu, W. Hinthorn, R. Xu, Q. Zeng, M. Zeng, X. Huang, and M. Jiang. Enhancing factual consistency of abstractive summarization. 2020. doi: 10.48550/ARXIV.2003.08612. URL https://arxiv.org/abs/2003.08612.