Predicate-Targeting Semantic Perturbations for Robust Factual Abstractive Summarization

Aayush Karan Harvard University akaran@college
Geoffrey Liu Harvard University geoffreyliu@fas
Lingdi Xu Harvard University lingdixu@g
Kat Zhang Harvard University katherinezhang@college
.harvard.edu

Abstract

Neural abstractive summarization models are able to generate summaries which mimic human references well in fluency. However, summarization models face issues with preserving the factual accuracy of the source document, which is a critical metric in downstream tasks and a major limiting factor of safe real-world applications of these models. One approach is to pipeline a separate factual corrector model (FCM) at the end of some summarization model. The FCM then corrects factual errors in the generated summary based on the content of the source document. However, existing FCMs are trained on errors specific only to entities in the summary. In this paper, we aim to improve upon existing FCMs by using a novel approach based on semantic triples to artificially perturb the data used in training to account for predicate errors, a type of error that the current FCMs fail to correct. We study the performance of our model trained on this data through empirical analysis using the XSum dataset alongside several well-known metrics for summary quality and factual correctness. Our results demonstrate that our model not only successfully corrects predicate errors but also generates higher quality summaries relative to existing FCMs with respect to the aforementioned metrics.

1 Introduction

Abstractive summarization – the task of generating a summary of a source document – is a well studied problem in natural language processing (NLP), with widespread applications ranging from assistance with medical queries to commercial convenience for newsletters, books, scientific journals, and other literature [19]. Many papers have found that transformer sequence-to-sequence (seq2seq) models are able to demonstrate fluency and grammatical correctness in the summaries they produce. However, human evaluations of these machine generated summaries have found that they often fail to factually represent the content in the original source document, sometimes up to 30% of the time [3, 2, 7, 14, 8]. Different types of factual errors have been documented for a wide variety of summarization models, including contradiction of the original meaning, erroneous fabrication of words, erroneous entities, and errors in sentence predicates, among many others [13].

Much of the previous literature (see Section 2) is concerned with correcting entity errors, which refers to corruptions of a key subject in a summary, typically a noun/pronoun, or of surrounding descriptors, such as numbers or dates [2]. However, in this paper we will focus on a more intractable type of factual error known as a *predicate* (or *relation*) *error*. These errors are distortions in phrases (known as *predicates*) that semantically link two key subjects or objects, destroying or even inverting their commonsense meanings. In this sense, relation errors are more difficult to correct than entity errors, which can leverage the power of current named entity recognition models to pinpoint the exact word location at which the error occurs.

To this end, we utilize the information extraction tool OpenIE [1], which extracts semantic triples from articles in the form of (subject, relation, object), representing common errors involving relations that may appear in generated summaries. We propose a novel perturbation mechanism that provides a gradient of semantic replacements to the relations extracted by OpenIE, using this to generate a dataset of corrupted summaries mixed with gold, ground-truth summaries. To obtain our model, we train a BART denoiser [10] on this perturbed dataset such that our model inputs a draft summary and its source document, outputting a factually corrected summary. In order to comprehensively capture our model's performance on the factual correction task, we manually evaluate a random sample of corrupted summaries that were corrected by a state-of-the-art baseline and our trained model. In addition, we utilize the renowned ROUGE [12] alongside FactCC [9] and FEQA [5] as metrics for syntactical quality and factual content respectively. Our experiments demonstrate that our model can successfully identify and rectify relation errors in summaries, resulting in high quality summaries that outperform the baseline in each of the aforementioned metrics.

2 Related works

Recently, [17] developed a comprehensive typology of factual errors that arise in summarization, identifying predicate, entity, and out-of-article errors as the most prevalent sources of factual error in machine summary their typology generalizes previous ones proposed by [14, 8] as it provides a more fine-grained deconstruction of the different sources of factual summarization errors. The authors identify six main types of factual errors that appear in summarization: Entity, Predicate, Circumstance, Coreference, Discourse-Link, and Out-of-Article. A description of these errors can be found in Table 1.

