



Improving Neural Abstractive Summarization with Reliable Sentence Sampling

Daniil Chernyshev^(✉) and Boris Dobrov

Lomonosov Moscow State University, Moscow, Russia
chdanorbis@yandex.ru

Abstract. State-of-the-art abstractive summarization models are able to produce summaries for various types of sources with quality comparable to human written texts. However, despite the fluency, the generated summaries are often erroneous due to factual inconsistencies caused by neural hallucinations. In this work, we study possible ways of reducing the hallucination rate during abstractive summarization. We compare three different techniques aimed at improving the correctness of the training procedure: control tokens, truncated loss, and dataset cleaning. To control hallucination rate outside of the training, we propose an improved algorithm for summary sampling - reliable sentence sampling. The algorithm utilizes fact precision metrics to sample the most reliable sentences for an abstractive summary. By conducting the human evaluation, we demonstrate the algorithm's efficiency in preserving summary factual consistency.

Keywords: Abstractive summarization · Hallucinations · Controlled generation

1 Introduction

Abstractive summarization is a form of document summarization where the summary is obtained by extracting salient parts and paraphrasing at the same time. Modern neural summarization models are able to create a summary of different levels of abstraction while achieving a high level of fluency and coherence. However, according to Maynez et al. [17], higher abstraction levels lead to increased information distortion in summary caused by text objects known as neural hallucinations.

Neural hallucinations are an integral part of text generation models which gives them the ability to create text objects not found in input data. Due to the focus of most previous works on extractive summarization datasets, the issue of neural hallucination correctness was poorly studied [13]. By studying the behavior of the existing models on one-sentence abstractive summarization dataset Xsum [18], Maynez et al. [17] found that up to 70% of generated summaries contain neural hallucinations and over 90% of hallucinated statements lead to factual inconsistency. Such a high level of factual hallucination raises serious concern about the reliability of existing methods and postpones their widespread practical application.

This paper explores possible ways of reducing hallucinations that may disrupt the fact set. To distinguish acceptable context synonyms from actual factual mistakes, we develop a new dataset for Russian news summarization¹ that exploits the concept of news clusters. We compare three existing methods that reduce the rate of factually incorrect hallucinations and study their effect on abstractive summarization model performance. Dataset cleaning proves to be the most efficient of all, increasing target replication precision up to 19% while also improving copying of named entities.

To improve factual consistency during model inference, we propose a new method for summary sampling - reliable sentence sampling (RSS). In contrast to previous works, our method controls the summary generation process on sentence-level instead of just selecting the most correct variant of a complete summary. We propose new metrics to measure summary reliability and demonstrate the efficiency of the new sampling algorithm in reducing the amount of generally incorrect facts. To study correlation with human judgment, we test our most successful version of RSS against human experts.

2 Related Work

2.1 Abstractive Summarization

One of the first works to apply the sequence-to-sequence model to abstractive summarization task was the work of Rush et al. [20] where an attention-based model for headline generation was proposed. Later See et al. [22] adapted a copy-attention mechanism to solve the out-of-vocabulary issue as well as improve model attention to essential text details. With the introduction of pre-trained language models with global context knowledge like BERT [3], the copy mechanism was no longer necessary for the extraction of important fragments. Liu et al. [16] proposed BERTSum model which adapts BERT encoder and uses a simple multilayer Transformer [23] decoder. Despite BERTSum surpassing copy-attention model scores, the full potential of language models was not utilized due to the lack of text generation tasks in the pretraining phase. Lewis et al. [15] solved this issue by employing denoising pretraining in BART language model which allowed it to be fine-tuned directly to various text generation tasks, including abstractive summarization.

2.2 Hallucinations in Summarization

With the first sequence-to-sequence summarization models, it became evident that generated summaries contained more sophisticated errors than just not following target vocabulary. Cao et al. [1] discovered that almost 30% of generated headlines contained factual mistakes. That claim was later supported in the work of Kryscinski et al. [13] where it was shown that models trained on more complex summarization datasets, such as CNN/Daily Mail, had a similar ratio of factually inconsistent summaries.

¹ <https://github.com/dciresearch/RNC-dataset>.

