Fine-grained Factual Consistency Assessment for Abstractive Summarization Models

Sen Zhang[1], **Jianwei Niu**[1*], **Chuyuan Wei**[2*]
[1] State Key Laboratory of Virtual Reality Technology and Systems,
School of Computer Science and Engineering, Beihang University, China
[2] School of Electrical and Information Engineering,
Beijing University of Civil Engineering and Architecture, China
{zhangsen,niujianwei}@buaa.edu.cn, weichuyuan@bucea.edu.cn

Abstract

Factual inconsistencies existed in the output of abstractive summarization models with original documents are frequently presented. Fact consistency assessment requires the reasoning capability to find subtle clues to identify whether a model-generated summary is consistent with the original document. This paper proposes a fine-grained two-stage **F**act **C**onsistency assessment framework for **Sum**marization models (**SumFC**). Given a document and a summary sentence, in the first stage, SumFC selects the top-K most relevant sentences with the summary sentence from the document. In the second stage, the model performs fine-grained consistency reasoning at the sentence level, and then aggregates all sentences' consistency scores to obtain the final assessment result. We get the training data pairs by data synthesis and adopt contrastive loss of data pairs to help the model identify subtle cues. Experiment results show that SumFC has made a significant improvement over the previous state-of-the-art methods. Our experiments also indicate that SumFC distinguishes detailed differences better.

1 Introduction

The goal of summarization models is to rephrase long texts to obtain a short and fluent text containing the original text's main idea. Recently, the abstractive summarization models (Rush et al., 2015; See et al., 2017; Liu and Lapata, 2019; Zhang et al., 2020) have made significant progress and are able to generate fluent and meaningful summaries. However, there are frequent factual errors in the summaries generated by the models, which is presented in Table 1 for example. Recent studies (Cao et al., 2018; Falke et al., 2019) have shown that around 30% of the summaries generated by abstractive summarization models contain factual errors. For this reason, the practicability of abstractive summarization models is limited. Therefore, the factual consistency improvement and automatic assessment continue to be a significant challenge.

Most of the commonly used automatic evaluation metrics (Lin and Hovy, 2002; Papineni et al., 2002; Lavie and Agarwal, 2007; Zhang et al., 2019) in text generation tasks are based on the overlap of n-gram but not capable of evaluating factual errors, and manual evaluation is time-consuming and costly. Therefore, it is vital to detect subtle factual inconsistency automatically for abstractive summarization models. Different approaches have been proposed to assess the factual consistency of summarization models, including extracting and checking facts from the source document and the generated text (Goodrich et al., 2019; Wang et al., 2020; Durmus et al., 2020), borrowing off-the-shelf natural language inference (NLI) datasets for factual consistency checking (Maynez et al., 2020; Falke et al., 2019), and training pre-trained models through artificial data (Kryscinski et al., 2020; Cao et al., 2020). A common drawback of these methods is that they can just deal with some obvious factual errors and ignore the textual details, while differences in the textual details often lead to drastic semantic changes.

In this work, we propose a fact consistency assessment framework for summarization models. We split the assessment process into two stages: in the sentence selection stage, top-K pieces of evidence are selected from the original document; in the consistency checking stage, each piece of evidence is reasoned with the summary sentence in detail, then SumFC aggregates results of top-K pieces of evidence. To better distinguish the differences between the positive and synthesized negative sample pairs, the contrastive loss is introduced into the training objectives. The experiments endorse the SumFC's effectiveness over state-of-the-art methods.

[*] Corresponding author.

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 107–116
November 7–11, 2021. ©2021 Association for Computational Linguistics

Document: Jerusalem (CNN)A Palestinian teenager's name will be removed from an Israeli memorial commemorating fallen soldiers and the victims of terrorism after his family and others complained. (...) His father, Hussein Abu Khdeir, said no one asked for his permission to put his son's name on the wall. "I refuse that my son's name will be listed between soldiers of the occupation," he said. Almagor, an organization that works on behalf of victims of terror in Israel, also opposes Abu Khdeir's inclusion on the memorial. Almagor described the teen's death as a rogue attack and said he's not a terror victim. (...)
Claim: his father, hussein abu khdeir, said he's not a terror victim.

Table 1: An example of factual inconsistency claim output by abstractive summarization model. Blue text highlights the evidence in the source document, red text highlights the factual errors in the claim.

