# Measuring Factual Consistency of Abstractive Summaries

**Anonymous ACL submission**

## Abstract

Recent abstractive summarization systems fail to generate factual consistent – faithful – summaries, which heavily limits their practical application. Commonly, these models tend to mix concepts from the source or hallucinate new content, completely ignoring the source. Addressing the faithfulness problem is perhaps the most critical challenge for current abstractive summarization systems. First automatic faithfulness metrics were proposed, but we argue that existing methods do not yet utilize all "machinery" that this field has to offer and introduce new approaches to assess factual correctness. We evaluate existing and our proposed methods by correlating them with human judgements and find that BERTScore works well. Next, we conduct a data analysis, which reveals common problems, ways to further improve the metrics and indicates that combining multiple metrics is promising. Finally, we exploit faithfulness metrics in pre- and post-processing steps to decrease factual errors made by state-of-the-art summarization systems. We find that simple techniques like filtering training data and re-ranking generated summaries can increase the faithfulness by a substantial margin.

## 1 Introduction

Abstractive summarization is the task of generating an informative and fluent summary that is faithful to the source document. Recent progress in neural text generation has led to significant improvements and well-performing state-of-the-art abstractive summarization systems (Zhang et al., 2019; Lewis et al., 2020; Qi et al., 2020). Despite these advances, recent models fail to meet one of the essential requirements of practical summarization systems: information of a generated summary must match the facts of the source document. We follow Cao et al. (2018) and refer to this aspect as faithfulness in this work. Recent studies have shown that around 30% of automatically generated summaries from neural summarization systems contain unfaithful information (Cao et al., 2018; Falke et al., 2019; Kryscinski et al., 2019), especially when a sentence combines content from multiple source sentences (Lebanoff et al., 2019). Figure 1 shows a misleading and unfaithful summary demonstrating this issue.

| | |
|---|---|
| Summary | New rules have come into place that you can eat your dog. |
| Source | The restaurant began serving puppy platters after a new law was introduced allowing dogs to eat at restaurants – as long as they were outdoors! |

Table 1: A generated, unfaithful summary found in the XSUM hallucination dataset by Maynez et al. (2020).

Researchers identified multiple reasons for unfaithful summaries. One reason is the inadequacy of automatic evaluation metrics. Typical metrics like ROUGE (Lin, 2004), BLEU (Papineni et al., 2002) or METEOR (Banerjee and Lavie, 2005) are insensitive to semantic errors. These n-gram-based approaches weight all portions of the text equally, even when only a small fraction of the n-grams carry most of the semantic content. Consequently, factual inconsistencies caused by small changes are overshadowed by high n-gram overlaps. Another reason is the way abstractive models are optimized. Generating summaries that highly overlap with human references does not guarantee faithful summaries (Zhang et al., 2020b).

Initial work on metrics to automatically assess faithfulness will be discussed in Section 2 and 3, however, no consensus has been reached to date. We argue that the currently available means to automatically evaluate faithfulness do not use the full potential that current NLP methods offer. In this work, we explore new methods to assess the faithfulness of generated texts and compare them to existing approaches. We qualitatively investigate the

outputs of various methods to analyze their problems as well as to reveal ways to improve them. Finally, we test different approaches to increase faithfulness of existing summarization systems. We study the following research questions (RQs) in this work:

1. Which faithfulness metric correlates best with human judgements?
2. How can the metrics be improved?
3. How can faithfulness metrics be integrated to develop more faithful summarization systems?

Together with this work, we release an easy-to-use, open-source library[1] to evaluate faithfulness that includes all metrics discussed in this paper.

## 2 Related Work

The lack of automatic evaluation metrics for faithfulness has motivated researches to develop new metrics that ideally mimic human judgements of factual consistency. Popular approaches are based on question answering (Wang et al., 2020; Durmus et al., 2020), textual entailment (Falke et al., 2019; Maynez et al., 2020) and contextual embeddings (Kryscinski et al., 2020).

Nan et al. (2021) focus on the problem of unfaithful entities where model-generated summaries contain named entities that do not appear in the source document. The authors perform named entity recognition and calculate the percentage of entities in the summary that can be found in the source. A low percentage means entity hallucination is severe. In addition, they propose precision-target and recall-target, which capture the entity-level accuracy of the generated summary with respect to the ground truth summary.

Goodrich et al. (2019) propose to measure the factual correctness with relation extraction methods. Facts are represented as subject-predicate-object triples and faithfulness is defined as the precision between the facts extracted from the generated summary and target summary.

## 3 Methods

We re-implement popular model-based faithfulness metrics and propose multiple new methods that extract and compare different information from text to assess factual consistency.

