Sentence simplification with ruGPT3

Shatilov A. A.
RANEPA
Moscow, Russia
shatilov-aa@ranepa.ru

Rey A. I.
RANEPA
Moscow, Russia
rey-ai@ranepa.ru

Abstract

This paper describes our solution for the RuSimpleSentEval shared task on sentence simplification held together with Dialogue 2021 conference. Our approach was to filter the provided dataset, finetune the pretrained ruGPT3 model on it and select generated simple candidates based on cosine similarity and ROUGE-L with a complex sentence as an input. The system achieved SARI 38.49 and took third place in the competition. We have reviewed and analyzed examples of simplified sentences produced by the model. The analysis showed that the sentences produced by the system lose the original meaning of the input sentence in about half of the cases.

Keywords: sentence simplification, ruGPT3, fine-tuning, text generation
DOI: 10.28995/2075-7182-2021-20-618-625

Упрощение предложений с помощью ruGPT3

Шатилов А. А.
РАНХиГС
Москва, Россия
shatilov-aa@ranepa.ru

Рей А. И.
РАНХиГС
Москва, Россия
rey-ai@ranepa.ru

Аннотация

В данной статье описано наше решение для соревнования по упрощению предложений RuSimpleSentEval, проводящегося в рамках конференции Диалог 2021. Наш подход заключался в фильтрации предоставленного набора данных, дообучении претренированной ruGPT3 модели и отборе сгенерированных с ее помощью примеров простых предложений на основе их косинусной близости и ROUGE-L ко входному сложному предложению. Система получила значение метрики SARI 38.49 и заняла третье место в соревновании. Мы провели обзор и анализ примеров упрощенных предложений, получаемых с помощью модели. Анализ показал, что упрощенные предложения теряют смысл оригинального сложного предложения примерно в половине случаев.

Ключевые слова: упрощение предложений, ruGPT3, дообучение, генерация текста

1 Introduction

Text simplification consists of modifying the content and structure of a text in order to make it easier to read and understand, while preserving its main idea and approximating its original meaning.

In the RuSimpleSentEval task simplification was performed at the sentence level. In this formulation, the goal is to obtain a simplified sentence from a complex one. The criteria for sentence complexity include presence of complex grammatical constructions, subordinate clauses, the presence of rare and ambiguous words, etc.

Our goal was to evaluate how well the finetuned autoregressive ruGPT3 model would handle the task. We approach the problem in three steps. At first, we use only Russian sentences from provided translated WikiLarge corpus and filter it by cosine similarity and ROUGE-L (Recall-Oriented Understudy for Gisting Evaluation for Longest Common Subsequence) [10] metric between complex and simplified sentences. We keep pairs with high cosine similarity and medium ROUGE-L values.

Further, we finetune pretrained ruGPT3 on the filtered dataset similarly to finetuning for paraphrasing [18].

And finally, we use the finetuned model to generate simplified candidates using the complex sentence as a prompt. To choose from several generated candidates a random forest model is used. It is trained to predict SARI (System output Against References and against the Input sentence) [13] metric on the validation dataset on 4 features: cosine similarity, ROUGE-L, complex sentence length in tokens and generated simplified sentence length in tokens. The candidate with the highest predicted SARI is selected as an output.

Our system achieved SARI 38.49 points on the private test set, with a third place finish.

In this paper, we describe our approach in more detail and analyze quality of generated simple sentences.

2 Related work

Most text simplification models treat sentence simplification as a monolingual machine translation task. Phrase-based and syntax-based translation models [21] were succesfully used for this. There are also such approaches as deletion [4] and candidate reranking [19] models.

Lately, the task of sentence simplification has mostly been handled with Seq2Seq models [16], for example, Nisioi et al. [6]. Zhang and Lapata [20] combined Seq2Seq with reinforcement learning to optimize a reward based on simplicity, fluency, and relevance. Martin et al. [3] enhanced the transformer architecture with conditioning parameters such as length, lexical and syntactic complexity. To improve text generation Lagutin et al. [8] used Implicit Unlikelihood Training - a method for regularizing output by finetuning a language model with policy gradient reinforcement learning.

One of the latest works [12] uses an unsupervised approach to automatically create training corpora for simplification in multiple languages from raw Common Crawl web data and train simplification systems in any language with a controllable generation mechanism.

In similar tasks of summarization and headlines generation in Russian, finetuning of BERT-based models (BertSumAbs, mBART) is usually used [1, 11, 2].

