Abstractive Summarization of Russian News Learning on Quality Media

Daniil Chernyshev(✉) and Boris Dobrov

Lomonosov Moscow State University, Moscow, Russia

Abstract. Summarization is becoming a demanded task in the modern world of ever-increasing document flow. This task allows to compress existing text while maintaining all salient information. However, building a neural summarization model requires training data which is scarce in some languages. In this work, we consider the problem of abstract summarization of news texts in Russian. We propose a new method for obtaining training data that uses the news leads of high-quality media that publishes news in accordance with the classical model. We prove dataset eligibility for training by building an abstractive summarization framework based on pre-trained language models and comparing summarization results with extractive baselines.

Keywords: Abstractive summarization · News summarization · BERT

1 Introduction

Summarization is representing the meaning of the analyzed text in the form of a short abstract. It is one of the most popular tasks in the modern world of ever-increasing document flow. Extractive summary is formed from fragments of the analyzed text. For abstractive summary, words and phrases that were not in the source text can be used. Both extractive and abstractive automatic summarization approaches pose serious challenges, however, only recently the former started receiving solutions with quality comparable to a human-written summary [14]. State-of-the-art architectures employ pre-trained language models which increase comprehension power and process text without complex preprocessing.

In this paper, we propose an algorithm for automatic synthesizing of a dataset for news article summarization. According to the inverted pyramid model[1], the beginning of the "correct" news text ("lead") should reflect the main content of the news. We consider the first paragraph of news published in high-quality media as a source of abstractive summary for other media. We demonstrate the eligibility of the resulting dataset for abstractive summarization experiments by comparing with existing counterparts and building an encoder-decoder framework similar to the state-of-the-art summarization model [9] that outperforms the baseline and produce coherent paraphrases.

[1] https://en.wikipedia.org/wiki/Inverted_pyramid_(journalism).

W. M. P. van der Aalst et al. (Eds.): AIST 2020, LNCS 12602, pp. 96–104, 2021.
https://doi.org/10.1007/978-3-030-72610-2_7

2 Related Work

The emergence of pre-trained language models has opened new ways of improving performance in various natural language processing tasks. By pre-training on tasks dedicated to learning contextual representations, these models extend their comprehension power. For instance, commonly used Bidirectional Encoder Representations from Transformers (BERT) [3] is pre-trained with a masked language modeling and a "next sentence prediction" task.

Abstractive summarization task can be formulated as the extraction of salient concepts from the text and then organizing them into a coherent summary. Among the first successful works in neural abstractive summarization were the work of Rush et al. [15] who employed sequence to sequence model. Their approach was later augmented with recurrent decoders [2]. However, the method was suited only for the problem of generating news article headings. Nallapati et al. [11] extended the model to a multi-sentence summary construction task by adapting CNN/Daily Mail dataset and introducing a hierarchical network. See et al. [14] proposed a pointer generator network to deal with the out-of-vocabulary issue, and coverage mechanism to reduce token repetition. Paulus et al. [13] applied a deep reinforcement learning model to the task and designed a special algorithm to alleviate text degeneration of long summaries. Celikymaz et al. [1] enhanced the original pointer generator network approach with multiple deep communicating agents encoder and decoder with a hierarchical attention mechanism. Gehrmann et al. [4] adopted a bottom-up attention approach to improve the detection of salient tokens. Narayan et al. [12] proposed a new model for generating extremely compressed summaries, based on convolutional neural networks and additionally conditioned on topic distributions. Liu et al. [9] adapted BERT encoder for summarization task and proposed a method for transferring extractive summarization experience to the abstractive summarization model. Zhang et al. [18] designed a two-stage transformer-based decoder that refines the resulting summary with BERT's language knowledge.

3 Dataset

For English news summarization task several datasets were proposed. CNN/Daily Mail [7] dataset was originally proposed for question answering task and contains articles and associated facts. New York Times dataset shares a similar structure however according to Narayan et al. [12], it has less bias towards extractive methods. The main problem with these datasets is that fact set requires additional processing to construct a coherent summary. Xsum [12] has the most abstractive summaries but is intended for extreme summarization which is limited only to one sentence and thus may not contain all important information.