2 Related Works

Fact Consistency Evaluation Previous work on assessing factual inconsistency in abstractive summarization can be broadly classified into fact extraction and text classification approaches. The approaches based on fact extraction (Goodrich et al., 2019; Wang et al., 2020; Durmus et al., 2020) evaluate summaries by comparing the critical facts extracted from the original text and the summary. Evaluating metrics based on text classification (Kryscinski et al., 2020; Maynez et al., 2020; Cao et al., 2020) regard the fact consistency assessment as a binary classification task. Based on relation extraction, Goodrich et al. (2019) extract fact triples from both the original text and the model-generated summary, then obtains the fact consistency score based on the overlap of fact triples. Wang et al. (2020) and Durmus et al. (2020) generate questions on the summary, then adopt question answering accuracy to assess the summary's factual correctness. Besides, some researchers consider consistency assessment as the text classification task, yet there are no publicly available large-scale human-annotated datasets. Maynez et al. (2020) and Falke et al. (2019) use an out-of-the-box NLI dataset to train the model, while Kryscinski et al. (2020) and Cao et al. (2020) obtain the dataset by data synthesis. Kryscinski et al. (2020) generate the training data by a series of rule-based transformations and fine-tune a BERT model. Cao et al. (2020) adopt pre-trained sequence-to-sequence (seq2seq) model to make factual error checking and error correction.

Fact Consistency Improvement Recently, many studies are proposed to improve the consistency between the input and output of the abstractive summarization models, and most of them address this problem by assisting the summarization decoder to be fact-aware. Li et al. (2018) apply multi-task learning to introduce the text entailment knowledge into the summarization model and adopt text entailment as a reward in the decoding process, which encourages the model to generate summaries entailed by the source text. Cao et al. (2018) extract factual descriptions as relation triples from the source document, and propose to force the decoder to attend to both the source text and the extracted facts. Gunel et al. (2019) incorporate entity-level knowledge from knowledge graph to sequence-to-sequence model. Zhu et al. (2021) extract and represent facts from articles in the form of knowledge graphs, then fuse them with the representation of articles in the transformer-based decoder via attention. Dou et al. (2021) propose a general guided framework that can introduce guidance information into the seq2seq summarization model to generate more faithful summaries and enhance the degree of controllability of text generation. In another direction, some studies propose a few pluggable approaches to improve factual consistency. Falke et al. (2019) rank the candidate summaries with the consistency assessment score to improve the factuality of the summarization model. Matsumaru et al. (2020) employ a binary classifier to filter out untruthful article-headline pairs from the supervision data, then apply filtered dataset to train the summarization model. Chen et al. (2021) propose to generate candidate summaries by replacing named entities and quantities of generated summaries with that from the source article, then a discriminative model is employed to select the best candidate as the final summary.

3 Proposed Approach

3.1 Artificial Training Data

There are no manually labeled large-scale training datasets for factual consistency assessment, and obtaining datasets by human annotation is an expensive and time-consuming task. To address this problem, we follow (Kryscinski et al., 2020) to generate training datasets, which help us quickly get

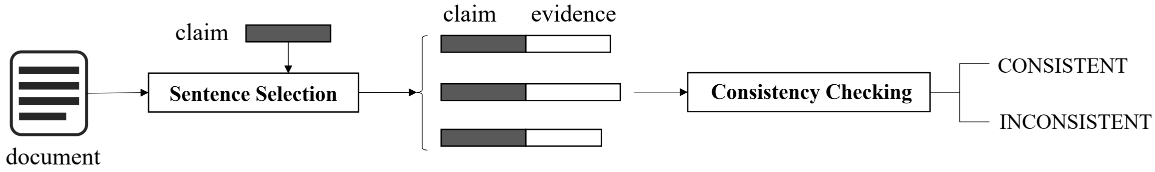

Figure 1: Our two-stage fact consistency assessment framework.

Document: (CNN)A North Pacific gray whale has earned a spot in the record books after completing the longest migration of a mammal ever recorded. The whale, named Varvara, swam nearly 14,000 miles (22,500 kilometers), according to a release from Oregon State University, whose scientists helped conduct the whale-tracking study. (...) During her 14,000-mile journey, Varvara visited "three major breeding areas for eastern gray whales," which was a surprise to Mate, who is also the director of the Marine Mammal Institute at Oregon State University. (...)
Claim: a north pacific gray whale swam nearly 14,000 miles from oregon state university.