[1] link anonymized / deleted for review

### 3.1 BERTScore

BERTScore (Zhang et al., 2020a) is an automatic evaluation metric for text generation. It utilizes contextual embeddings to compute a similarity score between every token in the candidate sentence and reference sentence. Computing the similarity with contextual embeddings is effective for matching paraphrases as well as capturing distant dependencies and ordering.

Let $x$ be a reference sentence $x = x_1, ..., x_n$ and a $y$ be candidate sentence $y = y_1, ..., y_m$ consisting of tokens $x_i$ and $y_j$, respectively. Every token in $x$ is matched to a token in $y$ to compute recall and each token in $y$ is matched to a token in $x$ to compute precision using maximum matching: each token is aligned to the most similar token in the other sentence. Three variants of BERTScore (precision, recall, F1) are shown below:

$$R_{BERT} = \frac{1}{|x|} \sum_{x_i \in x} \max_{y_j \in y} x_i^T y_j \quad (138)$$

$$P_{BERT} = \frac{1}{|y|} \sum_{y_j \in y} \max_{x_i \in x} x_i^T y_j \quad (139)$$

$$F1_{BERT} = 2\frac{P_{BERT} \times R_{BERT}}{P_{BERT} + R_{BERT}} \quad (140)$$

We use layer 8 of RoBERTa-large (Liu et al., 2019) fine-tuned on Multi-NLI (Williams et al., 2018) to compute BERTScore.

### 3.2 Textual Entailment (TE)

Textual Entailment (Dagan et al., 2005) is a popular approach to measure factual consistency employed e.g. by Falke et al. (2019), Maynez et al. (2020), Durmus et al. (2020). The basic intuition is that all information in a summary should ideally be entailed by the source document or perhaps be neutral to the source document, but the summary should never contradict it.

Let $E$ be a TE model that predicts the probability $E(a, b)$ that text $b$ is entailed by text $a$. The faithfulness score $f$ of a summary $S$ consisting of sentences $s_1, ..., s_n$ with respect to the original document $D$ with sentences $d \in D$ can be computed in 3 different ways:

$$f_{s2s}(S) = \frac{1}{n}\sum_{i=1}^{n} max_{d \in D} E(d, s_i)$$

$$f_{d2s}(S) = \frac{1}{n}\sum_{i=1}^{n} E(D, s_i)$$

$$f_{top2s}(S) = \frac{1}{n}\sum_{i=1}^{n} E(P, s_i)$$

The sentence-to-sentence (s2s) scoring method checks if every summary sentence is entailed by any source sentence. The document-to-sentence (d2s) checks if every summary sentence is entailed by the source document. The top-to-sentence (t2s) checks if every summary sentence is entailed by the $k$ most similar source sentences (calculated by comparing cosine-similarities of sentence embeddings) forming paragraph $P$.

We use BART-large (Lewis et al., 2020) and RoBERTa-large fine-tuned on the Multi-NLI dataset in our experiments.

### 3.3 Question Generation & Question Answering (QGQA)

The QGQA framework was concurrently introduced by Durmus et al. (2020) and Wang et al. (2020) and has been used in follow-up work, e.g. Maynez et al. (2020); Dong et al. (2020). The basic intuition of this framework is: if we ask questions about a summary and its source, we expect to receive similar answers if the summary is faithful. Naturally, more matched answers imply a more faithful summary as the information addressed by these questions is consistent between summary and source.

QGQA framework performs following steps to detect factual inconsistencies:

1. An answer candidate selection (AS) model selects important text spans.
2. A question generation (QG) model generates a set of question about a given generated text (e.g. the summary) using the answer candidates.
3. A question answering (QA) model answers these questions using both the source document and the generated text.
4. The faithfulness score is computed based on the similarity of the corresponding answers.

A similarity metric is necessary to compare corresponding answers. Typically F1 surface (token-level) similarity is used, which is standard for extractive QA. Other metrics are also possible, e.g. Exact Match, but we empirically find $F1$ performs best (Appendix A.1).

We use the transformers library (Wolf et al., 2020) to implement this framework. Named entities and noun phrases are extracted with spaCy[2] as answer candidates. We use T5-base[3] as QG model and RoBERTa-large fine-tuned on SQUAD2 (Rajpurkar et al., 2018) as QA model.

### 3.4 Sentence Similarity (SentSim)

The intuition of SentSim to measure faithfulness is that the information expressed in the summary should be the same as in the source document but paraphrased. Therefore, a summary sentence should be very similar to one or multiple important source sentences.