The largest news dataset published to date, Newsroom [5], contains 1.3 million articles with human-written summaries. The dataset was obtained by content scraping of article pages from selected publishers and choosing only articles with summaries with low text overlapping.

To our best knowledge, there are only two published Russian headline generation datasets Lenta.Ru-News[2] and Rossiya Segodnya[3] which does not contain multi-sentence summaries. Gusev [6] published a Russian dataset similar to Xsum but due to lower document quantity it requires extension for more stable model training. It would be convenient to translate the Newsroom dataset using machine translation methods, however, this may corrupt the contents by introducing translation artifacts. Instead, we ease the task by overlooking the extraction of all salient facts and proceeding just to paraphrasing with compression. Inspired by Grusky et al. [5] approach, we design a method to automatically construct a dataset for such task from raw internet resources.

We expanded Lenta.Ru-News dataset by collecting articles from "Коммерсантъ"[4] dating from 2016 to 2019. These publishers do not provide a summary as a separate text, but often incorporate a semblance in the main body as the first paragraph. A human-written summary may be found in metadata, however, for most articles, it takes the form of concatenated text parts of the first paragraph. Simple metadata extraction and filtering high-quality abstractive summaries would yield only a small fraction of data, insufficient for complex model training. To achieve universality and quantity we utilize the first paragraph as pseudo-summary. This approach will require excluding the first paragraph from the source text to prevent data leakage. However, there is no guarantee that the information presented in the excluded part will be reflected in the rest of the text.

To tackle this issue, we exploit the fact that publishers may cover the same story. By finding related articles we can construct pseudo-summary-source pairs by taking the first paragraph from one source and full text from another.

We use "Lenta.ru"[5] articles as a source since this publisher tends to cover most of the stories and usually presents the material earlier than others. "Коммерсантъ" is not a random choice for a target summaries either; this publisher has a special emphasis on business news, so it is expected to provide articles with more analytical background and, thus, more informative sentences. The difference in nature between these two publishers ensures the abstractiveness of paraphrase and strict role assignment provides a consistent style difference which allows to define the task as style transfer with compression.

Pairing is achieved with the following algorithm. For each possible article pair with the same publishing date we calculate TF-IDF cosine similarity. Then for each article-source (text source) of publisher B list top k articles-candidates (summary source) from publisher A according to similarity score (it is assumed that $|A| \leq |B|$). For these candidates in the top list we calculate context similarity by calculating BERT embeddings for each sentence in articles and then building cosine similarity matrix M between sentences of candidate and source.

[2] https://github.com/yutkin/Lenta.Ru-News-Dataset.
[3] https://github.com/RossiyaSegodnya/ria_news_dataset.
[4] https://www.kommersant.ru/.
[5] https://lenta.ru/.

Table 1. Dataset statistics.

	Russian news	CNN	Daily mail	NY times
Mean article length (words)	220.0	760.50	653.33	800.04
Mean summary length (words)	52.6	45.70	54.65	45.54
Lead-3 ROUGE-1	34.69	29.15	40.68	31.85
Lead-3 ROUGE-2	12.21	11.13	18.36	15.86
Lead-3 ROUGE-L	27.69	25.95	37.25	23.75

Sentence embeddings are obtained by average pooling second-to-last hidden layer of BERT of all of the tokens in the sentence. The context similarity between candidate c and target s is defined as the minimum of column-wise and row-wise maximums of matrix M:

$$\text{context}(c, s) = \min\{\max_i M_i;\ \max_i M_i^T\} \tag{1}$$

Finally, we assign candidate articles to the source which maximizes overall context similarity:

$$\begin{gathered}\sum_{\substack{i\in|A|\\ j\in|B|}} \text{context}(c_i, s_j) \to \max \\ \text{pair}(c_i) = s_i;\ \ pair(c_i) \neq pair(c_j),\ \ i \neq j\end{gathered} \tag{2}$$