Several hypotheses of factual inconsistency causes in abstractive summarization models were proposed. Kryscinski et al. [13] argued that the problem lies in the lack of constraints in the summarization task and the existence of incorrect examples in popular for the task datasets that were automatically scraped without validation. The effect of noise in the training dataset is demonstrated in the work of Dusek et al. [4] where authors managed to reduce the ratio of inconsistent text by cleaning noisy examples. Maynez et al. [17] agreed with the lack of constraints in the summarization task, however, it was also pointed out that the problem of factual inconsistency is natural for the text generation models due to their main creation tool - hallucinations.

A neural hallucination is an object generated by the model that is not directly related to the input data. Text generation models, such as abstractive summarization models, use hallucinations to paraphrase text. Maynez et al. [17] showed that during abstractive summarization hallucination rate may reach 70% and in the majority of cases generated hallucinations contain extrinsic information which cannot be validated with the source text. Their analysis of extrinsic hallucinations concluded that in 90% of cases new information was factually incorrect.

2.3 Methods of Improving Factual Consistency

Goodrich et al. [7] proposed using a fact set precision metric to measure the quality of generated summary in terms of fact consistency. Falke et al. [5] used a similar fact precision idea to choose the most factually correct variant among generated summaries, however, instead of extracting facts directly, the authors adopted NLI models to measure correctness sentence-wise. Kryscinski et al. [14] improved this approach by proposing a special model which measures fact consistency between a full source document and generated summaries by classifying each token. Later Zhou et al. [25] expanded the classification approach to detect hallucinations directly by fine-tuning pre-trained language models to distinguish distorted data. A more natural approach to measuring consistency was proposed by Wang et al. [24] where, instead of checking text inference, the correctness is determined by comparing answers found in the source and the summary for the set of questions generated for the summary.

Besides reranking techniques, there are methods that embed knowledge graphs into model attention, append facts to model input, unify entailment checking and text generation models, post-edit final summaries to correct errors, or directly optimize fact precision metrics during training using reinforcement learning [11].

3 Obtaining Dataset for Russian News Summarization

In this work, we focus on studying the possible ways of reducing hallucinations in Russian abstractive summarization models. By the date of writing, only one Russian summarization dataset was proposed - Gazeta [9]. This dataset contains 63435 examples with either extractive or abstractive summaries.

However, our inspection of the dataset revealed that not all text-summary pairs are correct for the summarization task. The example of bad pair is shown

in Table 1. The article is about the details of a show hosted at Kremlin, while the summary explains the possible reasons for the event as well as gives some national facts. This example would promote frequent hallucinations of the abstractive summarization model since most fragments of the summary cannot be extracted from the text directly and, thus, can only be guessed.

Table 1. Translated example of bad article-summary pair from test partition of Gazeta dataset. Red - parts that cannot be extracted or derived from the source text.

Article
About 11 thousand spectators saw all the best that is in the culture of Buryatia today. The Buryat State Academic Opera and Ballet Theater, the National Circus, the Buryat National Song and Dance Theater “Baikal” performed in the Kremlin, which became the winner of the show “Everybody Dance!” on the TV channel “Russia”, as well as other professional and amateur groups of the region. More than 300 artists from one region on the main stage of the country - it looks like a record for Russia
Summary
On February 25 and 26, Sagaalgan, the Eastern New Year, was celebrated in the Kremlin Palace of Congresses. Buryatia is the center of Russian Buddhism and one of the few regions of the country where the New Year is officially celebrated twice.
Link: https://www.gazeta.ru/social/2020/02/28/12980611.shtml

The existence of this kind of example can be explained by the nature of the method used to obtain the data which was proposed with the Newsroom dataset [8]. The method automatically builds a summary for a news article by exploring description HTML metatags. The problem with this approach is that there is no guarantee that the description metatag would contain an actual summary.

In Russian news writing, it is rare to find an exclusive human-written summaries due to a widespread scheme of article composition know as the Inverted pyramid. In this scheme, the information is presented according to relevance to the reader so that they would need to read only the first few sentences to understand the concept. This way the first paragraph composed of 3–4 sentences (or article lead) can act as a summary for the whole text. But the scheme does not guarantee the repetition of previously presented information. Thus, excluding article lead from the main body and using it as a summary and the remainder of the article as the source text may not be possible due to lack of contextual connection.