Table 2: An example of evidence drop approach where the most similar evidence is dropped. Gray text is the discarded evidence sentence of the original document during the evidence drop operation. In this way, inconsistency examples are created, where facts in claim sentence are not appeared in the document.

a large amount of weakly supervised training data from existing summarization datasets.

In our framework, the summary is split into sentences, which are called *claims* here. In the data synthesis procedure, instead of drawing a sentence as a claim from the original text, we randomly select a sentence from the reference summary, which is viewed as a consistent claim here. Then we randomly choose a data transformation method to generate an inconsistent claim and add random noise to the consistent and inconsistent claims. The random noise is injected to simulate the noise generated by summarization models, which benefits the model to be more robust. The data transformation methods include entity swap, date swap, number swap, pronoun swap, sentence negation, and evidence drop. Details of them can be found in (Kryscinski et al., 2020), except for our proposed evidence drop approach, which is presented as follows.

Recent studies (Maynez et al., 2020) have shown that summaries generated by the summarization models contain not only intrinsic hallucinations, but also extrinsic hallucinations. Model-generated summaries with extrinsic hallucinations describe the ideas or present the facts which do not appear in the original text, and it is reported that over 90% of them are erroneous. Previous data transformation approaches can just yield examples with intrinsic hallucinations, but we try to obtain inconsistent examples with extrinsic hallucinations by discarding several sentences in the document with the highest relevance to the claim sentence.

3.2 The Proposed Assessment Framework

The recent work (Lebanoff et al., 2019) has shown that most summary sentences are only relevant to a small number of sentences in the original text. In *document-sentence* consistency checking approach, evaluation models have to locate crucial evidence and conduct consistency checking, where many irrelevant sentences consume computation and even confuse models in this process. In the proposed *sentence-sentence* assessment framework, top-K most relevant sentences (*evidence*) are selected according to the similarity of TF-IDF score, which can be considered to be explanations for the final prediction. Each evidence and claim are formed into separate sentence pairs and sent to the consistency checking model. The checking model then performs evidence reasoning on each sentence pair, learning the reasoning relationships between evidence and claim and scoring the consistency of the claim. Finally, the model calculates the importance scores of each piece of evidence to the final result, and the weighted combination of K consistency scores is calculated as the final consistency result. Additionally, we force the model to focus on textual details that affect the consistency results by introducing contrastive loss to the training objective function.

The rest of this section describes the evidence reasoning between evidence and claim (Sec 3.3), the aggregation process of all consistency scores (Sec 3.4), and the calculation of contrastive loss (Sec 3.5).

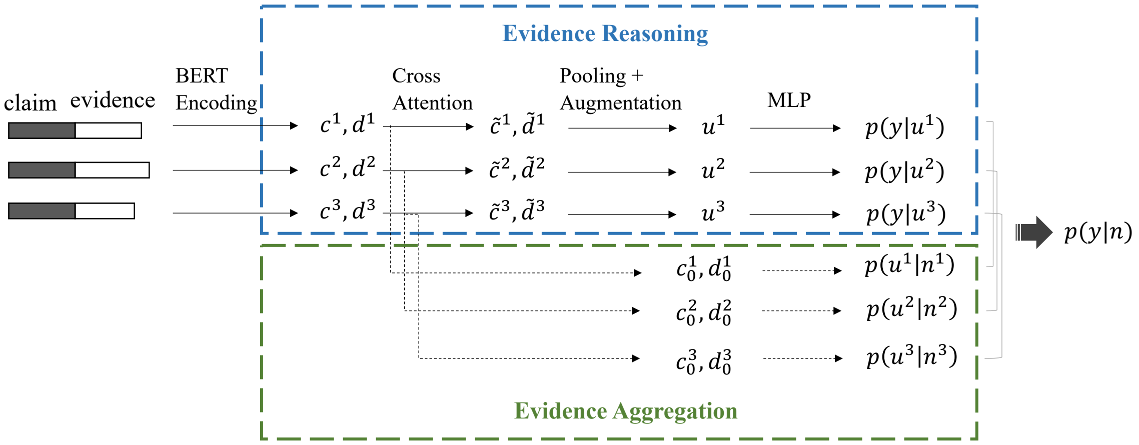

Figure 2: Illustration of consistency checking stage. In the evidence reasoning process, each sentence pair is scored by a reasoning model. Then each score is combined into the consistency score in the evidence aggregation process.