Error	Description
Entity (20%)	Entities in the summary are incorrect
Predicate (15%)	Summary predicate is inconsistent with the article
Circumstance (15%)	Predicate descriptors (e.g. location, time) are incorrect
Coreference (<2%)	Incorrect pronoun reference
Discourse-Link (<2%)	Incorrect temporal/causal link between statements
Out-of-article (30%)	Contains information not present in the article

Table 1: Types of factual errors in summarization and how prevalent they are, according to analysis of machine generated summaries of the XSum dataset by [17].

Much of the existing work in factual correctness for abstractive summarization can be conceived as addressing one or more of these types of errors. In particular, previous works have mainly addressed factual correction with respect to entity, out-of-article, circumstance, and to some extent, coreference errors while ignoring predicate/relation and discourse link errors [20, 2]. We believe that this is mainly because entity, out-of-article, and circumstance errors are easy to detect, and data augmentations that create entity-based corruptions in summaries can be easily generated due to wide-spread availability of entity-recognition models that are able to identify entities, pronouns, dates, etc.

While our focus on predicate errors is largely a novel direction, the vast prior literature still offers key insights towards formulating and evaluating our approach. This prior literature falls into two main categories: one aims to design neural architectures tailored to correcting factually incorrect summaries while the other aims to design metrics that accurately reflect the factual consistency of summaries with their sources.

2.1 Models for Generating Factual Summaries

One approach to enhancing network architectures for factual consistency is to incorporate factual information into the embeddings within the transformer seq2seq encoder-decoder that generate a summary to begin with. In [3], the authors propose a dual-attention seq2seq model that conditions summary generation on both the original source document alongside extracted factual information. [20] constructs a knowledge graph from factual relations, using a graph attention network to create embeddings that are subsequently included within the decoder attention network that generates summaries.

[20] also introduces another component to the neural architecture known as the Factual Corrector Model (FCM) that templates the design for our own model. Rather than influencing the initial generation of summaries, the FCM is an independently trained model whose task is to correct draft summaries if factually erroneous. In this way, we can reframe the problem of generating factually correct summaries as a pipeline of neural models which includes a summary generation phase and a summary correction phase. This idea is extended in [4], whose model is fine-tuned as a denoising autoencoder to recognize entity-based errors. This is further developed in [2], where the authors propose a BART-based model for factual error correction fine-tuned on corruptions involving erroneous entities, numbers, dates, and pronouns.

For model training, [15] explores data filtering, contrastive learning, and joint entity and summary generation to improve performance. In fact, [15] shows that a simple filtering of entity hallucinations (entities that appear in the summary but do not exist in the source) reduces entity error by twenty percent, a technique we incorporate into our own training methodology.

2.2 Factual Correctness Metrics

On the evaluation side, many papers utilize ROUGE [12] as a baseline metric. However, ROUGE evaluates token-based accuracy but is not as effective at measuring factual accuracy. [9] proposes FactCC, a weakly supervised BERT-based model that is pretrained on an artificially-generated dataset of transformed summaries from the CNN/Dailymail dataset, and its extension FactCCX, which uses additional span selection to improve the classification of summaries and provide explainability. FactCC shows correlation with human-based evalution – however, [20] shows that its performance may be degraded when applied to different datasets. [5] and [18] introduce FEQA (Faithfulness Evaluation with Question Answering) and QAGS (Question Answering and Generation for Summarization), respectively. Both are question and answer models, which extract question and answer pairs from the summary and then compares these answers against answers extracted from the source article in order to determine factual accuracy. Given a summary sentence, FEQA first produces a list of questions asking about key information in the sentence and their corresponding answers. Then a QA model is used to predict answers from the source document. Comparing the average F1 score against the “gold” summary reflects how faithful the summary is. The higher the score is, the more consistent the summary is with the source document. Although these metrics generate significantly improved correlation with human-based evaluation, they are also more costly to train and use.

3 Approach

We use the post-editing factual correction model (FCM) framework [20], [2] to design our model for correcting predicate and relation errors. These types of models take a given summary along with the source, and output a factually corrected summary. Our model is trained on a set of corrupted summaries from the XSum dataset. These corruptions are generated through perturbations of the relation triples generated by OpenIE.