Abstractive summaries are written using different wordings and formulations to express the same information. Consequently, SentSim has to successfully deal with highly paraphrased text detecting similar concepts expressed with different words on the one hand. On the other hand, it has to differentiate between similar and contrasting or contradicting information so that it can actually be used to score faithfulness.

We propose the following strategy to asses faithfulness with sentence similarity:

1. Apply sentence splitting to the source document and summary to obtain lists of sentences.
2. Match every summary sentence with the most similar source sentence to compute precision; vice-versa to compute recall.

The precision variant (recall is analog, F1 as usual) of SentSim is defined as follows: let $S = s_1, s_2, ..., s_3$ be the set of summary sentences and let $D = d_1, d_2, ..., d_3$ be the set of document sentences, then

$$P_{SentSim} = \frac{1}{|S|}\sum_{s_j \in S} \max_{d_i \in D} sim(d_i, s_j)$$

We utilize spaCy to apply sentence splitting and experiment with various implementations of $sim()$. We empirically find that F1 or BERTScore perform well to compare and align sentences (Appendix A.1).

[2] https://spacy.io/

[3] https://github.com/fajri91/question_generation

### 3.5 Named Entity Recognition (NER)

Factual inconsistencies can occur at different levels. The entity hallucination problem occurs when a summary contains named entities that do not appear in the source document. Intuitively, a summary containing many entities that do not appear in the source is less faithful than a summary that contains the same entities as the source.

We propose the following strategy to calculate faithfulness with NER:

1. Identify named entities in summary and source document.
2. Group all found entities according to their label (e.g. ORG, PER).
3. For each named entity of the summary, calculate the most similar entity of the same group in the source document and the corresponding similarity score.
4. The faithfulness score is the average over all similarity scores.

We rely on spaCy to extract named entities and experiment with different similarity metrics to compare named entities. We empirically find F1 or cosine-similarity perform well (Appendix A.1). Please note, this approach does not capture other aspects that influence faithfulness like relations between entities or context surrounding entities.

### 3.6 Open Information Extraction (Open IE)

At relation level, we compare the relations between entities appearing in the source document and the summary. The relation hallucination problem occurs when a summary contains the same entities as the source document but their relations do not appear in the source document.

Naturally, if a summary contains many relations not present in the source document it is less faithful than a summary that contains the same relations. More matched relations imply a more faithful summary since not only the entities but also their interaction is consistent. In contrast to NER, a perfect match of summary relations with source relations can guarantee a faithful summary.

We propose the following strategy to calculate faithfulness with Open IE:

1. Apply a co-reference resolution system to replace all pronouns in the texts with their respective entity.
2. Apply an Open IE system to extract summary triples ($R(s)$) and source triples ($R(d)$) of the form (subject, relation, object) representing any fact in the given text.
3. Compute a faithfulness score based on the comparison of the extracted relations.

We use the Stanford CoreNLP toolkit for Open IE (Angeli et al., 2015), which conveniently includes an option to apply co-reference resolution as pre-processing step. We experiment with different methods to compare triples. The Relation Matching Rate (Zhu et al., 2021) operates on fact triples and basically measures the ratio of correct hits. Additionally, we linearize fact triples by concatenating the subject, relation and object and apply Exact Match, F1 and BERTScore to measure similarity. We empirically find that F1 or BERTScore work best (Appendix A.1).

### 3.7 Semantic Role Labeling (SRL)

This approach is inspired by the YiSi metric (Lo, 2019). YiSi measures similarity between two sentences by aggregating the semantic similarities of semantic structures. We argue that comparing semantic frames in contrast to comparing tokens as e.g. in BERTScore brings more linguistic structure into the faithfulness assessment. This process can find crucial differences between the argument structure of summary and source, which is a desirable property considering faithfulness. It ensures that whole summary phrases are used in semantically similar way as in the source document and should help to identify cases where the summary derives from the originally intended meaning.

We propose the following strategy to calculate faithfulness with SRL:

1. Apply a SRL model to summary and source document to obtain labeled phrases (frames).
2. Optionally, filter and merge semantic role labels to increase robustness.
3. Group phrases by their label.
4. Align ($a$) source and summary phrases with same label using a similarity metric.
5. Aggregate the similarity scores of aligned phrases and average over all labels to compute faithfulness ($f$).

Formally, this calculation can be denoted as

$$a_{recall}(l) = \frac{1}{|P_{S,l}|} \sum_{p_i \in P_{S,l}} \max_{p_j \in P_{D,l}} sim(p_i, p_j)$$

$$f_{metric} = \frac{1}{|L|} \sum_{l \in L} a_{metric}(l)$$

where $metric \in \{precision, recall, F1\}$. The precision variant of alignment ($a$) is analog to $a_{recall}$, F1 is calculated as usual. $L$ is the set of all semantic labels, $sim$ is a similarity metric comparing two texts, $P_{D,l}$ and $P_{S,l}$ are sets of phrases with label $l \in L$ for source document $D$ and summary $S$, respectively.