The description HTML metatags are commonly filled with the news leads in case the author of the news article decides not to fill them. This can be easily detected by calculating ROUGE-L precision between description and article which would yield a result indicating an almost complete text overlap. However, it is possible that meta-tags were filled incorrectly and the article lead was separated from the main body. In this case, the ROUGE metric may yield results equivalent to an extremely paraphrased summary which, on the other hand, are valuable for abstractive summarization tasks. The same problem occurs if the

author did not follow the news article writing guidelines and decided to use description metatag to draw the reader’s attention with irrelevant details.

In Gazeta dataset, ROUGE score thresholds were used to filter the described cases of bad description tags, but it is evident that it was not sufficient for author intended mistakes. Thus, a more advanced filtering procedure is required to further refine this dataset for our research.

Another problem is that the task performance of the neural network depends on a number of examples presented in the training dataset and state-of-the-art models utilize datasets with more than 200k examples. This implies that, even after cleaning the Gazeta dataset, it would be required to expand it with more examples to achieve results similar to results on English counterparts. The issue could be alleviated with multilingual MLSUM [21] dataset that contains article-summary pairs from another Russian publisher “Moskovskij Komsomols”. However, the total count of relevant pairs is less than 27k while the low-quality pair filtering procedure did not employ context analysis which means that the dataset carries the same flaws.

Table 2. Dataset Statistics.

	Six publisher dataset	Russian news [2]	CNN [10]	Daily mail [10]	NY times
Mean article length (words)	201.06	220.0	760.50	653.33	800.04
Mean summary length (words)	41.16	52.6	45.70	54.65	45.54
Lead-3 ROUGE-1	34.93	34.69	29.15	40.68	31.85
Lead-3 ROUGE-2	16.84	12.21	11.13	18.36	15.86
Lead-3 ROUGE-L	27.90	27.69	25.95	37.25	23.75

Alternatively, we used an altered Newsroom approach. Instead of following unreliable HTML metatags, we utilize two facts: article lead can act as a summary, and the same event may be covered by different news publishers. By using articles within one news cluster, we construct pseudo-summary-source pairs: lead of the article from one publisher as the summary and full article as the source from another. The eligibility of this approach was demonstrated in previous work [2].

We extracted news articles from six publishers for the past 4 years: “Газета.ру” (Gazeta), “РИА” (RIA), “Интерфакс” (Interfax), “Коммерсантъ” (Kommersant), “Дождь” (TV Rain), “Lenta.ru”. To obtain news clusters, we used DBSCAN and multilingual paraphrase xlm-r model [19] to build text embeddings for article headlines and leads. As cluster centroids and summary sources, we chose articles from “Коммерсантъ” since this publisher focuses on analytics and it was expected to have the most informative leads. We filtered out all summary-source pairs which had either $\text{ROUGE-1}_{precision} < 30$ or $\text{ROUGE-L} > 80$ to exclude examples with low relevance or high text overlap.

The resulting dataset contains 206 487 pairs. The data is divided into training (80%), development (10%), and test (10%). The statistics are presented in

Table 2. As it can be seen, the ROUGE scores for the Lead-3 solution for the new dataset are better than previously proposed “Lenta-Коммерсантъ” dataset [2] and bear more similarity to existing English counterparts. However, the difference in ROUGE-L scores is negligible which suggests the same ratio of completely extractive fragments in summaries and, thus, the same level of abstractness.

4 Improving Quality During Training

Since abstractive summarization models are prone to hallucinations, it is important that the training procedure won’t promote this type of error. Maynez et al. [1] demonstrated that the common training setting used in the field is insufficient and requires additional constraints. In this section, we describe three existing approaches to reducing hallucinations during training.

4.1 Control Tokens

In sequence-to-sequence models, the previous token in the first step of the decoding process is usually set as a BOS token (beginning of the sentence). However, instead of using universal BOS for all sentences, we can set multiple variants that would condition token probability distribution.

In the method proposed by Fillipova [6] for the data-to-text task, these variants of BOS token are set to represent desired hallucination level of generated text. For the training set they are determined according to formulae:

$$\text{BOS}(G, S) = \text{BOS}_{\text{hal}(G, S)} \quad (1)$$

$$\text{hal}(G, S) = \lfloor (1 - \text{ROUGE-1}_{\text{precision}}(G, S)) \cdot 5 \rfloor \quad (2)$$

where G - target document (gold), S - source document, $\text{BOS}(G, S)$ - function that determines the initial decoding token for text-summary pair. This way the model will learn to retain source document language at BOS_0 and, on the contrary, at BOS_5 generate text with the most diverse vocabulary.