3.1 Perturbation Generation

To compel our model to adequately correct for predicate and relation errors, we first use Stanford's OpenIE [1] to generate a semantic triple (s, r, o) consisting of a subject s and object o (both typically descriptive spans around nouns), along with a relation r that is a predicate phrase establishing dependency between s and o. These relation triples form the basis of our predicate and relation perturbation strategy. In our approach we consider three types of perturbations: a) S-O Swap, b) Naïve Semantic Tree Search, and c) Smart Swap.

The S-O Swap interchanges the subject with the object in the relation triple, inducing the mapping $(s, r, o) \to (o, r, s)$. In our experimentation, these type of perturbations assist in teaching the model the sequential dependency between subjects and objects, rather than simply associating the set $\{s, o\}$ with r.

The other types of perturbations directly target relations and are of the form $(s, r, o) \to (s, r', o)$, relying on our Semantic Tree Search (STS): a novel method for generating a gradient of word-based semantic substitutions. For the STS we utilize WordNet [6], an open source database that groups words into sets of synsets, or conceptually aligned synonyms. Given a word, we generate a set of synonyms and antonyms, representing opposite sides of a semantic gradient. Choosing one of these sides at random and selecting a random synset within, we traverse a minimal spanning tree of related synsets up to random depth, giving a replacement word. By iterate this process for a random number of times, we are able to generate vast variety for replacements that maintain the same semantic context.

Naiv̈e Swap simply replaces an arbitrary verb, adverb, or adjective from the summary with a replacement obtained from STS. Smart Swap extracts relation triples from the summary and replaces at most four verbs, adverbs, or adjectives found in the associated relation, fundamentally corrupting the commonsense content of the summary. All verb tenses are corrected to match that of its predecessor.

3.2 Model

3.3 Datasets

We perform data augmentation and train our model using the benchmark dataset XSum [16], which contains 227K news articles and their corresponding human reference summaries.

Type	Example
Original	Fugitive US whistleblower Edward Snowden is still in the transit area at Moscow airport, Vladimir Putin has confirmed.
S-O Swap	Fugitive US whistleblower **Vladimir Putin** is still in the transit area at Moscow airport, **Edward Snowden** has confirmed.
Naïve Swap	Fugitive US whistleblower Vladimir Putin is still in the **reports** area at Moscow airport, Edward Snowden has confirmed.
Smart Swap	Fugitive US whistleblower Vladimir Putin is **no longer** in the transit area at Moscow airport, Edward Snowden has confirmed.

Table 2: Example relation perturbations.

3.4 Model Implementation

Through the artificial data perturbation process we are provided with a data triplet, $(\mathcal{D}, \mathcal{S}, \mathcal{S}_c)$, which corresponds to the source document $\mathcal{D}$, the original summary $\mathcal{S}$ and the corrupted summary $\mathcal{S}_c$. Given this data triplet, the training object is to recover the true summary $\mathcal{S}$ given the original source document and the corrupted summary $(\mathcal{D}, \mathcal{S}_c)$. To analytically solve this problem, we can express this as maximising a the likelihood of $\mathcal{P}(\mathcal{S} \mid \mathcal{S}_c, \mathcal{D})$ using an encoder-decoder architecture. To do this, we concatenate the summary with the original document and feed the concatenation as input to the model.

We used a pre-trained BART model [11] which is pretrained as a denoising Seq2Seq encoder-decoder model. The model inputs are the source document, $\mathcal{D}$ and the corrupted summary, $\mathcal{S}_c$ (which are inputted to the model with a separator token between them) and trained to recover the original summary, $\mathcal{S}$. Random noise such as deletion and repetition of random tokens is applied to 10% of the inputs for the model as the BART model is trained using these types of noising during pre-training.

3.5 Training process and hyper-parameters

We train our model using our three proposed corruptions, the S-O, Naiv̈e, and Smart Swaps. Following the perturbation frequency from [2], these corruptions comprise of 30% of the summaries in the data used to train the model. We also corrupt 10% of summaries through back-translation to introduce variation in the vocabulary used within the summary without changing the semantic meaning of the summary. The rest of the summaries, 60%, are not explicitly corrupted, however each summary has a 10% chance of random noise token corruption, where 80% of these (8% total) has one of their tokens deleted and 20% (2% of total summaries) having one token duplicated.