The convergence of the algorithm depends on the selected number of top articles k. In experiments, we found that good estimation is

$$k = \max\{|A|, |B|\} \cdot 0.15 \tag{3}$$

However, the resulting set of article pairs requires additional refining as it will contain low similarity (or exact) pairs. Low quality pairs are the result of duplicate articles within source set (extensions of previous articles) or lack of event coverage in candidate set. To ensure text-summary connection (possibility of summary extraction) we use ROUGE-N (percentage of overlapping N-grams) and ROUGE-L (share of longest common subsequence). Low similarity pairs are filtered out by simply setting threshold $t_1 = 0.25$ for ROUGE-1 score (F1 measure) between paraphrase and source. And to remove pairs with high text overlapping we set a limit $t_2 = 0.35$ for ROUGE-L.

Our news dataset consists of 26555 article-paraphrase pairs. The data is divided into training (80%), development (10%) and test (10%). Statistics are represented in Table 1. Interestingly, despite the same origin of article and paraphrase (both are parts of main article body), the dataset Lead-3 ROUGE scores are comparable to abstractive summarization counterparts with exclusive human-written summaries. Since summaries tend to repeat some parts of source article this demonstrates acceptable level of data leakage as well as indicates the similarity of tasks.

4 Model

We use a standard encoder-decoder framework for abstractive summarization [14]. The encoder captures salient text information and encodes it in vector form. Many approaches employ separate attention mechanism to determine word (token) importance, however, modern Transformer-based [16] language models such as BERT [3] incorporate it in the encoding mechanism. In addition, BERT is specifically pre-trained for text comprehension which makes it a preferable choice for encoding of salient information.

The decoder takes encoded vector and attempts to decode it in "the right way". "The right way" depends on problem formulation or, to be more precise, target structure. If the problem is formulated as the extraction of salient sentences, then the decoder tries to reconstruct these sentences by extracting information from input. And if it is style transfer, the decoder builds a new version of the text with respect to salient word distribution. Text compression is not a separate task but in fact, a simplification that removes the minimum length constraint and allows the decoder to produce sentences of any length. In training phase decoder learns to produce position-vise token distribution so length constraint is imposed by target sentence length distribution. The coherence of produced sentences is also determined by target token distribution as it means the dependency of current position token distribution from previous. And paraphrasing is a difference between source and target global token distribution.

As for encoder, we use BERT and decoder consists of 8-layer Transformer. There are other decoder architectures that have proved to be efficient, however, our goal is to demonstrate the similarity of style transfer with compression and abstractive summarization tasks, so we aim for simplicity.

The BERT is prepared in accordance with Liu et al. [9] approach. However, we do not pre-train BERT on extractive summarization task. Taking into account the nature of the training dataset it is evident that pre-training on extractive summarization would lead to favoring sentences at the beginning of the source, lowering the comprehension power of the abstractive summarization model. Alternatively, we use a special version of RuBERT [8] pre-trained on Russian news article language[6].

Since the encoder is pre-trained and the decoder must be trained from scratch the training could become unstable. For example, the encoder might overfit the data before the decoder finished the fine-tuning, or vice-versa. To alleviate this issue, we adopt a scheduled learning rate mechanism [16].

Both encoder and decoder use Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, but the learning rate is set according to the formula:

$$lr = \hat{lr} \cdot \min\{step^{-0.5}, \text{step} \cdot \text{warmup}^{-1.5}\} \tag{4}$$

where $\hat{lr} = 2e^{-3}$ and warmup $= 40000$ for encoder, $\hat{lr} = 2e^{-1}$ and warmup $=$ 20000 for decoder. This way by the end of the encoder warmup stage the decoder will accumulate enough gradients to become stable.

[6] https://github.com/dciresearch/RuBERT-News.

Table 2. ROUGE F1 results on test set of Russian news dataset.