We use SRL BERT (Shi and Lin, 2019) trained on the English OntoNotes 5 dataset (Hovy et al., 2006) for semantic role labeling, which is available in the AllenNLP (Gardner et al., 2018) toolkit. Following Lo (2019), we merge semantic role labels into more general role types (who, did, what, whom, when, where, why, how) for more robust performance. We experiment with multiple methods to calculate similarity scores for phrases ($sim()$) and empirically find that cosine similarity of contextual sentence embeddings performs best (Appendix A.1).

## 4 RQ1: Best faithfulness metrics

We evaluate all faithfulness metrics described in Section 3 on the XSUM hallucination dataset (Maynez et al., 2020) and compute the correlation with human judgements. The dataset contains human faithfulness judgements (averaged to faithfulness scores) for 2000 document-summary pairs obtained by randomly sampling 500 articles from the XSUM (Narayan et al., 2018) test set and applying different summarization models: pointer-generator network (See et al., 2017), a transformer-based model (Vaswani et al., 2017) with randomly initialized weights, the pre-trained transformer-based model BERT (Devlin et al., 2019) and a topic-aware convolutional model (Narayan et al., 2018). Three annotators per document-summary pair were given the task to identify unfaithful text spans (hallucination spans) in the summary.

We apply a faithfulness metric on all document-summary pairs and calculate Spearman correlation (p) and Pearson correlation (r) coefficients between human judgements and predicted faithfulness scores. Results are reported in Table 2.

BERTScore achieves the highest correlation with human judgements. Entailment, SentSim and SRL perform similarly. Open IE is the last one in this ranking. It's performance might be improved by using more recent state-of-the-art models for the individual tasks.

We also evaluate all faithfulness metrics on the sentence re-ranking experiment by Falke et al. (2019). This dataset contains contains 373 triples, each triple consists of a source sentence and two summary sentences. Source sentences are taken from the CNN/DailyMail dataset and the summary sentences are generated by the summarization model from Chen and Bansal (2018). One summary sentence is faithful to the source sentence, whereas the other summary sentence is factually inconsistent.

We test how often a metric prefers the correct sentence i.e. gives a higher score to the faithful sentence. Results are shown in Table 3.

Entailment distinguishes best between unfaithful and faithful sentences, achieving 88.5% correct predictions outperforming even human performance. All other faithfulness metrics perform in a comparable range on this task, ranking about 70% example sentences correctly. The only exceptions are Open IE and NER. Both metrics perform worse than Random. We qualitatively find that, in almost every example, the entities mentioned in the summary sentences are also present in the source sentence explaining the poor ranking performance.

| Method | Pearson (r) | Spearman (p) |
|---|---|---|
| **BERTScore** | **0.501** | **0.486** |
| Entailment | 0.366 | 0.422 |
| SentSim | 0.392 | 0.389 |
| SRL | 0.393 | 0.377 |
| NER | 0.252 | 0.259 |
| QGQA | 0.228 | 0.258 |
| Open IE | 0.169 | 0.185 |

Table 2: Pearson (r) and Spearman (p) correlation coefficients for faithfulness measured between human faithfulness judgements and different automatic methods.

| Method | Correct | Delta |
|---|---|---|
| Random | 50.0% | 0 |
| NER | 29.5% | -20.5 |
| Open IE | 49.0% | -1 |
| ESIM (Falke et al., 2019) | 67.6% | +17.6 |
| SRL | 69.4% | +19.4 |
| SentSim | 69.7% | +19.7 |
| FactCC (Kryscinski et al., 2020) | 70.0% | +20 |
| QGQA | 71.9% | +21.9 |
| BERTScore | 77.5% | +27.5 |
| **Entailment** | **88.5%** | **+38.5** |
| Human (Falke et al., 2019) | 83.9% | +33.9 |

Table 3: Results on the sentence re-ranking experiment. Human performance was crowd-sourced. Ties are counted as incorrect predictions.

## 5 RQ2: Analysis of faithfulness metrics

### 5.1 Metric Comparison

The discussed faithfulness metrics compare fairly different information: tokens, entities, phrases, answers to questions etc. In this section, we test whether this is actually the case and aim to find a combination of metrics that focus on different aspects, ideally leading to a better faithfulness assessment. We correlate all faithfulness metrics with each other using the XSUM hallucination dataset. The results are shown in Figure 1.