3 Task description

Most text simplification models are trained on parallel data: pairs of complex sentence - simple sentence. There was no such dataset for the Russian language, so the organizers of the shared task prepared it for this competition [14].

The training dataset is based on the English Wikipedia and Simple English Wikipedia materials translated into Russian. It was also allowed to use additional data, such as a corpus of paraphrases of news headlines ParaPhraserPlus [7]. This corpus consists of different variants of headlines (from 2 to 15) for one news item.

The validation and test datasets for the shared task are gathered on a crowdsourcing platform. These datasets consist of pairs of one complex sentence and from 1 to 5 simple sentence variants. All data are presented on the competition github.[1] Datasets sizes[2] are presented in Table 1.

Dataset type	Dataset size
Training Wikipedia (all)	248,111
Training ParaPhraserPlus	1,725,393
Validation	1,000
Public test	1,000
Private test	1,126

Table 1: Dataset sizes

[1] https://github.com/dialogue-evaluation/RuSimpleSentEval

[2] The number of different news items is shown for ParaPhraserPlus, the number of pairs of complex input - simple references is shown for other datasets

SARI (System output Against References and against the Input sentence) is used as an automatic quality metric for the task. It is a lexical simplicity metric that measures ”how good” the words added, deleted and kept by a simplification model are. The metric compares the model’s output to multiple simplification references and the original sentence.

A specific implementation of the metric is used from the EASSE library [5] with a modification to account for a variable number of reference sentences. [3]

4 System description

4.1 Data

For our experiments we used only Russian sentences from the provided dataset. It contains 248,111 pairs of sentences and is quite noisy, so additional filtering was applied. To select good examples we used these metrics calculated between complex and simple sentences:

- cosine similarity of embeddings, obtained with BERT large model (uncased) for Sentence Embeddings in Russian language from Sberbank [4]. It shows how similar the sentences are in terms of meaning.
- ROUGE-L F1-score - Longest Common Subsequence (LCS) based statistics. It identifies longest co-occurring in sequence n-grams automatically. It shows how similar the sentences are in terms of common words.

Joint distributions of these metrics for different datasets are shown in Figure 1. For each dataset, histogram of cosine similarities is shown at the top and histogram of ROUGE-L F1 scores is shown on the right. It can be seen, that training dataset distributions differ from validation dataset, because of alignment errors and because sentences in the Simple English Wikipedia are not always simplified versions of sentences from the English Wikipedia.

Filtered dataset was obtained by choosing sentence pairs that have:

- cosine similarity between 0.6 and 0.99
- ROUGE-L between 0.1 and 0.8
- token length of the simple sentence which is less than or equal to the token length of the complex sentence

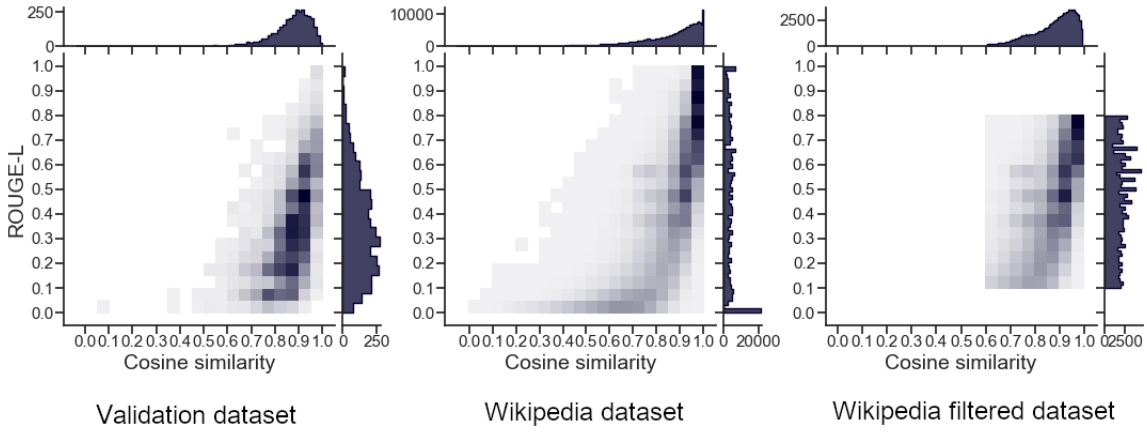

Figure 1: Joint distributions of cosine similarities and ROUGE-L

Final size of the filtered dataset is 120,765 sentence pairs - about half of the original.