3.3 Evidence Reasoning

Evidence reasoning between evidence and claim sentence is achieved by pre-trained encoding and self-attention (Vaswani et al., 2017) across two sentences. An uncased, base BERT (Devlin et al., 2019) is used as the encoder, and each evidence-claim pair is concatenated. To represent each sentence in a sentence pair, [CLS] token is inserted at the start of each sentence. Specifically, in concatenated sentence-sentence pair n^p, the claim contains m tokens, and the evidence includes n tokens.

$$H^p = \mathrm{BERT}(n^p), \tag{1}$$

$$c^p = H^p_{0\sim m-1}, \tag{2}$$

$$d^p = H^p_{m\sim m+n-1}, \tag{3}$$

where c^p and d^p corresponds to the representation of claim and evidence respectively. Like ESIM (Chen et al., 2017), each word in c^p and d^p are matched with each other by dot product to obtain e^p_{ij}, which is adapted to compute the attention weight of each word over the other sentence. Then we could obtain the representation of claim and evidence after cross attention.

$$e^p_{ij} = c^p_i * d^p_j, \tag{4}$$

$$\tilde{c}^p_i = \sum_{j=1}^{n} \frac{\exp(e_{ij}) * d^p_j}{\sum_{k=1}^{n} \exp(e_{ik})}, \forall i \in [1, \cdots, m], \tag{5}$$

$$\tilde{d}^p_j = \sum_{i=1}^{m} \frac{\exp(e_{ij}) * c^p_i}{\sum_{k=1}^{m} \exp(e_{kj})}, \forall j \in [1, \cdots, n], \tag{6}$$

where $\tilde{c}^p_i$ is a weighted summation of d^p, which can be intuitively viewed as matching the most relevant features from d^p to represent $\tilde{c}^p_i$. It's all the same for $\tilde{d}^p_i$. To better represent the features of the sentence pair after cross-attention, we perform maximum pooling and average pooling on $\tilde{c}^p$, $\tilde{d}^p$ respectively. To compute the final sentence pair representation u^p, the difference and dot product operations are conducted to augment the difference representation between claim and evidence.

$$\tilde{c}^p_{max} = \mathrm{max\_pooling}(\tilde{c}^p), \tag{7}$$

$$\tilde{c}^p_{avg} = \mathrm{average\_pooling}(\tilde{c}^p), \tag{8}$$

$$\tilde{d}^p_{max} = \mathrm{max\_pooling}(\tilde{d}^p), \tag{9}$$

$$\tilde{d}^p_{avg} = \mathrm{average\_pooling}(\tilde{d}^p), \tag{10}$$

$$u^p = [\tilde{c}^p_{max}; \tilde{d}^p_{max}; |\tilde{c}^p_{max} - \tilde{d}^p_{max}|; \tilde{c}^p_{max} * \tilde{d}^p_{max}; \tilde{c}^p_{avg}; \tilde{d}^p_{avg}; |\tilde{c}^p_{avg} - \tilde{d}^p_{avg}|, \tilde{c}^p_{avg} * \tilde{d}^p_{avg}], \tag{11}$$

Finally, u^p is used to predict the factual consistency score of the claim for each piece of evidence.

$$p(y|u^p) = \mathrm{softmax}(\mathrm{Linear}(u^p)) \tag{12}$$

3.4 Evidence Aggregation

After obtaining the scores of the claim's correctness from each piece of evidence in the evidence reasoning process, evidence aggregation is needed to obtain the final consistency score. Here we consider 4 primary strategies for aggregating k evidence scores:

Max Choose the maximum evidence reasoning score as the final score.

Min Choose the minimum evidence reasoning score as the final score.

Avg Compute the average of the evidence reasoning scores as the final score.

