UL FRI Data Science - Natural Language Processing, 2022-2023
Project Report

Paraphrasing sentences in Slovene

Drejc Pesjak[1], Ilija Tavchioski[2], Klemen Vovk[3]

Abstract

With the exponential growth of the natural language processing field using deep learning methods, especially transformers, we can see many *state-of-the-art* models that have great performance on various tasks such as sentiment analysis, machine translation, text generation, sentence paraphrasing, and others. However, most of these methods are trained and used only on data written in English, our research goal is to develop a method for sentence paraphrasing in Slovenian language. We fine-tuned an already pre-trained GPT model (the Slovenian variant) on Slovenian data, and achieved promising performance on our test data, while also having a high average grade from human evaluators. Although the model performance is not yet production ready, it is a valid starting point to enrich the family of transformer-based models to other languages such as Slovenian.

Keywords

Natural Language Processing, Sentence paraphrasing, Transformers

[1] *dp8949@student.uni-lj.si, 63180224* [2] *it8816@student.uni-lj.si, 63180383* [3] *kv4582@student.uni-lj.si, 63190317*

Introduction

The task of paraphrasing sentences is to change the original sentence while preserving the meaning it conveys. For example, *Elephants consume small plants, bushes, fruit, and they can consume up to 170 kilograms of food per day.* can be paraphrased to *An elephant can eat up to 170 kilograms of small plants, bushes and fruit per day.*. Paraphrasing can be used to aid machine translation, improve question answering [1], detect plagiarism and origin [2], and evaluate machine-generated text (i.e. summaries) [3]. This task has been greatly aided by the boom of deep learning (mainly transformers [4]). In this work, we tackle paraphrasing Slovene sentences by fine-tuning a large language model on a dataset of Slovene original and paraphrased sentences that have been obtained by translation from English, as there is vastly more (labeled) data available in English.

Related work

In related work, we can find several approaches to paraphrasing sentences in English and also in other languages as well. Mentioned are rule-based approaches [5] and translation-based approaches [6] which are the most common today, including paraphrasing with monolingual data, (two corpora: one is in English and of high quality, and the other is translated to Slovenian.). We leverage the English corpus to train the model to understand the relationship between paraphrases and then use the Slovenian corpus to teach the model to use the learned relationships to generate paraphrases in Slovenian.

In addition, similar methods are developed using bilingual corpora [7] where the paraphrase is determined by shared translation between the languages. On the other hand, with the rise of transformers [4] we can find several approaches using generative models such as GPT-2 [8] where the model is fine-tuned on a predefined dataset with paraphrasing examples and performs with decent results. In addition, there are some approaches using other techniques such as deep reinforcement learning as well [9].

Methodology

In this section, we describe the methodology we used for the task of paraphrasing Slovene sentences.

Dataset creation

Sentence similarity in machine translation dataset

We explored different methods to create a dataset suitable for fine-tuning a Language Model (LM). One of the approaches we considered was using a machine translation dataset. Specifically, we used the cjvt/rsdo4_en_sl dataset [10], which consists of one million translated sentences from English to Slovene. Our goal was to find similar sentences in English and extrapolate their Slovene counterparts, thereby creating a parallel dataset for training our LM.

To achieve this, we used cosine distance on BERT embeddings [11] to measure the similarity between sentences in English and Slovene. We found that despite using a state-of-the-art language model, the results were not satisfactory. Out of the first 10,000 sentences, only 200 pairs had a similarity score higher than 95%. This was not sufficient for fine-tuning a LM, as we required a larger dataset to improve the performance of our model. Therefore, we concluded that using a machine translation dataset alone was not effective in creating a high-quality parallel corpus.

Slovene paraphrasing with GPT prompting

Another approach we tried for creating a Slovene paraphrasing dataset was using GPT prompting. We used the cvjt/gpt-sl-base model [12] to generate paraphrases for the given input sentence. To do this, we prompted the GPT model to complete the original sentence with a paraphrased version. For instance, given the input sentence ”Today is a beautiful day,” we prompted the model with the phrase ”{input_sentence}, oziroma drugače,” which translates to ”{input_sentence}, or in other words.”