Model	ROUGE-1	ROUGE-2	ROUGE-L
Baseline			
Oracle	47.92	29.62	44.99
Lead-3	33.28	13.01	27.30
Pointer generator	31.63	13.82	31.62
TextRank	27.95	10.95	27.16
Ours			
RuBERTAbs (Standard)	30.18	12.45	27.17
RuBERTAbs (News)	36.52	15.80	33.59

5 Implementation

For model implementation we used PyTorch. We applied dropout with probability 0.1 and label smoothing with smoothing factor 0.1. The Transformer decoder has 768 hidden units and the hidden size for all feed-forward layers is 2048. The models were trained for 150000 steps on a single Tesla P100. The choice of training steps was determined by model performance on validation set, where it was concluded that further training resulted in text degeneration and overfitting.

For paraphrase decoding we used beam search with beam size 5 and length normalization [17] with $\alpha = 0.7$. To avoid token repetition, we adopt a trigram blocking mechanism [13]. In consequence to BERT's WordPiece tokenizer we do not need any copy mechanism to deal with out-of-vocabulary words as it gives the ability to generate substitution according to context.

6 Results

For dataset we evaluated unigram and bigram overlap (ROUGE-1 and ROUGE-2) and longest common subsequence (ROUGE-L) without stemming. These metrics were used in all previous works and proved to be informative in terms of similarity to a human-written summary.

We show our results in Table 2. The first section of the table covers the baseline methods: TextRank [10], Pointer generator [14], Lead-3 and Oracle. Lead-3 is just the first 3 sentences of the source text. The Oracle is the best possible result that could be obtained with extractive summarization methods. To construct an oracle summary, we use a greedy algorithm to select a set of source sentences that maximizes the ROUGE-2 score for the target summary. This set could be considered as an upper bound for any summarization task.

The second section demonstrates the results of two variants of our model: RuBERT based abstractive summarization model (RuBERTAbs) with standard RuBERT (Standard) and a special version for news language (News). As it can be seen, the standard variant falls short of even Lead-3 baseline while the

variant pre-trained on news articles non-Oracle baselines in terms of all ROUGE scores. That suggests that pretraining BERT encoder on tasks with the same language (set of possible words) as the main one is an efficient way to boost the performance of the full model. The difference between Lead-3 and RuBERT (News) may be considered not substantial, however, the results of the best models on CNN/Daily Mail dataset show a similar margin [9].

Table 3. ROUGE-1 F1 comparison Prediction with Gold and Lead-3

Gold vs. Lead-3	Prediction vs. Lead-3	Prediction vs. Gold
33.28	40.96	36.52

We investigated some properties of the resulting solution (Prediction) compared with the leads of the analyzed news: Lead-3 (news lead of Lenta.ru) and Gold (news lead of "Коммерсантъ"). It turns out to be closer to each of the Lead-3 and Gold than they are among themselves (Table 3).

Table 4. ROUGE-1 Recall results on test set of Russian news dataset.

Model	Lead-3	Gold	Prediction
Lead-3 $\wedge$ Gold	37.08	30.18	60.13

Prediction is much closer to the intersection of Lead-3 and Gold than each of the news leads (Table 4). This indirectly indicates the automatic selection in the abstractive summary more important details of news content and ignoring the secondary. Table 5 provides a translated example of the resulting annotation.

7 Conclusion

In this paper, we demonstrated the application of pre-trained language models for abstractive summarization of Russian news. We proposed a method for building a learning dataset for the task and developed a fine-tuning process for proper training. The experimental results across the dataset show that the model outperforms simple extractive and abstractive baselines and indicate the importance of BERT pretraining on task's language. While we were preparing this article, we discovered a parallel work [6] that also considered the problem of abstractive summarization and proposed a new dataset with human-written summaries for Russian language. In further work, we plan to use this data to improve our approach and adopt methods with additional conditions for a summary generation.

Acknowledgements. We thank Mikhail Tikhomirov (Research Computing Center of Lomonosov Moscow State University) for providing pre-trained models for our research.