Wgt Compute the weighted summation of the evidence scores as the final score. Intuitively, the higher the similarity to the claim, the higher the importance of the evidence. Due to [CLS] token insertion, here we adopt c_0^p and d_0^p to represent two sentences, respectively. Then c_0^p and d_0^p are used to compute the importance score $p(u^p|n^p)$ for each piece of evidence in the aggregation process.

$$p(u^p|n^p) = \mathrm{softmax}_p(\mathrm{Linear}(c_0^p) * \mathrm{Linear}(d_0^p)) \tag{13}$$

The final consistency score is a weighted sum of all evidence scores $p(y|u^p)$.

$$p(y|n) = \sum_{p=1}^{K} p(y|u^p) * p(u^p|n^p) \tag{14}$$

3.5 Contrastive Loss

In Section 3.1 we obtain positive and negative sample pairs by data synthesis. Consistent and inconsistent examples in a certain pair have high similarity in claims and the source document but different factual consistency labels, which leads to the difficulty of model processing. Referring to siamese networks (Chopra et al., 2005) in face recognition, we add contrastive loss into the model training objectives to force the model to learn more about the subtle differences between positive and negative samples. To begin with, we aggregate the representation of each piece of evidence in an example by the importance score of evidence to get v.

$$v = \sum_{p=1}^{K} u^p * p(u^p|n^p) \tag{15}$$

Then the distance between the representation of sample pairs is calculated to get our contrastive loss, where margin m is a set threshold.

$$\mathrm{Loss}_{contra} = \max(m - ||v_{pos} - v_{neg}||_2, 0) \tag{16}$$

Finally, cross-entropy loss and contrastive loss are combined as the model's objective function for training, and $\alpha \in [0, 1]$ is a hyper-parameter to adjust the importance of the contrastive loss.

$$\mathrm{Loss} = (1 - \alpha) * \mathrm{Loss}_{ce} + \alpha * \mathrm{Loss}_{contra} \tag{17}$$

4 Experiments

4.1 Experimental Details

Experiment Setup We implement our model using Huggingface Transformers library (Wolf et al., 2020) in PyTorch (Paszke et al., 2017), and the experiments are conducted on an NVIDIA V100 GPU with 32G memory. We keep the K=3 most relevant sentences in the sentence selection phase. The max sequence length of the sentence pairs is set to 150 during the evidence reasoning. We feed training data into the model in pairs (one consistent sample and corresponding inconsistent sample). The model is trained on generated data for four epochs with the batch size set to 28, which takes around 10 hours. AdamW optimizer with an initial learning rate 3e-5 is used for training. The weight of cross-entropy loss α is set to 0.5 during training. The proposed framework applies a weighting strategy by default unless otherwise specified.

We generate the training dataset based on the CNN/DailyMail (Nallapati et al., 2016) summarization dataset, and a total of 277,098 sample pairs are generated. The data are classified into positive (inconsistent) and negative (consistent), with each class accounting for 50%. Concerning validation and test data, we adopt the human-annotated small dataset released by Kryscinski et al. (2020). The best model checkpoints are chosen based on the validation performance. Moreover, five rounds of experiments are conducted to minimize random errors.

Baselines Several previous works attempt to train models on NLI datasets to evaluate factual consistency of model-generated summaries. We compare our model with these works, including models trained using the **MNLI** (Williams et al., 2018) dataset and the **FEVER** (Thorne et al., 2018) dataset. We also compare with recent model-based automatic evaluation metrics: **FactCC/FactCCX** (Kryscinski et al., 2020), **QAGS** (Wang et al., 2020).

4.2 Main Results

We present the experimental results in Table 3. The models using NLI datasets transfer poorly to the factual consistency assessment, and one known reason is domain shift. In contrast, methods based on the weakly supervised dataset synthesized by rule-based transformation strongly outperform the NLI dataset. And there is a significant improvement

Model	BA	F1
BERT+MNLI (Kryscinski et al., 2020)	51.51	0.0882
BERT+FEVER (Kryscinski et al., 2020)	52.07	0.0857
FactCCX (Kryscinski et al., 2020)	72.88	0.5005
FactCC (Kryscinski et al., 2020)	74.15	0.5102
SumFC (ours)	**80.41**	**0.5722**

Table 3: Fact consistency assessment results of balanced accuracy (BA) and F1 scores on the test set.