4.2 Truncated Loss

At the late stages of the model training, the examples that meet requirements for multiple learned data patterns may alter established rules. If those examples are correct, this may result in overall performance improvement. However, it is very likely that the most divergent cases of those examples are incorrect for the task and do not follow necessary constraints. In abstractive summarization, such an example would be a target summary that contains external information that cannot be derived from the source text only. This kind of example would promote additional hallucinations since unrelated parts could only be guessed using accumulated knowledge.

To tackle the issue of examples that promote hallucinations at late stages of training, Kang et al. [12] proposed a truncated loss. The idea behind this method is to collect loss statistics and determine normal values for loss function.

First, the loss history of size k is accumulated. Then example truncation threshold C is calculated by taking $1 - c$ quantile over the distribution of losses. After initializing the threshold training loss takes the following form:

$$\hat{L}(H, X) = L(H, X) - \sum_{l(H, x) > C} l(H, x) \quad (3)$$

$$L(H, X) = \sum_{x \in X} l(H, x) \quad (4)$$

where H - model, X - batch, $l(H, x)$ - pair-wise loss function for examples $x \in X$. Error history is reset after each threshold calculation and the threshold is recalculated every k steps. Thus, all examples with loss higher than truncation threshold value C are ignored.

4.3 Dataset Cleaning

As it was mentioned before, it is possible that some examples in the training set may promote model-specific errors such as hallucinations that break factual consistency. The straightforward solution is to remove these examples or fix parts that are incorrect in terms of task constraints.

For our dataset, we found that the BERTSum model struggled to reproduce multi-sentence citations due to preprocessing trait, that forces to split them with sentence separator token (e.g. SEP). These long citations are indistinguishable from ordinary text fragments from the model viewpoint and may be paraphrased and combined with any text part, which may create factual inconsistency. We also found that some summaries contained additional named entities that are considered obvious for the reader but cannot be extracted directly from the source text. For example, mentioned in the article “coronavirus” is expanded as “coronavirus COVID-19” in the summary, which is incorrect in terms of task constraints if the source article does not contain the named entity “COVID-19” since there are other types of “coronavirus”.

To improve model training performance, we remove the following summary fragments:

- Citations that require inserting SEP token (multi-sentence citations)
- Named entity expansions that are not supported by source article.

5 Reliable Sentence Sampling

The existing approaches for summary generation may use different techniques to improve overall text consistency. However, their efficiency is limited by underlying sampling algorithms that have high computation cost per explored text variant. This happens to be a crucial issue for filtering methods based on factual consistency metrics that require reviewing a large number of possible variants. We propose a general improvement to the sampling procedure that allows to apply these methods more frequently as well as reduce the required number of alternatives at each sampling step.

5.1 Summary Sampling Issues

Neural abstractive summarization models use sampling algorithms to generate text from predicted conditional probability distribution. Wide-spread beam search sampling tracks k most probable hypotheses at each step. However, these hypotheses may not have any significant difference and be synonymous with any in the set.

By increasing beam width k we also increase diversity in a set of hypotheses at the cost of increased memory consumption and slower step iteration. Falke et al. [13] exploit this idea to improve the factual consistency of generated summaries by reranking a large number of final hypotheses. The main flaw of this approach is that the number of possible hypotheses increases exponentially with each beam search step and by the end of summary sampling a large fraction of factually correct hypotheses would be filtered out due to lower support from learned training examples.

This problem cannot be solved with higher k values due to hardware limitations. Applying the hypotheses reranking technique at every sampling step is not an alternative either as it has a very high risk of collapsing probability space to a variant unexplored during model training which may lead to incoherent text.

On the other hand, to properly evaluate text factual correctness, it must consist of finished ideas since the fact set may change with each added word. To alleviate the issue, we exploit the fact that text is naturally structured into finished ideas with sentences.