A learning rate of 5×10^{-5}, with a polynomial learning rate scheduler and 2000 warm-up steps to train the model for a total of 5 epochs. We used HuggingFace's pretrained BART-base model, which took 16 hours to fine-tune on one GPU.

3.6 Baseline Factual Corrector Model

For evaluation comparison, we also train a baseline factual corrector model (EntitySwap) found in [2]. Their corruptions only included entity and circumstance-based corruptions, such as entity, pronoun, date and number swaps. Since no model checkpoints are given by the authors, we trained this model using the same hyperparameters and corruption process described in 3.5.

4 Experiments

In order to evaluate our model, we created a set of 1,147 corrupted summaries from the XSum test dataset using our perturbation methods. Of these, we chose 100 summaries for each type of corruption and allowed our model and the baseline model to attempt to correct these summaries by feeding in the corresponding concatenated summary and document.

We perform three types of model evaluations to demonstrate the efficacy of our model as opposed to the baseline, as well as issues that may arise when evaluating summaries using metrics based on current factual summarization methods. The three evaluations we will consider in this section are as follows:

- Using common factual accuracy metrics to evaluate our model's predictions against baseline model and reference summaries
- Evaluating FactCC's ability to detect predicate errors
- Manually evaluating our model's ability to correct predicate errors against the baseline

4.1 Metrics

The first type of evaluation is to benchmark our model to commonly used factual accuracy metrics ROUGE [12], FactCC [9], and FEQA [5]. This evaluation does not, and is not meant to, prove or disprove the accuracy or effectiveness of the model we proposed as ROUGE is not a measure of factual consistency. Rather, we view this evaluation as an interesting observation that we can interpret in the context of our next two evaluations.

	ROUGE-1	ROUGE-2	ROUGE-L
Corrupted	0.94	0.83	0.89
EntitySwap	0.93	0.84	0.90
Our Model	**0.96**	**0.92**	**0.95**

Table 3: ROUGE and ROUGE-L metric scores for corrupted summaries, as well as our model's corrections and EntitySwap's corrections, compared against the ground truth.

Table 3 reports ROUGE-1/2/L F1 scores for the summaries generated by both models on the 300 sample summaries forming the test set, as well as the original corrupted summaries, against the reference summaries provided by the XSum dataset. [1] The results demonstrate that the baseline model does not show much improvement above the corruptions, while our model does, which indicates that our model may be more effective in correcting predicate-based errors than the baseline.

	FactCC	FactCCX	FEQA
EntitySwap	0.16	0.30	0.19
Our Model	**0.21**	0.30	**0.20**

Table 4: FactCC, FactCCX and FEQA scores for summaries generated by our model and EntitySwap.

Table 4 reports FactCC scores for the our model and the baseline. Our model demonstrates a significant improvement in FactCC score over the baseline model. However, FactCCX scores for the two are identical.

In addition to FactCC and FactCCX, we use FEQA, which utilizes question answering to score the factual correctness of a generated summary against its corresponding original document. From Table 4 we can see that our model receives a higher FEQA score than the baseline model. In order to determine whether FactCC, FactCCX or FEQA can detect predicate errors, we turn to our next evaluation metric.

4.2 Factcual Consistency Evaluation

The second type of evaluation is to demonstrate whether FactCC and FEQA detect predicate errors when scoring summaries. We construct a simple test, where we calculate the FactCC score and FEQA score for the correct summaries and the predicate or relation perturbed summaries.

	FactCC	FactCCX	FEQA
Corrupted	0.14	0.29	0.19
Gold Standard	0.22	0.31	0.23

Table 5: FactCC, FactCCX and FEQA scores for our corruptions and the gold truth summaries.

Table 5 reports FactCC, FactCCX and FEQA scores for the corrupted summaries in our test set, as well as their corresponding references. The differences in scores demonstrate that FactCC and FEQA can detect predicate-based factual errors, while FactCCX struggles to reflect predicate-based errors. From this conclusion, we can say that the difference in FactCC and FEQA scores in Table 4 shows that our model improves upon the baseline model in terms of factual accuracy.