Model	% Correct
Random	50.0%
InferSent (Falke et al., 2019)	58.7%
BERT+NLI (Falke et al., 2019)	64.1%
ESIM (Falke et al., 2019)	67.6%
FactCC (Kryscinski et al., 2020)	70.0%
QAGS (Wang et al., 2020)	72.1%
SumFC (ours)	**78.7%**

Table 4: Percentage of correctly ordered sentence pairs of different assessment models on the summary sentence ranking dataset.

of SumFC over FactCC in the approaches using weakly supervised data on both balanced accuracy[1] and F1 score metrics.

To test the model's capability to discriminate textual nuances, we conduct the sentence ranking experiment published by Falke et al. (2019). In this sentence ranking experiment, each original text is paired with two summary sentences. The two summary sentences have similar expressions, yet one is a positive sample, and the other is negative. The experiment detects whether the model prefers positive samples. The results are shown in Table 4. SumFC has the best ability to distinguish the differences in detail. Compared with QAGS, SumFC improves performance by 6.6% higher.

4.3 Ablation Study

4.3.1 Effect of the Aggregation Strategy

We compare 4 different evidence aggregation strategies mentioned above and present the results in Table 5. We find that the aggregation strategy in our proposed evaluation framework has a great influence on the evaluation performance. There is a considerable difference among the results obtained by different aggregation strategies: the Max strategy and the Avg strategy obtain very poor results, while the Min strategy and the Wgt strategy perform much better.

For the poor results of the Max strategy and the Avg strategy, we find that there is some irrelevant evidence in the top-k evidence, which cannot prove whether the claim is consistent. In the evidence reasoning process, these irrelevant sentence pairs would get rather high inconsistency scores. In the Max strategy, the maximum inconsistency score of these irrelevant pieces of evidence would be selected as the final score. For the Avg strategy, these scores also overwhelm the score of the most related evidence. In contrast, while the most relevant evidence has a very low reasoning score in the Min strategy, it will be voted as the final score. And vice versa, it also gets a relatively high score. In the Wgt strategy, a weighted summation based on the importance of the evidence is made to calculate the final consistency score, thus the proposed framework could focus on the most relevant evidence.

Model	BA	F1	% Correct
Aggregation Strategy			
Max	36.88	64.55	63.8%
Min	**58.44**	79.98	78.2%
Avg	40.43	66.32	64.0%
Wgt	57.22	**80.41**	**78.7%**
Sentence Selection			
top-1	57.01	76.83	**79.5%**
top-2	57.16	77.62	77.4%
top-3	**57.22**	**80.41**	78.7%
Dataset			
FactData	54.70	75.20	70.9%
OurData	**57.22**	**80.41**	**78.7%**

Table 5: Experimental results of ablation study on our proposed framework.

4.3.2 Effect of the Sentence Selection

In the sentence selection stage, the proposed framework selects the top-K most important evidence from the original text based on the cosine similarity of TF-IDF. To explore the influence of K on performance, we adjust the size of K during the training phase to observe the performance change. Surprisingly, performance on different K was comparable. Even under the extreme condition K = 1, the proposed framework still achieves similar experimental results. This is likely due to the majority of the evidence for claims concentrates on a certain sentence, and the most important evidence can also be easily found out in the sentence selection phrase. Even if the evidence with little importance is dropped, it has a subtle impact on the evaluation process.

To better understand sentence selection operation, we conduct the evidence recall experiment

[1] Balanced accuracy is the average recall obtained on each class, which is useful when the classes are imbalanced.

Position	#Recalled
1st	352
2nd	15
3rd	1
Others	5

Table 6: The number of recalled evidence at different position on the evidence recall experiment.

to understand whether the ground truth evidence can be recalled in the sentence selection stage. We adopt the sentence ranking dataset from Falke et al. (2019) to carry out our experiment. As described above, each piece of data contains two similar summary sentences. Besides, a ground truth evidence sentence from the document is also provided. Experiments show that about 95% of the human-annotated evidence can be recalled in the top-1 position, and around 99% of the evidence can be recalled among the top-3 evidence. According to the experimental results, we find that most of the key evidence can be recalled in the sentence selection stage.