We found that this approach produced some promising results, with the GPT model generating a range of plausible paraphrased sentences for a given input. However, this approach suffered from the slow speed of generation. The cvjt/gpt-sl-base model [12] is a large language model with more than 125 million parameters, and generating paraphrases for a single sentence could take up to several minutes. Additionally, it was sometimes challenging to determine where the GPT model ended the paraphrased sentence. This made it challenging to identify the start and end points of the generated paraphrase, making it challenging to create a high-quality dataset. In the end, not enough data could be generated so we abandoned this method and considered premade datasets.

Machine translation of a paraphrase dataset

The ParaBank2 dataset [13] is a large corpus of 18 million paraphrased English sentences. Our objective was to obtain the Slovene translations of the English sentences, which we tried through two methods.

First, we attempted to use machine translation tools such as DeepL, Google Translate, and other alternatives to generate the translations. Although these tools produced more accurate translations than the Slovene neural machine translation (NMT) model [14], the translation speed was too slow, even when employing proxies to circumvent rate limiting. Consequently, we found this approach unsuitable for amassing a sufficiently large dataset for model fine-tuning.

In the end, we utilized the SloveneNMT model [14] to translate the ParaBank2 dataset locally. As there is no rate limiting associated with this model, we selected the top 100,000 rows of the ParaBank2 dataset [13], which are ordered in descending order of paraphrase quality, for translation. This process was computationally demanding and required 10 hours of processing on an RTX 3090 with 24GB of VRAM. We should note that during translation, some paraphrased sentences were translated into identical sentences, resulting in duplicates that were removed before model fine-tuning. The statistical analysis of the final dataset is shown in Table 1, showing that the training and evaluation sets are not much different in terms of statistical measures.

Statistics	Train		Eval.	
	Sent.	Para.	Sent.	Para.
Avg. #words	8.84	8.84	9.4	9.4
Avg. word’s length	5.27	5.27	5.27	5.25
Min. #words	1	1	1	1
Min. word’s length	1	1	1	1
Max. #words	55	119	55	48
Max. word’s length	31	83	31	31

Table 1. Statistical analysis of the dataset used for training and evaluating the models.

Building a Model

For building our model, we used the GPT-sl-base language model [12], which is a Slovene variant of the GPT family of models. This model was based on the BigScience workshop fork of Megatron and was pre-trained on large Slovene corpora, including Gigafida, KAS, slWaC, and MaCoCu. With about 110 million parameters, it has 12 transformer layers, a dimension of 768, and 16 attention heads. This model can process sequences up to 1024 tokens in length and has a tokenizer trained on a smaller subset of corpora, with a vocabulary of 60k tokens.

Before fine-tuning the GPT-sl-base model on the generated dataset, we split the translated dataset into three separate sets for training, evaluation, and testing. Specifically, we use a 60%, 20%, and 20% split for the training, evaluation, and testing sets, respectively. This allowed us to train the model on a sufficiently large dataset while also evaluating its performance on unseen data to ensure that the model was not overfitting to the training set.

To fine-tune this pre-trained GPT-sl-base model on our generated dataset, we use transfer learning. The idea is to take a pre-trained model, which has already learned features on large datasets, and use it to adapt to a new, smaller dataset with the knowledge it has already gained previously. We fine-tuned the GPT-sl-base model using the dataset we created in the earlier stages of our project. During the fine-tuning process, we trained the model with the parameters showed at Table 2. Our first fine-tuned model was the model at checkpoint of 1.23 epcohs and the second one is at the checkpoint of 1.65 epochs. We trained the model on the Kaggle platform, utilizing two Tesla T4 GPUs with 16GB of memory, and the training took approximately 12 hours before capping out due to Kaggle limitations.

Parameter	Value
Epochs	3
Weight decay	0.01
Learning rate	0.00005
Batch size	16

Table 2. Parameters of the models for fine-tunning.

The results of the fine-tuned model were quite promising, with a training loss drop from 0.36 to 0.076. Although we were not able to train the model for the intended number of epochs, we still managed to achieve a considerable improvement in the model's performance on the generated dataset. This demonstrates that the GPT-sl-base model is an effective starting point for the development of a Slovene paraphrasing LM. Further work could explore the use of other pre-trained models or larger datasets to achieve even better results.