Interestingly, BERTScore has a fair correlation with SentSim and SRL, suggesting it already has a well understanding of semantic roles and semantic similarity in general. In contrast, its correlation with QGQA is rather low, which seems reasonable given the large methodological difference of these measures. It correlates moderately with Entailment and NER indicating that entailment information and entity comparison are not entirely covered. We believe combining BERTScore, QGQA and either Entailment or NER is promising.

Data to learn a reliable combination of metrics is not available, since manual faithfulness evaluation is time-consuming and expensive. However, in a preliminary experiment, we learn a linear combination of multiple metrics with 10-fold cross-validation on the XSUM hallucination dataset. Table 4 shows combining BERTScore, Entailment and QGQA achieves an average Spearman correlation of 0.559, which is a relative improvement of 15% over BERTScore.

### 5.2 Error Analysis

Aiming to reveal weaknesses and room for improvement, we qualitatively investigated outputs of 3 metrics for 100 source-summary pairs of the XSUM hallucination dataset. We decided for QGQA as it is used by some researchers and for BERTScore and Entailment since they performed well in our previous experiments.

We identified 3 major problems of BERTScore: relations, compound nouns and numbers. Summaries and sources that share the exact same phrases but are used in different relations or contexts are problematic. Even though BERTScore relies on sophisticated contextualized embeddings, it fails to detect such factual inconsistencies since it performs token-level comparisons. This is also problematic when assessing compound nouns. Arbitrarily assembled compound nouns like "Macedonia's Prime Minister Justin Riot" are considered faithful by BERTScore. Finally, deviating numerical values are not detected by BERTScore. Whether someone achieved the second or first place in a race, or whether 5 or 5000 people were killed, makes no difference to BERTScore. This error mainly attributes to close proximity of numbers in the embedding space.

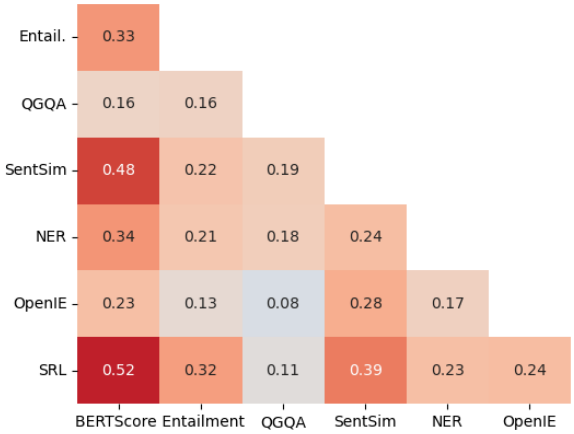


Figure 1: Spearman correlation of faithfulness metrics with each other computed on the XSUM hallucination dataset.

| Combination | Correlation |
|---|---|
| BERTScore | 0.485 |
| BERTScore + NER | 0.493 |
| BERTScore + QGQA | 0.514 |
| BERTScore + Entailment | 0.535 |
| BERTScore + Entailment + QGQA | 0.559 |

Table 4: Averaged Spearman correlations of linear metric combinations with human faithfulness judgements.

Entailment mainly suffers from one problem: unfaithful verbs. We found that verbs have the most impact on the predictions: whenever the verb is not entailed, the metric predicts very low scores. While this makes sense for entailment, it is problematic for faithfulness assessment: partly faithful sentences – like "Moscow imposed sanctions on Turkey" and "Russia suspended all sanctions against Turkey." – that only contain one incorrect verb are evaluated as unfaithful.

The main issues of QGQA are that some generated questions are nonsensical, the QA model is sometimes not able to find the correct answer and the answer comparison sometimes fails to distinguish between correct and incorrect answers. This leads to a problematic compounding of errors and, thus, improving any of its components would be beneficial.

### 5.3 Upper-bounds

In preliminary experiments, we tried to assess the impact of fixing some of the errors mentioned above. We asked human annotators to compare tokens (BERTScore), compare answers (QGQA), compare sentences (Entailment) simulating a token similarity metric, answer similarity metric and sentence similarity metric with human-like performance. As compared to their fully automatic counterpart, these human-enhanced metrics performed about 50% (BERTScore), 80% (QGQA), 75% (Entailment) better on 100 selected examples of the XSUM hallucination dataset. Thus, there is considerable headroom for metrics improvement.

## 6 RQ3: Towards faithful summarization

### 6.1 Training data filtering

We observe that many ground truth summaries of the XSUM dataset are unfaithful: We apply named entity recognition on all source-summary pairs and compute their faithfulness with BERTScore. We find that the average faithfulness is 0.85 and that about 50% summary entities[4] do not appear in the source document.