4.3.3 Effect of the Dataset

With the lack of large-scale human-annotated datasets in the fact consistency evaluation task, we obtain the dataset by data synthesis. Due to the difference in synthesis methods, models might show different performances on different artificial datasets. To explore the impact of the dataset, we compare two different datasets: the training dataset released by Kryscinski et al. (2020) (**FactData**[2]), and our artificial training dataset (**OurData**). The experimental results are presented above. We find that the datasets have a great impact on the evaluation performance. Our proposed framework performs better on OurData on all metrics. And we also notice that the sentence ranking score is quite sensitive to the dataset, and OurData greatly improves the ranking accuracy. To observe the effects of different datasets more intuitively, we draw the Precision-Recall (PR) curves of the models trained on the two datasets while taking FactCC as the baseline. It can be observed that PR curve of OurData almost completely covers the other two curves, and gets the largest area under the curve (AUC).

We hypothesize that this is due to two primary differences in the synthesis process where (i) OurData transforms human-written summary sentences to obtain training examples, instead of randomly selecting sentences from the original document then transforming it; (ii) we introduce evidence drop approaches to get training data with extrinsic hallucinations in OurData. In comparison, OurData is more difficult to classify and closer to the data in the real scene, which assists the proposed framework to performs better.

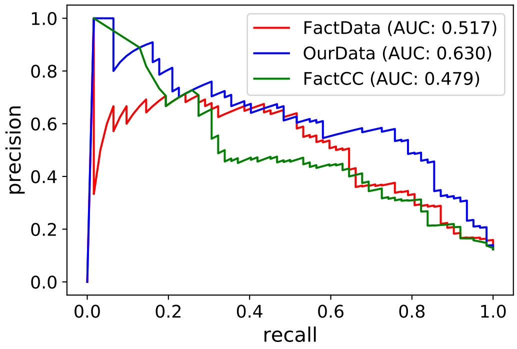

Figure 3: Precision-Recall curve on different dataset, while taking FactCC as the baseline.

4.4 Discussion

To sum up, the proposed two-stage framework and the improvement of the naturalness of the artificial dataset can improve the performance of evaluation. We can confirm this improvement from F1, balanced accuracy, the ratio of the correctly ordered sentence, and PR curves.

The one-stage models are supposed to find out the location of the evidence from the article and then conduct evidence reasoning. In the proposed two-stage framework, crucial evidence is filtered out in the sentence selection stage, which reduces the difficulty of consistency checking in the second stage. In the sentence selection stage, we employ a relatively simple metric of sentence similarity, and the results of the sentence-recall experiment confirm its effectiveness under the current experiment. On more challenging tasks, more sophisticated semantic similarity metrics could be adopted to achieve better performance, such as BERT-based approaches.

5 Case Study

Several typical examples of the model outputs are listed in Table 7. We find that the sentence selection does help to find out the crucial evidence, which also provides explanations for the model outputs. When the claim partially represents a certain

[2] https://github.com/salesforce/factCC

Document #1: (CNN)Georgia Southern University was in mourning Thursday after five nursing students were killed the day before in a multivehicle wreck near Savannah. (...) University President Brooks A. Keel said in a statement. "The loss of any student, especially in a tragic way, is particularly painful. Losing five students is almost incomprehensible." Georgia Southern flew flags at half-staff and counseling was offered to students. (...) **Claim:** georgia southern university was in mourning after five nursing students died. **FactCC:** consistent **SumFC:** consistent **Human Annotation:** consistent
Document #2: (...) Prosecutors have listed, as they must, the aggravating circumstances that make this horrific mass murderer deserve the harshest punishment. The killing was "heinous, cruel and depraved." He placed a bomb in a crowd, set it to kill and maim children and adults indiscriminately – if that's not heinous, cruel and depraved, what is? Cruelty classically consists of a desire to cause pain and suffering in innocent victims, or, at the opposite extreme, it reflects a cold, callous indifference. (...) **Claim:** the bombing was "heinous, cruel and depraved". **FactCC:** inconsistent **SumFC:** consistent **Human Annotation:** consistent
Document #3: (CNN)Larry Johnson remembers the fear and feeling of helplessness from being on the SkyWest Airlines flight that made an emergency landing in Buffalo, New York. " (...) Minutes later, Johnson says, the attendant announced there was a pressurization problem and told passengers to prepare for the emergency landing. (...) It later issued a statement that did not reference any pressurization issues. (...) The spokeswoman said that maintenance personnel found no indication of a pressurization problem with the aircraft, an Embraer E170, and that the airline continues to investigate the cause. (...) **Claim:** the airline says it's investigating the cause of a pressurization problems. **FactCC:** consistent **SumFC:** consistent **Human Annotation:** inconsistent