[1] Since corrupted summaries only change a couple tokens in the ground truth summaries, our ROUGE scores are on the higher end. On generated summaries, our model keeps nearly the same ROUGE scores as baseline summary generator model as it is a factual corrector model, so we do not report these.

	Corrupted/Changed Correctly	Corrupted/Unchanged	Corrupted/Changed Incorrectly	Uncorrupted/Unchanged
Smart Swap (EntitySwap)	3	41	6	50
Smart Swap (Our Model)	**24**	17	9	50
Naive Swap (EntitySwap)	3	42	5	50
Naive Swap (Our Model)	**26**	20	4	50
S-O Swap (EntitySwap)	4	19	27	50
S-O Swap (Our Model)	**34**	3	13	50

Table 6: Manual evaluation of our model against the baseline for smart, naive, and subject-object swaps. In each category, 100 samples were chosen, and 50 of them were corrupted while the other 50 were not. This table displays the number of corrupted summaries that were changed correctly, incorrectly, and not changed at all for both models, as well as the number of uncorrupted summaries that remained unchanged.

4.3 Manual Evaluation

Finally, we manually evaluate our model's ability to correct predicate errors. In order to do this, we take the 300-sample test set of articles. For each of the 100 smart, naive, and subject-object swapped summaries, we randomly choose 50 of them to revert back to their uncorrupted versions to determine whether our model is able to leave the uncorrupted summaries unchanged. We compare how many summaries our model corrects to how many summaries are corrected by the baseline post error-corrector model trained purely on entity errors.

Table 6 shows the results from the comparison between the entity-detecting baseline model and our model. We count the number of summaries that were corrupted and changed, left alone, or changed in a way that was incomplete or incorrect, as well as the number of uncorrupted summaries that were left unchanged. First, we see that our model and the baseline model both do not change any uncorrupted summaries, thus showing that our model does not sacrifice accuracy on uncorrupted summaries.

Additionally, our model presents a striking improvement over the baseline for all three types of corruptions. Specifically, the baseline model changes at most 4 out of 50, or 8%, of corrupted summaries correctly, while our model changes at least 24 out of 50, or 48%, of corrupted summaries to their original meaning. We also note that the baseline model is largely unable to detect smart and naive swaps, whereas it is able to detect subject-object swaps, but changes most of the corrupted summaries incorrectly.

Table 7, Table 8, and Table 9 show example corruptions and the corresponding fixes by our model and the baseline EntitySwp model. These results demonstrate that our model is able to correct these corruptions back to their original meaning, sometimes through replacing the original verb from the ground truth summary and sometimes through finding a synonym for the original verb. In contrast, the baseline model is unable to return any of the corrupted example sentences to their correct meanings – in the case of the smart and subject-object swapped summaries, it introduces descriptors that contain false information, and in the case of the naively swapped summary, it actually flips the verb's meaning.

5 Conclusion

In this paper, we propose a novel method of generating summary perturbations that directly target relations. We use these corruptions in the context of a BART denoising task, training a factual corrector module to correct machine-generated summaries that contain relation errors. We compared our model to a previous factual corrector model (EntitySwap) and found that we achieved higher ROGUE scores as well as FactCC and FEQA scores. In addition, we found that our factual corrector was much more capable at correcting relation errors of each type that we generated artificially, without forcing corrections on already correct summaries. In the future, we plan to explore whether including entity corruptions along with relation corruptions in the BART training set improves performance as compared to using two pipelined factual correctors per error type. We also plan to conduct more testing to evaluate the performance of our model relative to state-of-the-art factual abstractive summarizers.

References

[1] Gabor Angeli, Melvin Premkumar, and Christopher Manning. Leveraging linguistic structure for open domain information extraction. 2015.

[2] Meng Cao, Yue Dong, Jiapeng Wu, and Jackie Chi Kit Cheung. Factual error correction for abstractive summarization models, 2021.

[3] Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. Faithful to the original: Fact aware neural abstractive summarization. *CoRR*, abs/1711.04434, 2017.