9 Appendix

Table 5. Translated example of output summaries on Russian news dataset.

Original text				
British boxer Tyson Fury announced his return to the professional ring. The athlete announced this on his Twitter account. "Breaking news! The big comeback will take place on May 13th. We are working on finding an opponent. Follow the news," Fury wrote in his microblog. On December 3, Fury announced his return to boxing after recovering from drug addiction...				
Model	Summary	R1	R2	RL
Lead-3	British boxer Tyson Fury announced his return to the professional ring. The athlete announced this on his Twitter account. "Breaking news!	43.47	24.99	43.47
Pointer generator	British boxer Tyson Fury has negotiated with the footballer's rival WBO and WBA World Pion, Fury said. We are working on finding an opponent on Twitter	51.06	32.65	55.31
RuBERTAbs (News)	British boxer Tyson Fury announced his return to the United States. "We want to fight on May 13," he tweeted	55.55	37.29	51.85
Oracle	British boxer Tyson Fury announced his return to the professional ring. The athlete announced this on his Twitter account. The big comeback will take place on May 13th. We are working on finding an opponent. Follow the news, "Fury wrote in his microblog.	65.51	39.39	65.51
Gold	British boxer Tyson Fury said he will return to the ring on May 13. "We are working on finding an opponent," he wrote on Twitter			

References

1. Celikyilmaz, A., Bosselut, A., He, X., Choi, Y.: Deep communicating agents for abstractive summarization. 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. 1662–1675 (2018). https://doi.org/10.18653/v1/N18-1150
2. Chopra, S., Auli, M., Rush, A.M.: Abstractive sentence summarization with attentive recurrent neural networks. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 93–98 (2016). https://doi.org/10.18653/v1/N16-1012
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). https://doi.org/10.18653/v1/N19-1423
4. Gehrmann, S., Deng, Y., Rush, A.: Bottom-up abstractive summarization. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 4098–4109 (2018). https://doi.org/10.18653/v1/D18-1443

5. 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). https://doi.org/10.18653/v1/N18-1065
6. Gusev, I.: Dataset for automatic summarization of russian news (2020), arXiv:2006.11063
7. Hermann, K., Kovciský, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., Blunsom, P.: Teaching machines to read and comprehend. In: NIPS'15, p. 14 (2015)
8. Kuratov, Y., Arkhipov, M.: Adaptation of deep bidirectional multilingual transformers for russian language (2019), arXiv:1905.07213
9. Liu, Y., Lapata, M.: Text summarization with pretrained encoders, pp. 3721–3731 (2019). https://doi.org/10.18653/v1/D19-1387
10. Mihalcea, R., Tarau, P.: TextRank: bringing order into text. In: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, pp. 404–411 (2004)
11. Nallapati, R., Zhou, B., dos Santos, C., Gulcehre, Ç., Xiang, B.: Abstractive text summarization using sequence-to-sequence RNNs and beyond. In: Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning, pp. 280–290 (2016). https://doi.org/10.18653/v1/K16-1028
12. 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). https://doi.org/10.18653/v1/D18-1206
13. Paulus, R., Xiong, C., Socher, R.: A deep reinforced model for abstractive summarization (2017)
14. See, A., Liu, P., Manning, C.: Get to the point: Summarization with pointer-generator networks, pp. 1073–1083 (2017). https://doi.org/10.18653/v1/P17-1099
15. Sutskever, I., Vinyals, O., Le, Q.: Sequence to sequence learning with neural networks. In: NIPS'14, p. 10 (2014)
16. Vaswani, A., et al.: Attention is all you need. In: NIPS'17 (2017)
17. Wu, Y., et al.: Google's neural machine translation system: Bridging the gap between human and machine translation (2016), arXiv:1609.08144
18. Zhang, H., Cai, J., Xu, J., Wang, J.: Pretraining-based natural language generation for text summarization. In: Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL) pp. 789–797 (2019). https://doi.org/10.18653/v1/K19-1074