[3] https://github.com/Andoree/sent_simplification

[4] https://huggingface.co/sberbank-ai/sbert_large_nlu_ru

4.2 Model

We used a pretrained autoregressive GPT2-like [9] model with 350M parameters from Sberbank [5] called rugpt3medium_based_on_gpt2 - it's the largest model that fit into one 11GB 2080Ti GPU.

Finetuning was done on the prepared examples from the filtered Wikipedia dataset using transformers library [17]. These examples were fed into the model with the addition of special tokens (`<|startoftext|>` - in the beginning, `<|sep|>` - between complex and simple sentences, `<|pad|>` - padding token):

```
<|startoftext|>Complex sentence.<|sep|>Simple sentence.
```

After finetuning it is possible to feed into the model a prepared example as follows:

`<|startoftext|>New complex sentence.<|sep|>` and have the model generate a simplified sentence.

4.3 Selection of generated candidates

Feeding the complex sentence as a prompt into the model can generate several simple sentences. Parameters that were used to generate candidate examples were chosen empirically and presented in Table 2:

Parameter	Value
top_k	50
top_p	0.95
temperature	0.9
max_length	200
length_penalty	0.7
num_return_sequences	5

Table 2: Generation parameters

To select one of the generated candidates as an output we do the following. At first, we generate 5 candidates for each input complex sentence in validation dataset and calculate SARI for each of them. Then, similarly to Wikipedia dataset filtration, we calculate cosine similarity and ROUGE-L between input complex sentence and each of the candidates. Additionally, two more features are calculated: token lengths of input sentence and candidate sentence.

After this we have 4 features:

- cosine similarity between input and candidate
- ROUGE-L between input and candidate
- input sentence token length
- candidate sentence token length

We use the calculated SARI metric as a target and fit a Random Forest model on these features of the provided validation dataset, so that we don't have to manually set tresholds. Setting a small `max_depth` value serves as a regularization. These parameters of the Random Forest model showed the best results: `n_estimators=1000`, `max_depth=5`.

Having trained a Random Forest model on the validation data, we can predict SARI for each generated candidate for the test dataset and choose one with the highest predicted value as an output.

5 Experiments and Results

Code is available on GitHub[6]. rugpt3medium_based_on_gpt2 model was finetuned on one 11GB 2080Ti GPU using transformers library with parameters[7] that are presented in Table 3

[5] https://github.com/sberbank-ai/ru-gpts

[6] https://github.com/InstituteForIndustrialEconomics/DialogueEvaluation21_RuSimpleSentEval

[7] Maximum batch size, that fitted on GPU was 4, for other batch sizes gradient accumulation was used

Parameter	Value
num_train_epochs	3
per_device_train_batch_size	4, 8, 32, 64
learning_rate	5e-5
lr_scheduler_type	linear
warmup_steps	500

Table 3: ruGPT3 finetuning parameters

Random Forest model for generated candidates selection was used from scikit-learn library [15] We used n_estimators=1000 and changed max_depth parameter of the model for the experiments: None, 3, 5, 10.

First, the quality of the models finetuned with the same parameters but on different datasets was compared. Model trained on the filtered Wikipedia dataset showed better results. After that we finetuned model with different batch sizes, and finally evaluated how max_depth parameter of Random Forest candidate selection model influences the result.

Results are presented in Table 4

dataset	Batch size	Max_depth of RF	SARI on the public leaderboard
Original	4	None	37.75
Filtered	4	None	38.30
Filtered	8	None	38.55
Filtered	32	None	38.22
Filtered	64	None	38.34
Filtered	8	3	38.58
Filtered	**8**	**5**	**38.84**
Filtered	8	10	38.65

Table 4: Experiments results

For final submission to private leaderboard the model finetuned on filtered dataset with batch_size=8 was used, max_depth of candidate selection Random Forest model was set to 5. Final SARI score achieved with this system is 38.49. Results of our final model, baseline and the best models are presented in Table 5

Team name	SARI on public	SARI on private	Place on private
Pestova (organizers' baseline)	30.15	-	-
ashatilov (our final model)	38.84	38.49	3
orzhan	40.23	39.28	2
qbic	-	39.69	1

Table 5: Competition results

6 Analysis and Examples

We annotated a sample of 100 simple sentences predicted for the private dataset for the loss of the original meaning of the input complex sentence. The annotation was made in three classes:

- loss of the original meaning - opposite meaning or nonsense
- partial loss of the original meaning - loss of some important detail, but overall meaning is retained
- no loss of the original meaning

Results are presented in the Table 6. As shown in the table, about half of the sentences completely or partially lose the original meaning of input complex sentence.