[4] Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. Unified language model pretraining for natural language understanding and generation. *CoRR*, abs/1905.03197, 2019.

[5] Esin Durmus, He He, and Mona Diab. Feqa: A question answering evaluation framework for faithfulness assessment in abstractive summarization. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.

[6] Christiane Fellbaum. *WordNet*. The Encyclopedia of Applied Linguistics., 2012.

[7] Yi-Chong Huang, Xia-Chong Feng, Xiao-Cheng Feng, and Bing Qin. The factual inconsistency problem in abstractive text summarization: A survey. *CoRR*, abs/2104.14839, 2021.

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

[9] Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard 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, November 2020. Association for Computational Linguistics.

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

[11] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *CoRR*, abs/1910.13461, 2019.

[12] Chin-Yew 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.

[13] Klaus-Michael Lux, Maya Sappelli, and Martha Larson. Truth or error? towards systematic analysis of factual errors in abstractive summaries. In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, pages 1–10, Online, November 2020. Association for Computational Linguistics.

[14] Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan Thomas Mcdonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of The 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

[15] Feng Nan, Ramesh Nallapati, Zhiguo Wang, Cicero Nogueira dos Santos, Henghui Zhu, Dejiao Zhang, Kathleen McKeown, and Bing Xiang. 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, April 2021. Association for Computational Linguistics.

[16] Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *CoRR*, abs/1808.08745, 2018.

[17] Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. Understanding factuality in abstractive summarization with frank: A benchmark for factuality metrics, 2021.

[18] Alex Wang, Kyunghyun Cho, and Mike Lewis. Asking and answering questions to evaluate the factual consistency of summaries. *CoRR*, abs/2004.04228, 2020.

[19] Yuhao Zhang, Derek Merck, Emily Tsai, Christopher D. Manning, and Curtis Langlotz. 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, July 2020. Association for Computational Linguistics.

[20] Chenguang Zhu, William Hinthorn, Ruochen Xu, Qingkai Zeng, Michael Zeng, Xuedong Huang, and Meng Jiang. Enhancing factual consistency of abstractive summarization, 2021.

	Sentence
Source	Writing in the Wall Street Journal, **Mike Pence** said the **Religious Freedom Restoration Act (RFRA)** had been "grossly misconstrued" as anti-gay… Signed into **state law last week**, the bill prevents the state from forcing people to provide services they say are contrary to their religion.
Gold Standard	The governor of Indiana has **defended a new law** that has unleashed a wave of condemnation across the country.
Corruption	The governor of Indiana has **settled a cubic yard law** that has unleashed a wave of condemnation across the country.
EntitySwap	The governor of Indiana has **settled a civil rights law** that has unleashed a wave of condemnation across the country.
Our Model	The governor of Indiana has **defended a landmark law** that has unleashed a wave of condemnation across the country.

Table 7: Example correction on a smart swap corruption. Our model is able to amend the corruption by replacing the verb and using a synonym for the original adjective, while the EntitySwap baseline model is unable to change the verb and uses an adjective that changes the meaning of the sentence.

	Sentence
Source	Mr Oyston sued Stephen Reed over material posted on a fans' webzine, Back Henry Street, in June 2015. Mr Reed's website posting claimed the club chairman entered into a foul-mouthed rant at him in public, held a gun in such a way as to make Mr Reed believe he was about to be shot at... [The judge] ordered **Mr Reed to pay Mr Oyston £30,000** and his legal costs.
Gold Standard	Blackpool Football Club's chairman Karl Oyston **has won** £30,000 in libel damages from an abusive fan who claimed Mr Oyston threatened him with a shotgun.
Corruption	Blackpool Football Club's chairman Karl Oyston **has existed** £30,000 in libel damages from an abusive fan who claimed Mr Oyston threatened him with a shotgun.
EntitySwap	Blackpool Football Club's chairman Karl Oyston **has been ordered to pay** £30,000 in libel damages from an abusive fan who claimed Mr Oyston threatened him with a shotgun.
Our Model	Blackpool Football Club's chairman Karl Oyston **has received £30,000** in libel damages from an abusive fan who claimed Mr Oyston threatened him with a shotgun.