Evaluation

To evaluate the performance Table. 3 of the fine-tuned GPT-sl-base model, we used several metrics that take into account the syntax difference of the sentences along with the semantic difference. All of the following scores range from 0 to 1, with 1 indicating a perfect match between the machine-generated and human-generated translations. In addition, we calculated the metrics on the dataset that was used for evaluation.

BLEU [15], which is a commonly used metric for evaluating the quality of machine-translation output. While the BLEU score is a widely used metric for evaluating machine translation, one of its main limitations is that it only considers the number of matching n-grams in the machine-generated and human-generated translations and does **not take into account the semantic similarity** between the two.

ROUGE-L [16] is another metric that we considered, it is very similar to BLEU since it also does not take into account the semantic meaning of the sentences as it calculates the longest subsequence of words between the two paragraphs.

BERTScore is our own implemented score, where we first calculate the embeddings of both paragraphs, by using the *SLOBERTa* [17] which is a RoBERTa model pre-trained on data in Slovenian language, and then calculate the *cosine similarity* between the embeddings. In contrast to the previous two metrics, the *BERTScore* shows the semantic differences between the paragraph and its paraphrased counterpart.

ParaScore [18] was the last metric considered for this research. It is also a score that does not show semantic meaning between the paragraphs but is taking into account more features than *BLEU* and *ROUGEL*.

Finally, as shown in Table 3, we almost achieved the similarity between the paragraphs in our dataset with respect to the *BERTScore* and *ParaScore*, and improved with respect to the *BLEU* and *ROUGEL* scores which was expected since many of the paraphrased paragraphs are the same as the initial paragraph.

Another aspect is that the first model that we fine-tuned had a better performance since the second model was clearly beginning to overfit.

Score	Dataset	Model(1.23)	Model(1.65)
BLUE	0.21±0.27	0.33±0.37	0.31±0.36
ROUGEL	0.54 ± 0.24	0.56 ± 0.32	0.55 ± 0.32
BERTScore	0.97±0.02	0.95±0.05	0.95±0.05
ParaScore	0.89±0.11	0.77±0.29	0.76±0.28

Table 3. The performance of our models on the evaluation set.

Human evaluation

In addition to the previous metrics for evaluation, we added human evaluation where we considered three people that are familiar with the problem and have different biases towards the data. Since the dataset is too large and consequently would take an enormous amount of time to manually evaluate each pair of sentences, we randomly selected 100 instances for evaluation. Our grade system was composed of five numerical grades (1, 2, 3, 4, 5), where 5 means that the meaning of the paraphrased sentence totally matches the meaning of the original sentence and 1 means that the paraphrased sentence is completely mismatched.

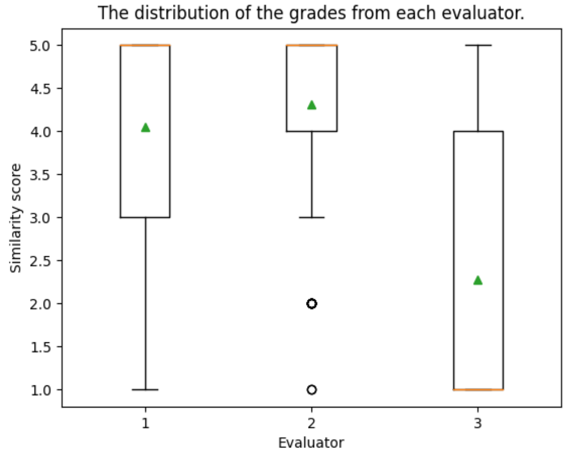

Figure 1. Box plot of evaluators on similarity of 100 sentence pairs paraphrased from the best model.

As we can see in Figure 1, the results from the first two evaluators show the decent performance of the fine-tuned model.