Result	Number of examples
No loss of meaning	47
Partial loss of meaning	18
Total loss of meaning	35

Table 6: Annotation results of prediction sample

Examples of annotations are presented in Tables 7, 8, 9:

Input:	Взрослых собак, не содержащихся во дворе или открытом вольере, рекомендуется выгуливать минимум два (а желательно три), раза в день.
Output:	Питомников, которые содержат маленьких детей, рекомендуется выпускать на прогулки минимум два или три раза в день.
Input:	Насаждение лучистой сосны в районе Kaingaroa Forest создали крупнейший в мире искусственно выращенный лес.
Output:	В районе Kaingaroa Forest выращивают самую большую в мире светлую сосну.
Input	Земля по отношению к Марсу является внутренней планетой, так же, как Венера для Земли.
Output:	Марс по отношению к Земле - внутренняя планета.

Table 7: Examples with loss of meaning

Input:	После дискуссии заместители получили право задавать вопросы в суде, голосовать на конференциях судей и получать всесторонние консультации.
Output:	Заместители имеют право задавать вопросы в суде и получать подробную информацию из судов.
Input:	Изменение крови характеризуется лейкоцитозом, при тяжёлых формах имеется резкий сдвиг влево с выходом в кровь миелоцитов и юных клеток.
Output:	Изменение крови характеризуется лейкоцитозом и множественными молодыми клетками.

Table 8: Examples with partial loss of meaning

Input:	Вооружённые силы в России не являются самостоятельным политическим субъектом, не принимают участия в борьбе за власть и формировании государственной политики.
Output:	Вооруженные силы России не являются отдельной политической организацией.
Input:	Фэнтези, в форме фольклора, легенд и фантастики, фигурировала в живописи ещё до оформления в отдельный жанр фэнтези-арта.
Output:	Фэнтези фигурировала в искусстве еще до создания отдельного жанра фэнтези-арта.

Table 9: Examples without the loss of meaning

7 Conclusion

In this paper, we evaluated a simple approach for sentence simplification by finetuning a pretrained ruGPT3 model and selecting generated samples based on similarity and difference between input complex and output simplified sentences. The system got SARI 38.49 and placed 3rd on the private leaderboard.

Yet despite high values of the automatic quality metric, sentences produced by the system completely or partially lose the original meaning of the input sentence in about half of the cases. As a result, the system can be used to generate several examples of simplified sentences for further manual selection.

References

[1] Bukhtiyarov Alexey, Gusev Ilya. Advances of Transformer-Based Models for News Headline Generation. — 2020. — 2007.05044.

[2] Chernyshev Daniil, Dobrov Boris. Abstractive Summarization of Russian News Learning on Quality Media // Analysis of Images, Social Networks and Texts / Ed. by Wil M. P. van der Aalst, Vladimir Batagelj, Dmitry I. Ignatov et al. — Cham : Springer International Publishing, 2021. — P. 96–104.

[3] Martin Louis, Sagot Benoît, Éric de la Clergerie, Bordes Antoine. Controllable Sentence Simplification. — 2020. — 1910.02677.

[4] Coster Will, Kauchak David. Learning to Simplify Sentences Using Wikipedia // Proceedings of the Workshop on Monolingual Text-To-Text Generation. — Portland, Oregon : Association for Computational Linguistics, 2011. — Jun. — P. 1–9. — Access mode: https://www.aclweb.org/anthology/W11-1601.

[5] EASSE: Easier Automatic Sentence Simplification Evaluation / Fernando Alva-Manchego, Louis Martin, Carolina Scarton, Lucia Specia // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations. — Hong Kong, China : Association for Computational Linguistics, 2019. — Nov. — P. 49–54. — Access mode: https://www.aclweb.org/anthology/D19-3009.

[6] Exploring Neural Text Simplification Models / Sergiu Nisioi, Sanja Štajner, Simone Paolo Ponzetto, Liviu P. Dinu // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). — Vancouver, Canada : Association for Computational Linguistics, 2017. — Jul. — P. 85–91. — Access mode: https://www.aclweb.org/anthology/P17-2014.