Table 8: Example correction on a naive predicate swap corruption. Our model corrects the nonsensical verb created by the corruption by using a synonym, while the entity model actually changes the sentence to have opposite meaning from the original.

	Sentence
Source	Arsenal dominated with Gervinho failing to hit the target from a good position and Jake Kean making several key saves. Tomas Rosicky hit the bar after the break but the Championship side scored when Kazim-Richards followed in on Martin Olsson's shot… **Blackburn are yet to concede in the FA Cup this season and have now reached the quarter-finals** for the first time since 2007 and sit six points off the play-off places in the Championship.
Gold Standard	Colin Kazim-Richards's late goal stunned Arsenal as **Blackburn Rovers** reached the **FA Cup quarter-finals**.
Corruption	Colin Kazim-Richards's late goal stunned Arsenal as **FA Cup quarter** reached the **Blackburn Rovers -finals**.
EntitySwap	Colin Kazim-Richards's late goal stunned Arsenal as the **FA Cup quarter-** reached the **Blackburn Rovers** **semi-finals**.
Our Model	Colin Kazim-Richards's late goal stunned Arsenal as **Blackburn Rovers** reached the **FA Cup quarter-finals.**

Table 9: Example correction on a subject-object swap corruption. Our model is able to reverse the swap, while the EntitySwap baseline model is not only unable to do so, but also adds false information into the sentence.

A Impact Statement

Our model (and BART-based factual corrector models in general) can be applied to a variety of different settings in which summarization of articles or documents is necessary, ranging from medical documents [19] to news articles, as discussed in our paper. Our paradigm for generating perturbations can be used on any dataset to generate corruptions. Moreover, our approach is such that a factual corrector trained according to our data augmentation method can be appended at the end of any current summarization model.

Our model presents an improvement on a previously-unexplored space of factual errors that are nonetheless highly prevalent [17]. Its effectiveness in detecting and fixing predicate errors presents a step forward in combating the factual inaccuracies that can prevent the safe deployment of abstractive summarization methods in real-world settings. There are, however, potential risks to utilizing a factual corrector model to post-process generated summaries. For one, a factual corrector model runs the risk of introducing further inaccuracies into a generated summary. We see an example of this in our comparison to the baseline EntitySwap model, which can often change descriptors to include falsified information, rather than fixing predicate errors. While our model improves upon the baseline in terms of the number of summaries changed correctly and incorrectly, improvement upon the baseline does not indicate that our model is ready to be deployed immediately. Additionally, training on one dataset – in this case, XSum – means that our model best internalizes the summary style of gold standard summaries from that dataset. In order for it to be more extensible, more extensive data augmentation may be required.

We also note that there is a risk that our proposed relation-based data augmentation methods can be adopted to automatically corrupt short summary-like texts to generate misleading news article headlines, or even generate false relations between a patient and their diagnosis.

Additionally, our paper brings up the yet-answered question of a metric on which we can evaluate factual accuracy. While we argue that our perturbations are simple enough that the output of our model can be manually evaluated, this process is neither scalable nor extensible to more complex factual error correction tasks. Examples annotated by [17] often consist of multiple different types of errors combined, and they are often ambiguous. Evaluation of summarization models and factual correctors by Amazon's Mechanical Turk is still a method favored by many researchers, as metrics are often insufficient on their own to capture all types of errors and provide explainability [8]. However, the widespread use of Mechanical Turk necessitates that evaluators are fairly compensated and also questions how many different evaluators must look at each example in order to prevent a biased evaluation from influencing model performance measurements. In the long term, solving this problem would require creating better factual accuracy metrics, which introduces further trade-offs. Still, training more comprehensive and accurate factual accuracy metrics often requires high amounts of computing power.

Finally, on a larger scale, since models such as ours are not perfect and we do not possess foolproof metrics for evaluating factual accuracy, application to real-world settings introduce a dilemma: should summaries that are machine-generated or factually corrected be labeled, or presented differently to the general public? This question should be answered and addressed before a model like ours is used in a real-world setting.