Algorithm 1. Reliable sentence sampling. Input: k – beam width, source – source text, metric – reliability metric function. Output: summary text

```

function RSS( $k$ , source, metric):
    summary, bestState ← SamplerInit()
    while summary[-1] ≠ EOS:
        cands, result = RSS_step( $k$ , source, summary[-1], bestState)
        if length(cands) = 1 and cands[0] = result:
            return summary.concat(result)
        bestHyp, bestState ← selectByRelScore(cands, metric, source)
    summary.concat(bestHyp)

function RSS_step( $k$ , source, BOS_token, state):
    hyp, states ← BeamSearchInit( $k$ , BOS_token, state)
    cands ← []
    while hyp[0][-1] ≠ EOS:
        scores, states ← DecoderStep(hyp, states)
        hyp ← Top_K(hyp, scores)
        for h in hyp:
            if h[-1] = SEP:
                cands.append((h, getState(h, states)))
    final_hyp ← hyp[0]
    return cands, final_hyp

```

5.2 Algorithm

Instead of building a summary token-to-token, we generate text sentence-to-sentence. The idea is to generate multiple sentences-candidates by saving each complete sentence found in the probable hypotheses set and then expand the most appropriate by resuming generation from the respective decoding state.

Our implementation is the following (Algorithm 1). Denote sentence separator token as SEN. At each step of beam search for each hypothesis in top-k set, save token sequence and decoder state if the last token in the hypothesis is SEN. Repeat beam search iteration until top-1 hypothesis' last token is not an end-of-text token. This hypothesis is saved in case there are no alternatives in saved sequences (sentences). If the list of saved sequences contains less than two sequences and the only element is equal to the top-1 hypothesis of the final beam search step, then add the top-1 hypothesis to the accumulated summary and return it as the resulting summary. Else choose from the list sentence with the highest reliability score and add it to the summary. Restart decoding procedure considering the last token of summary and recorded decoder state for last added sentence.

5.3 Reliability Scores

There are two types of information unreliableness: local and global. In terms of abstractive summarization, local unreliableness can be any fact introduced in a summary that is not supported directly by the source text. However, these facts may be obvious for the reader and, thus, do not break the overall reliability of the text. For example, in news articles, it is common to omit the year in rare events like elections or championships so this information cannot be directly inferred from the text but would not be a factual mistake if encountered in summary.

Global unreliableness on the other hand is always an incorrect fact regardless of the information source. The problem is that proving that fact is globally unreliable requires processing a full set of relevant information sources.

Thus, it may not be possible to separate local fact unreliableness from global, but by filtering out locally unreliable facts we can ensure global reliability of a summary and improve quality.

Local unreliableness can be measured as fact set difference as was proposed by Goodrich et al. [7]. In our work, we consider facts related to named entities or main events. Using this set, we compute two metrics: named entity overlap (NEO) and named entity fact overlap (NEFO). The formulae are following:

$$\text{NEO}(A, S) = \frac{|\text{NE}(A) \cap \text{NE}(S)|}{|\text{NE}(A)|} \quad (5)$$

$$\text{NEFO}(A, S) = \frac{|\text{NE-facts}(A) \cap \text{NE-facts}(S)|}{|\text{NE-facts}(A)|} \quad (6)$$

where A - generated summary, S - source text, $\text{NE}(X)$ - named entities, $\text{NE-facts}(X)$ - named entity facts in form of subject-predicate-object.

6 Implementation

For the base of our experiments, we have chosen BERTSumAbs [16] abstractive summarization model. Since we study ways of reducing hallucination rate, the results of our research would extend to any other sequence-to-sequence neural abstractive summarization model. All settings are the same as in the original model, except we use a special version of BERT pretrained on Russian news language [2] and train model for 100k steps.

For truncated loss, we used a history of $k = 10000$ elements and quantile level $1 - c = 0.7$. For named entity detection and relation extraction, we used Space-ru model for spaCy library. Two types of facts were extracted:

- Object<-Action->Subject - extracts details about the main event
- Named_entity(Object) and Object->Modifier – extracts all modifiers related to named entities.

All words in fact triplets were lemmatized to take into account possible paraphrases.

7 Experiments

7.1 Training-Based Methods

Considered methods do not alter the sampling algorithm or interfere with a fact set directly, so ROUGE-1 precision and NEO would be enough to describe changes in terms of text quality.

Table 3. Evaluation results of training-based methods.