[7] Gudkov Vadim, Mitrofanova Olga, Filippskikh Elizaveta. Automatically Ranked Russian Paraphrase Corpus for Text Generation // Proceedings of the Fourth Workshop on Neural Generation and Translation. — Online : Association for Computational Linguistics, 2020. — Jul. — P. 54–59. — Access mode: https://www.aclweb.org/anthology/2020.ngt-1.6.

[8] Lagutin Evgeny, Gavrilov Daniil, Kalaidin Pavel. Implicit Unlikelihood Training: Improving Neural Text Generation with Reinforcement Learning. — 2021. — 2101.04229.

[9] Language Models are Unsupervised Multitask Learners / Alec Radford, Jeff Wu, Rewon Child et al. — 2019.

[10] Lin Chin-Yew. ROUGE: A Package for Automatic Evaluation of Summaries // Text Summarization Branches Out. — Barcelona, Spain : Association for Computational Linguistics, 2004. — Jul. — P. 74–81. — Access mode: https://www.aclweb.org/anthology/W04-1013.

[11] Malykh Valentin, Porplenko Denis, Tutubalina Elena. Generating Sport Summaries: A Case Study for Russian // Analysis of Images, Social Networks and Texts / Ed. by Wil M. P. van der Aalst, Vladimir Batagelj, Dmitry I. Ignatov et al. — Cham : Springer International Publishing, 2021. — P. 149–161.

[12] Multilingual Unsupervised Sentence Simplification / Louis Martin, Angela Fan, 'Eric de la Clergerie et al. // ArXiv. — 2020. — 05. — Vol. abs/2005.00352.

[13] Optimizing Statistical Machine Translation for Text Simplification / Wei Xu, Courtney Napoles, Ellie Pavlick et al. // Transactions of the Association for Computational Linguistics. — 2016. — Vol. 4. — P. 401–415. — Access mode: https://www.aclweb.org/anthology/Q16-1029.

[14] Sakhovskiy Andrey; Izhevskaya Alexandra; Pestova Alena; Tutubalina Elena; Malykh Valentin; Smurov Ivan; Artemova Ekaterina. RuSimpleSentEval-2021 Shared Task: Evaluating Sentence Simplification for Russian // Computational Linguistics and Intellectual Technologies: Papers from the Annual Conference "Dialogue". — Vol. XX. — 2021. — P. xx–xx.

[15] Scikit-learn: Machine Learning in Python / F. Pedregosa, G. Varoquaux, A. Gramfort et al. // Journal of Machine Learning Research. — 2011. — Vol. 12. — P. 2825–2830.

[16] Sutskever Ilya, Vinyals Oriol, Le Quoc V. Sequence to Sequence Learning with Neural Networks // Advances in Neural Information Processing Systems / Ed. by Z. Ghahramani, M. Welling, C. Cortes et al. — Vol. 27. — Curran Associates, Inc., 2014. — Access mode: https://proceedings.neurips.cc/paper/2014/file/a14ac55a4f27472c5d894ec1c3c743d2-Paper.pdf.

[17] Transformers: State-of-the-Art Natural Language Processing / Thomas Wolf, Lysandre Debut, Victor Sanh et al. // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. — Online : Association for Computational Linguistics, 2020. — Oct. — P. 38–45. — Access mode: https://www.aclweb.org/anthology/2020.emnlp-demos.6.

[18] Witteveen Sam, Andrews Martin. Paraphrasing with Large Language Models // Proceedings of the 3rd Workshop on Neural Generation and Translation. — Hong Kong : Association for Computational Linguistics, 2019. — Nov. — P. 215–220. — Access mode: https://www.aclweb.org/anthology/D19-5623.

[19] Wubben Sander, van den Bosch Antal, Krahmer Emiel. Sentence Simplification by Monolingual Machine Translation // Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). — Jeju Island, Korea : Association for Computational Linguistics, 2012. — Jul. — P. 1015–1024. — Access mode: https://www.aclweb.org/anthology/P12-1107.

[20] Zhang Xingxing, Lapata Mirella. Sentence Simplification with Deep Reinforcement Learning // Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. — Copenhagen, Denmark : Association for Computational Linguistics, 2017. — Sep. — P. 584–594. — Access mode: https://www.aclweb.org/anthology/D17-1062.

[21] Zhu Zhemin, Bernhard Delphine, Gurevych Iryna. A Monolingual Tree-based Translation Model for Sentence Simplification // Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010). — Beijing, China : Coling 2010 Organizing Committee, 2010. — Aug. — P. 1353–1361. — Access mode: https://www.aclweb.org/anthology/C10-1152.