We argue that the unfaithfulness of summarization models trained on the XSUM dataset is – among other factors – a consequence of the unfaithful training data. To address this issue, we propose to filter the training data so that it consists of mostly faithful ground truth summaries.

We construct two new training data sets. The first one consists of source-summary pairs with a BERTScore higher than the average of 0.85. The second training data set consist of source-summary pairs where every named entity mentioned in the summary can be found in the source document. Table 5 lists the number of samples before and after applying the filtering.

Next, we fine-tune two current transformer-based summarization models, T5-small (Raffel et al., 2020) and BART-base (Lewis et al., 2020), on all 3 datasets to obtain 6 different summarization models. Please refer to Appendix A.3 for details about the training. We evaluate the models with ROUGE (R) to assess informativeness (ROUGE-1, ROUGE-2), fluency (ROUGE-L) and BERTScore (BS) to assess faithfulness (as BERTScore has highest correlation with human judgements). The results are listed in Table 6.

[4]We only consider the tags PER, LOC, ORG, FAC, GPE, NORP, EVENT.

| Dataset | xsum | xsum-bert | xsum-ner |
|---|---|---|---|
| # training examples | 204,045 | 146,745 | 101,844 |

Table 5: The number of training examples of the XSUM dataset and two filtered variants of it. The BERTScore variant is filtered using a threshold, while the NER variant is filtered by comparing the entities of the summary with the source document.

| Model | BS | R-1 | R-2 | R-L |
|---|---|---|---|---|
| T5-xsum | 91.70 | **34.23** | **12.05** | **26.83** |
| T5-xsum-bert | 92.49 | 33.49 | 11.43 | 26.14 |
| T5-xsum-ner | **92.54** | 32.11 | 10.49 | 25.00 |
| BART-xsum | 89.67 | **41.94** | **18.98** | **34.00** |
| BART-xsum-bert | **90.30** | 40.82 | 17.94 | 33.02 |
| BART-xsum-ner | **90.30** | 39.44 | 16.79 | 31.84 |

Table 6: Evaluation of the summarization models T5 and BART trained on different variants of the XSUM dataset. XSUM is the original dataset, XSUM-BERT is a filtered variant where every ground-truth summary has a BERTScore larger than 0.85 and XSUM-NER is a filtered variant where every entity of the ground-truth summary can be found in the source document.

T5-small and BART-base achieve a BERTScore of 91.70 and 89.67 when trained on XSUM, whereas they achieve a BERTScore of 92.5 and 90.3 when trained on a filtered version of the dataset. While these improvements are rather small, they indicate that training data has indeed influence on faithfulness. Models trained on the whole XSUM datasets achieve higher ROUGE scores since they are trained on more data. Training on a filtered dataset affects the ROUGE scores losing about 1 - 2 points. Interestingly, T5 achieves higher faithfulness scores than the state-of-the-art BART model. We qualitatively find that summaries generated by BART are more abstractive in direct comparison to T5, which naturally leaves more room to make hallucination errors.

### 6.2 Summary re-ranking

Since the faithfulness metrics are independent of the ground truth summary, we explore whether they can be used to re-rank summaries. We apply them in a post-processing step: after generating multiple summaries, we use BERTScore and Entailment to select the best.

We generate 10 candidate summaries per document with T5-small and BART-base on the XSUM test set. Next, we use both metrics to assess the faithfulness of the candidate summaries and select the one with the highest score. Table 7 shows the

| Model | Filtered? | Re-ranking? | BERTScore | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|---|---|---|
| T5-small | False | False | 91.70 | **34.23** | **12.05** | **26.83** |
| T5-small | False | Entailment | 91.68 | 34.08 | 11.92 | 26.59 |
| T5-small | False | BERTScore | 92.60 | 33.70 | 11.63 | 26.26 |
| T5-small | True | False | 92.54 | 32.11 | 10.49 | 25.00 |
| T5-small | True | BERTScore | **93.51** | 31.35 | 10.04 | 24.36 |
| BART-base | False | False | 89.67 | **41.94** | **18.98** | **34.00** |
| BART-base | False | Entailment | 89.54 | 41.48 | 18.69 | 33.68 |
| BART-base | False | BERTScore | 90.51 | 41.38 | 18.54 | 33.56 |
| BART-base | True | False | 90.32 | 39.44 | 16.79 | 31.84 |
| BART-base | True | BERTScore | **91.10** | 39.39 | 16.76 | 31.81 |

Table 7: Evaluation of the summary re-ranking approach combined with the training data filtering approach to increase faithfulness. We compare the performance of T5-small and BART-base with and without applying these techniques. The training data was filtered using the NER and the summary candidates were re-ranked with either BERTScore or Entailment faithfulness metric.

evaluation results of ROUGE for informativeness (ROUGE-1, ROUGE-2), fluency (ROUGE-L) and BERTScore for faithfulness. Re-ranking the summaries with BERTScore successfully increases the faithfulness of the summarization model. However, the model loses about 1 - 2 ROUGE points. Entailment cannot improve the factual correctness, even though it showed promising results in Section 4. While suitable for re-ranking sentences, it seems inappropriate for re-ranking summaries.