Method	ROUGE-1 precision	NEO
Base model	41.24	58.01
Truncated loss	45.22	58.88
Control tokens	43.22	57.16
Dataset cleaning	49.16	61.65
Dataset cleaning + Truncated loss	48.34	61.05

The evaluation results are presented in Table 3. As it can be seen, the most efficient method is dataset cleaning. It can be explained by the duality of this method effect: it eases the task by removing parts of the text that are hard to replicate and it reduces the average summary length, thus, lowering the overall amount of compared tokens. Interestingly, loss truncation lowers the quality after dataset cleaning despite being the second most efficient method. This indicates that removed long citations were one of the sources of examples that promote hallucinations.

The inferior performance of control tokens may be explained by a class imbalance of hallucination levels: maximal level on the dataset is 3 (a result of ROUGE

filtering during dataset building) and the fraction of examples with this hallucination level is 42% while the fraction of examples with level 1 and lower is 11% in total. The model just did not accumulate enough experience to generate text at the lowest hallucination levels which, according to NEO, did lead to inconsistencies in named entity replication.

7.2 Reliable Sentence Sampling

Since our method filters out parts of the generated text and alters the resulting fact set, it is important to additionally compare ROUGE-1 recall with target summary as well as compare NEFO with the source text.

Table 4. Reliable sentence sampling (RSS) results on the test set.

Method	ROUGE-1 PRECISION	ROUGE-1 recall	NEO	NEFO	Cluster NEO	Cluster NEFO
Beam search ($k = 5$)	49.16	42.34	61.65	39.57	65.82	44.41
RSS NEO	38.72	50.16	70.77	46.78	76.25	52.13
RSS NEFO	41.78	49.04	59.89	50.42	65.93	62.74
Lead-3	33.12	50.54	—	—	—	—

Table 4 shows the results of the evaluation. As expected, both variants of reliable sentence sampling (RSS) improve their respective metrics. However, in both cases, ROUGE-1 precision is lower than in summaries obtained by the original beam search algorithm while recall is higher. Comparing with simple extraction of the first 3 sentences of the article (Lead-3), we can see a similar pattern which suggests that RSS may promote more extractive behavior of text generation procedure. This indirectly means that the overall rate of hallucination is also reduced. Another interesting fact is that RSS with NEFO lowers the average NEO. This may be explained by contextual synonyms since the metric does not consider them.

To further study the effect of unreliableness reduction, we compared resulting summaries with concatenated source-articles from the same news clusters (Table 4, Cluster metrics). Combining articles within one news cluster allows us to approximate the global fact set for the event and, thus, measure the probability of encountering globally unreliable facts.

As it can be seen, all metrics have improved. The difference between original beam search and RSS with NEFO in terms of NEO has been eliminated at the cluster level. This confirms the hypothesis of the effect of contextual synonyms and suggests that the difference of 1–2% should be considered negligible. NEFO shows the largest improvement of 12% at cluster level, but its maximal value is still lower than NEO. This can be explained by strict constraints imposed on fact triplets where the difference in any component is an error.

7.3 Human Evaluation

To evaluate the efficiency of our method, we use new metrics that have not been studied thoroughly. Specifically, we do not have any information about normal values and correlation with human judgment.

To address the issue, we decided to use a crowd-sourcing platform Yandex.Toloka to evaluate the efficiency of reliable sentence sampling with NEFO metric. To build tasks we choose 100 generated summaries from the test set that had more than 3 alternative sentences for the first two steps of RSS algorithm and had ROUGE-2 score higher than 15% (to ensure summary coherence).

For each summary, we built a task in the following form: source text and 3 sentences-alternatives found by RSS algorithm. The task presented to the human judge was to choose the most factually correct and most detailed (in case variants seemed to have an equal level of correctness) variant. We specifically instruct people to consider as a factual mistake only information that is not supported by or even contradicts the source text and use all other mistakes only to break ties. The “most detailed” requirement was determined during our preliminary evaluations where it was found that variants that were universally correct for any article of a similar topic were preferred by low-quality human judges as they fulfilled the factual correctness constraint but did not require reading the source text.

Sentences supplied to human judges had one chosen by RSS, one that was included by the original beam search algorithm in the final summary, and one random alternative among top-5 sentences in terms of NEFO metric, but with a different set of factual mistakes. In other words, we asked human judges to use their knowledge to perform a factual correctness ranking procedure in RSS algorithm.