Discussion

In general, the performance of the model is very accurate on short sentences and paragraphs (ex. original sentence: ”V slabšem položaju smo.”, and paraphrased: ”mi smo v slabšem položaju.”). The model can perfectly handle numbers (ex. original sentence: ”štiri dolarje in 27 centov.”, and paraphrased: ”štiri dolarje in sedemintrideset centov.”). There were a lot of short sentences that were duplicated, mainly because of the fact that they were too short to be paraphrased. The models also can handle acronyms (ex. original sentence: ”Uradni list Evropske unije L 210 z dne 31. julija 2006”, and paraphrased: ”(UL L 210, 31. julij 2006”). On the other hand, the model failed to paraphrase longer sentences, a common occurrence is to first successfully paraphrase the sentence and then add some meaningless random text, since the model failed to generate the ”end of sentence” token. Also, the model uses the change of gender pronouns to paraphrase the original sentence which is not desirable.

Conclusion

In conclusion, we obtained the data in the Slovenian language (by translation) for our research, trained a large language model such as GPT (Slovenian variant), and achieved decent performance which can be comparable with the paraphrasing models that are available for other lesser-known and utilized languages. Although the model has worse performance with longer sentences and paragraphs, it works very well on short sentences and paragraphs.

Future work

Our next goal would be to improve our fine-tuned GPT model in terms of performance by selecting better fine-tuning parameters, obtaining a larger dataset (while also cleaning it more thoroughly), improving translations, and considering some other pre-trained large language models for text generation.

In addition to the computational evaluation, we would like to improve the human evaluation by considering more instances for comparison which will give us less uncertainty in measuring the performance of our fine-tuned model.

References

[1] Wee Chung Gan and Hwee Tou Ng. Improving the robustness of question answering systems to question paraphrasing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6065–6075, 2019.

[2] Saša Petrović, Miles Osborne, and Victor Lavrenko. Using paraphrases for improving first story detection in news and twitter. In *Proceedings of the 2012 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 338–346, 2012.

[3] Liang Zhou, Chin-Yew Lin, Dragos Stefan Munteanu, and Eduard Hovy. Paraeval: Using paraphrases to evaluate summaries automatically. In *Proceedings of the human language technology conference of the NAACL, main conference*, pages 447–454, 2006.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[5] Kathleen R. McKeown. Paraphrasing using given and new information in a question-answer system. In *17th Annual Meeting of the Association for Computational Linguistics*, pages 67–72, La Jolla, California, USA, June 1979. Association for Computational Linguistics.

[6] Regina Barzilay and Kathleen R. McKeown. Extracting paraphrases from a parallel corpus. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ACL '01, page 50–57, USA, 2001. Association for Computational Linguistics.

[7] Jonathan Mallinson, Rico Sennrich, and Mirella Lapata. Paraphrasing revisited with neural machine translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 881–893, Valencia, Spain, April 2017. Association for Computational Linguistics.

[8] Sam Witteveen and Martin Andrews. Paraphrasing with large language models. *CoRR*, abs/1911.09661, 2019.

[9] Zichao Li, Xin Jiang, Lifeng Shang, and Hang Li. Paraphrase generation with deep reinforcement learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3865–3878, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[10] Andraž Repar and Iztok Lebar Bajec. Parallel corpus EN-SL RSDO4 1.0, 2021.

[11] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019.

[12] CJVT. CJVT/gpt-sl-base. https://huggingface.co/cjvt/gpt-sl-base, 2021.

[13] J Edward Hu, Rachel Rudinger, Matt Post, and Benjamin Van Durme. Parabank: Monolingual bitext generation and sentential paraphrasing via lexically-constrained neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6521–6528, 2019.

[14] Iztok Lebar Bajec, Andraž Repar, Jure Demšar, Žan Bajec, Mitja Rizvič, Borut Kumperščak, and Marko Bajec. Neural machine translation model for slovene-english language pair rsdo-ds4-nmt 1.2. 6. 2022.

[15] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Annual Meeting of the Association for Computational Linguistics*, 2002.

[16] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[17] Matej Ulčar and Marko Robnik-Šikonja. Sloberta: Slovene monolingual large pretrained masked language model. *Proceedings of SI-KDD within the Information Society 2021*, pages 17–20, 2021.

[18] Lingfeng Shen, Lemao Liu, Haiyun Jiang, and Shuming Shi. On the evaluation metrics for paraphrase generation, 2022.