In a last experiment towards more faithful summarization systems, we combine training data filtering and summary re-ranking. We use T5-small and BART-base trained on the filtered XSUM-NER dataset to generate 10 summary candidates for every document of the XSUM test set. Next, the best candidate summary is selected using BERTScore as this metric performed well in previous experiments. Table 7 compares summarization models with and without filtering and re-ranking. Combining training data filtering and summary re-ranking can successfully improve the faithfulness increasing the faithfulness from 91.7 to 93.51 and from 89.67 to 91.1 for T5 and BART, respectively. However, ROUGE scores suffer from this losing about 2 - 3 points. We argue this trade-off is worth it as – even though ROUGE is a widely used metric – ROUGE is not as important as faithfulness since unfaithful summaries are basically useless in practice.

## 7 Conclusion

We found that BERTScore correlates well with human judgements and is able to successfully re-rank sentences and summaries. However, the error analysis revealed that BERTScore suffers from multiple problems, which can be mainly attributed to the token-by-token comparison. Analysing the correlation of all metrics with each other indicated that combining BERTScore with other metrics like QGQA, Entailment and NER could lead to an even better faithfulness assessment, as they focus on different aspects. Finally, we demonstrated that exploiting faithfulness metrics for pre- and post-processing like training data filtering and summary re-ranking can increase faithfulness without changing a model's architecture. However, to achieve more significant improvements, advanced modeling techniques are necessary, which incorporate faithfulness directly into the model.

With this work, we laid a solid basis for further development and improvement on faithfulness metrics. We also released an open-source library including all discussed metrics to encourage further experimentation and to facilitate evaluation. In further work, we aim to experiment with keyword extraction and to combine multiple existing metrics once more annotated datasets become available. Moreover, we plan to integrate faithfulness into the training process of summarization models by altering training objectives or MLE training in general. This requires faithfulness metrics with fast execution speed or metrics that can be directly included into the model, which is an interesting research direction for the future.

## References


Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D. Manning. 2015. 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. Association for Computational Linguistics.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018. Faithful to the original: Fact aware neural abstractive summarization. In *Proceedings of the 32th AAAI Conference on Artificial Intelligence*, pages 4784–4791, New Orleans, Louisiana, USA.

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.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Proceedings of the First International Conference on Machine Learning Challenges: Evaluating Predictive Uncertainty Visual Object Classification, and Recognizing Textual Entailment*, page 177–190.

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.

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.

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.

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.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia. Association for Computational Linguistics.

Ben Goodrich, Vinay Rao, Peter J. Liu, and Mohammad Saleh. 2019. Assessing the factual accuracy of generated text. In *Proceedings of the 25th International Conference on Knowledge Discovery + Data Mining*, page 166–175, New York, New York, USA.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. OntoNotes: The 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60, New York City, USA. Association for Computational Linguistics.

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

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

Logan Lebanoff, John Muchovej, Franck Dernoncourt, Doo Soon Kim, Seokhwan Kim, Walter Chang, and Fei Liu. 2019. Analyzing sentence fusion in abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 104–110, Hong Kong, China. 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.

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.

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

Chi-kiu Lo. 2019. YiSi - a unified semantic MT quality evaluation and estimation metric for languages with different levels of available resources. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 507–513, Florence, Italy. 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.

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.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

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.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. 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.

Peng Shi and Jimmy Lin. 2019. Simple bert models for relation extraction and semantic role labeling. Computation and Language repository, arXiv:1904.05255.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, page 6000–6010, Long Beach, California, USA.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. 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 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

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

*System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2019. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, pages 11328–11339, Vienna, Austria.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020a. Bertscore: Evaluating text generation with bert. In *Proceedings of the 8th International Conference on Learning Representations*, Accepted as poster. Online.

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.

Chenguang Zhu, William Hinthorn, Ruochen Xu, Qingkai Zeng, Michael Zeng, Xuedong Huang, and Meng Jiang. 2021. Enhancing factual consistency of abstractive summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 718–733, Online. Association for Computational Linguistics.