In total, 200 tasks were given (100 texts with 2 sets of sentences each). Each judge was asked to complete 5 tasks and each task was completed by 10 different people. To control the quality, we assess all tasks after the full completion of the task pool. We set a minimum time threshold equal to the average time of completion multiplied by 75% (to take into account the variance) and include into each set of tasks “honeypot”, which could be solved incorrectly only by choosing a random answer.

Table 5. Human evaluation results of reliable sentence sampling.

Support (number of judges)	% of tasks	Average NEFO (RSS variant)	Average NEFO difference	Average NEO (RSS variant)
4–5	14.56	55.0	6.55	59.53
6–7	38.83	67.62	16.3	71,53
8–10	46.60	82.15	26.11	57,63

The results are presented in Table 5. First of all, the decision of RSS algorithm received minimal support of 4 people, and the fraction of cases with support

lower than 6 is less than 15%. Almost in half of the tasks the RSS variant had 8 to 10 support. If we study NEFO values, we can see that with a higher difference between the variant chosen by RSS algorithm and closest alternative (most frequent human judge variant) and average NEFO values (for RSS variant) more than 80% there is a high probability that the RSS variant is the most correct. And on the contrary, if NEFO is lower than 60% and the difference is negligible then RSS algorithm might choose a suboptimal variant. The positive correlation between the NEFO score of the RSS variant and human judge support observed in the table proves that the algorithm seeks the most factually correct variant and improves the overall factual consistency of generated summary.

After inspecting tasks with the lowest support, we found the main flaw of NEFO metric: the metric ignores the facts that do not distort the main idea of the sentence but may be incorrect from general knowledge. An example of such a task is demonstrated in Table 6.

Table 6. Example of a task with low support.

Original	Translation
Source text	
Водолазы обнаружили в Черном море не фюзеляж самолета Ту-154, а крупные обломки, передает РИА «Новости» со ссылкой на представителя Южного регионального поисково-спасательного отряда МЧС. «Там не фюзеляж, там крупные обломки, на место направлен аппарат «Фалькон» для обследования местности, по результатам обследования будет приниматься решение об их подъеме», — сообщил он.	The divers found in the Black Sea not the fuselage of the Tu-154 aircraft, but large debris, RIA Novosti reports with reference to a representative of the South Regional Search and Rescue Unit of the Ministry of Emergency Situations. “There is no fuselage, there is large debris, a Falcon apparatus has been sent to the site to survey the terrain, and a decision will be made to lift them based on the results of the survey,” he said.
Sentence variants	
Водолазы обнаружили в Черном море не фюзеляж потерпевшего крушение военно-транспортного самолета, а крупные обломки.	The divers found in the Black Sea not the fuselage of the crashed military transport aircraft , but large debris.
Водолазы обнаружили на глубине 27 м в Черном море не фюзеляж, а крупные обломки, сообщил представитель Южного регионального поисково-спасательного отряда МЧС.	Divers found not the fuselage, but large debris at a depth of 27 m in the Black Sea, a representative of the South Regional Search and Rescue Unit of the Ministry of Emergency Situations said.
Водолазы обнаружили на глубине 27 м в Черном море не фюзеляж, а крупные обломки самолета Ту-154, сообщил представитель Южного регионального поисково-спасательного отряда МЧС.	Divers found not the fuselage, but large debris of the Tu-154 aircraft at a depth of 27 m in the Black Sea, a representative of the Southern Regional Search and Rescue Unit of the Ministry of Emergency Situations said.

The text is about additional details on found wreckage of the crashed airplane. Two variants contain additional details about the depth where the wreckage was found. This statement cannot be found in the source article and since NEFO considers numbers as named entities these variants are marked as incorrect. Thus, RSS proceeds with the first variant that does not contain any factual mistakes related to named entities. However, the first variant does contain a factual mistake - it classifies the crashed airplane as a cargo plane while in reality, it is a passenger aircraft. 5 out of 10 judges decided that this is a more serious mistake than unobvious details about the depth and chose the third variant. This suggests that NEFO ranking is similar to the model of perception of the average reader: the correctness of emphasized text fragments (e.g., named entities or citations) has the highest importance.