Table 7: Comparison of evaluation outputs of FactCC and SumFC on test dataset. There are factual errors in some claims generated by abstractive models. Top-2 evidence selected by the proposed framework has been highlighted.

sentence from the source text in case 1, the evaluation models can easily output the correct prediction. In case 2, the claim replaces the killing with the bombing, and FactCC makes a wrong judgment. Moreover, FactCC and SumFC both fail in case 3. The claim and the source text both mention the pressurization problem and the summary claims it is investigating the cause of the pressurization problem. According to the source text, it is clear that the aircraft has no pressurization problem, and they continue to investigate the cause of the emergency landing.

In conclusion, FactCC can handle some simple cases, while SumFC is further able to handle cases that requires simple reasoning. For cases that require more complicated reasoning skills or commonsense knowledge, both models do not work very well.

6 Conclusions

In this paper, we propose a clean and intuitive factual assessment framework that splits the assessment process into two stages, including the sentence-selection stage and the consistency-checking stage. We demonstrate the proposed fine-grained approach leads to more accurate factual assessment and outperforms the state-of-the-art methods by a large margin. We have shown that the evidence extracted in the sentence selection step can also provide explanations for the evaluation process. And we also have a comparative case analysis on our proposed framework and recent models, and point out the shortcomings of current assessment approaches.

In the future, we would like to explore the interaction and inference between source sentences, build a more authentic training dataset, or incorporate common sense knowledge into text generation assessment. Moreover, it is also interesting to explore whether the proposed framework can be developed into a general automatic evaluation framework for text generation.

Acknowledgments

This work was supported by National Natural Science Foundation of China (61772060). And we would like to thank anonymous reviewers for their valuable comments and helpful advice.

References

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

Ziqiang Cao, Furu Wei, Wenjie Li, and Sujian Li. 2018. Faithful to the original: Fact aware neural abstractive summarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1657–1668, Vancouver, Canada. Association for Computational Linguistics.

Sihao Chen, Fan Zhang, Kazoo Sone, and Dan Roth. 2021. Improving Faithfulness in Abstractive Summarization with Contrast Candidate Generation and Selection. In *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Sumit Chopra, Raia Hadsell, and Yann LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE.

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.

Zi-Yi Dou, Pengfei Liu, Hiroaki Hayashi, Zhengbao Jiang, and Graham Neubig. 2021. Gsum: A general framework for guided neural abstractive summarization. In *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

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.

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

Beliz Gunel, Chenguang Zhu, Michael Zeng, and Xuedong Huang. 2019. Mind the facts: Knowledge-boosted coherent abstractive text summarization. In *Proceedings of The Workshop on Knowledge Representation and Reasoning Meets Machine Learning in NIPS 2019*, volume 32.

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.

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

Logan Lebanoff, Kaiqiang Song, Franck Dernoncourt, Doo Soon Kim, Seokhwan Kim, Walter Chang, and Fei Liu. 2019. Scoring sentence singletons and pairs for abstractive summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2175–2189, Florence, Italy. Association for Computational Linguistics.

Haoran Li, Junnan Zhu, Jiajun Zhang, and Chengqing Zong. 2018. Ensure the correctness of the summary: Incorporate entailment knowledge into abstractive sentence summarization. In *Proceedings of the 27th*

International Conference on Computational Linguistics, pages 1430–1441, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Chin-Yew Lin and Eduard Hovy. 2002. Manual and automatic evaluation of summaries. In *Proceedings of the ACL-02 Workshop on Automatic Summarization*, pages 45–51, Phildadelphia, Pennsylvania, USA. Association for Computational Linguistics.

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

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

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gülçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

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.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch. In *NIPS 2017 Workshop on Autodiff*.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

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.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

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, Rémi 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 M. 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 Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Chenguang Zhu, William Hinthorn, Ruochen Xu, Qingkai Zeng, Michael Zeng, Xuedong Huang, and Meng Jiang. 2021. Enhancing factual consistency of abstractive summarization. In *North American Chapter of the Association for Computational Linguistics (NAACL) 2021*.