# A Appendix

## A.1 Comparing texts

Most faithfulness metrics introduced in Section 3 compare texts to compute the faithfulness score. We experiment with various similarity metrics to implement the faithfulness metrics and evaluate them on the XSUM hallucination dataset (Table 8 and the sentence re-ranking experiment (Table 9). The cosine-similarity (CS) metric is calculated on sentence embeddings generated by off-the-shelf sentence-transformers[5]. We find using Exact Match in QGQA is the best trade-off between performance and computation time. SRL performs best with CS. Depending on the task, NER performs best with either F1 or CS. Both, SentSim and Open IE perform best with either F1 or BERTScore.

| Method | Similarity | Pearson (r) | Spearman (p) |
|---|---|---|---|
| QGQA | EM | 0.200 | 0.226 |
| QGQA | F1 | 0.228 | 0.258 |
| **QGQA** | **BERTScore** | **0.252** | **0.258** |
| QGQA | CS | 0.216 | 0.222 |
| NER | EM | 0.251 | 0.255 |
| **NER** | **F1** | **0.252** | **0.259** |
| NER | BERTScore | 0.151 | 0.195 |
| NER | CS | 0.200 | 0.204 |
| SRL | EM | 0.234 | 0.273 |
| SRL | F1 | 0.359 | 0.363 |
| SRL | BERTScore | 0.270 | 0.344 |
| **SRL** | **CS** | **0.393** | **0.377** |
| SentSim | EM | -0.039 | -0.039 |
| **SentSim** | **F1** | **0.392** | **0.389** |
| SentSim | BERTScore | 0.374 | 0.372 |
| SentSim | CS | 0.387 | 0.369 |
| Open IE | EM | 0.042 | 0.076 |
| **Open IE** | **F1** | **0.169** | **0.185** |
| Open IE | BERTScore | 0.013 | 0.212 |
| Open IE | CS | 0.134 | 0.186 |

Table 8: Comparison of different similarity metrics used in various faithfulness metrics. The table lists correlations with human faithfulness judgements. We experiment with Exact Match (EM), F1 (on token-level), BERTScore and CS.

## A.2 Input for textual entailment

We evaluate different input techniques (sentence-to-sentences (s2s), document-to-sentence(d2s), top-to-sentence (top2s)) for an entailment model on the XSUM hallucination dataset and find that d2s works best as shown in Table 10.

[5]https://www.sbert.net/index.html

| Method | Similarity | Correct |
|---|---|---|
| QGQA | EM | 67.29% |
| QGQA | F1 | 68.36% |
| QGQA | BERTScore | 69.17% |
| **QGQA** | **CS** | **69.71%** |
| NER | EM | 18.50% |
| NER | F1 | 18.50% |
| NER | BERTScore | 26.54% |
| **NER** | **CS** | **29.49%** |
| SRL | EM | 50.67% |
| SRL | F1 | 66.76% |
| SRL | BERTScore | 67.83% |
| **SRL** | **CS** | **69.44%** |
| SentSim | EM | 2.95% |
| SentSim | F1 | 56.03% |
| **SentSim** | **BERTScore** | **69.71%** |
| SentSim | CS | 68.36% |
| Open IE | EM | 26.27% |
| Open IE | F1 | 46.11% |
| **Open IE** | **BERTScore** | **49.06%** |
| Open IE | CS | 47.99% |
| Open IE | RMR1 | 21.98% |
| Open IE | RMR2 | 26.27% |

Table 9: Comparison of different similarity metrics used in various faithfulness metrics evaluated on the sentence ranking experiment from Falke et al. (2019). We experiment with Exact Match (EM), F1 (on token-level), BERTScore and CS.

| Method | Pearson (r) | Spearman (p) |
|---|---|---|
| s2s | 0.152 | 0.190 |
| **d2s** | **0.366** | **0.422** |
| top2s | 0.251 | 0.302 |

Table 10: Evaluation of different input techniques for entailment models. The table lists correlations with human faithfulness judgements.

## A.3 Training data filtering

We use the transformers library (Wolf et al., 2020) to fine-tune T5-small and BART-base on 3 datasets (original XSUM dataset and 2 filtered variants). BART (Lewis et al., 2020) is a summarization model pre-trained with a masking technique very similar to the summarization task, while T5 (Raffel et al., 2020) is pre-trained for multiple tasks including summarization. We limit the number of input tokens to 512. We use the Adam optimizer with default parameters for training: no weight decay, learning rate is set to 5e-05, beta1 to 0.9, beta2 to 0.999 and epsilon to 8e-08. We use a linear learning rate scheduler, no warm-up steps. Both models were trained until convergence with a batch size of 16 examples on a single V100 GPU. This took 5 and 3 epochs for the T5-small and BART-base model, respectively. Summaries are generated using beam-search with 4 beams.