8 Conclusions

In this work, we studied the possible ways of improving summary reliability by reducing the rate of incorrect hallucinations. From existing methods, dataset cleaning proved to be the most efficient. The second efficient method, truncated loss, showed half as much quality gain, while in conjunction with dataset cleaning it lowers quality of the generated summary. We proposed a new method for summary sampling that improves factual consistency by choosing the most reliable sentences. To measure reliability, we proposed two new metrics: NEO and NEFO. We demonstrated the efficiency of reliable sentence sampling in improving the global reliability of resulting summaries as well as proved the correlation of NEFO ranking with human judgment.

Acknowledgments. The study is supported by Russian Science Foundation, project 21-71-30003.

References

1. Cao, Z., Wei, F., Li, W., Li, S.: Faithful to the original: fact aware neural abstractive summarization. In: Thirty-Second AAAI Conference on Artificial Intelligence, pp. 4784–4791 (2018)
2. Chernyshev, D., Dobrov, B.: Abstractive summarization of Russian news learning on quality media. In: van der Aalst, W.M.P., et al. (eds.) AIST 2020. LNCS, vol. 12602, pp. 96–104. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-72610-2_7
3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: 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), pp. 4171–4186 (2019)
4. Dušek, O., Howcroft, D.M., Rieser, V.: Semantic noise matters for neural natural language generation. In: Proceedings of the 12th International Conference on Natural Language Generation, pp. 421–426 (2019)

5. Falke, T., Ribeiro, L.F.R., Utama, P.A., Dagan, I., Gurevych, I.: 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*, pp. 2214–2220 (2019)
6. Filippova, K.: Controlled hallucinations: learning to generate faithfully from noisy data. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16–20 November 2020. Findings of ACL*, vol. EMNLP 2020, pp. 864–870 (2020)
7. Goodrich, B., Rao, V., Liu, P.J., Saleh, M.: Assessing the factual accuracy of generated text. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 166–175 (2019)
8. Grusky, M., Naaman, M., Artzi, Y.: Newsroom: a dataset of 1.3 million summaries with diverse extractive strategies. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 708–719 (2018)
9. Gusev, I.: Dataset for automatic summarization of Russian news. In: Filchenkov, A., Kauttonen, J., Pivovarov, L. (eds.) *AINL 2020. CCIS*, vol. 1292, pp. 122–134. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59082-6_9
10. Hermann, K., et al.: Teaching machines to read and comprehend. In: *NIPS* (2015)
11. Huang, Y., Feng, X., Feng, X., Qin, B.: The factual inconsistency problem in abstractive text summarization: a survey. *ArXiv abs/2104.14839* (2021)
12. Kang, D., Hashimoto, T.B.: Improved natural language generation via loss truncation. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 718–731. Association for Computational Linguistics (2020)
13. Kryscinski, W., Keskar, N.S., McCann, B., Xiong, C., Socher, R.: 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)*, pp. 540–551 (2019)
14. Kryscinski, W., McCann, B., Xiong, C., Socher, R.: Evaluating the factual consistency of abstractive text summarization. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9332–9346 (2020)
15. Lewis, M., et al.: 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*, pp. 7871–7880 (2020)
16. Liu, Y., Lapata, M.: 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)*, pp. 3730–3740 (2019)
17. Maynez, J., Narayan, S., Bohnet, B., McDonald, R.: On faithfulness and factuality in abstractive summarization. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1906–1919 (2020)
18. Narayan, S., Cohen, S.B., Lapata, M.: 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*, pp. 1797–1807 (2018)
19. Reimers, N., Gurevych, I.: Making monolingual sentence embeddings multilingual using knowledge distillation. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4512–4525 (2020)

20. Rush, A.M., Chopra, S., Weston, J.: A neural attention model for abstractive sentence summarization. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 379–389 (2015)
21. Scialom, T., Dray, P.A., Lamprier, S., Piwowarski, B., Staiano, J.: MLSUM: the multilingual summarization corpus. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 8051–8067 (2020)
22. See, A., Liu, P., Manning, C.: 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), pp. 1073–1083, January 2017
23. Vaswani, A., et al.: Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 6000–6010 (2017)
24. Wang, A., Cho, K., Lewis, M.: Asking and answering questions to evaluate the factual consistency of summaries. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 5008–5020 (2020)
25. Zhou, C., et al.: Detecting hallucinated content in conditional neural sequence generation. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pp. 1393–1404 (2021)