University *of Ljubljana*
Faculty *of Computer and Information Science*

Paraphrasing sentences

Franc Benjamin Demšar, Lenart Rupnik, Gregor Zadnik

Abstract

In this report we describe the basic idea of a pipeline for generating a model which paraphrases slovenian sentences. We incorporate ideas from related works and present our initial datasets.

Keywords

Paraphrase, Translation, Slovenian, English

Advisors: Slavko Žitnik, Aleš Žagar

1. Introduction

The task of paraphrasing sentences can be tackled in many ways. Traditional approaches include rule-based and thesaurus-based, while state-of-the-art approaches rely on neural networks [1].

Our approach uses back-translation to generate paraphrases from our chosen Gigafida [2] Slovenian dataset. We will then use this combined data to learn generative models which generates a paraphrase for the input sentence.

We will build two models that generates paraphrases - the base model which will be based on LSTM and the state of the art which will be based on Transformers. We will briefly explain the background of those technologies. In further sections we will present metrics for paraphrasing sentences, both automatic and manual. Having results of those metrics we will discuss about it's strengths and weaknesses.

2. Related works

For a general overview of the task of paraphrasing we turned to Zhou and Bhat, 2021 [1]. We decided to use the translation method to generate training data.

Federmann et al., 2019 [3] explore the possible approaches to translating text. They found that Neural machine translation (NMT) performs better than human translation in many cases. It is also mostly free which makes it a good option for us students. Another important finding was that paraphrasing works best when translating into and from related languages.

For our evaluation metrics we drew inspiration from Shen et al., 2022 [4] who compare existing paraphrasing metrics and propose their own: ParaScore. We will evaluate the model automatically with their proposed metrics (METEOR, BLEU-4, Rouge-1) and manually with subjective metrics such as adequacy, diversity and fluency.

3. Data

The dataset we are using is a subset of the entire Gigafida corpus, called ccGigafida [5]. It contains approximately 9% of the entire Gigafida dataset and is available online.

To train the models for paraphrase generation we needed to use our data and construct ('text', 'paraphrased text') pairs. We did that by translating our corpus into english and back to slovene. We treat the double-translated text as a paraphrase to be used for training. For translating we used the Neural Machine Translation model for Slovene-English language pair [6].

The NMT model has a issues translating some of the text from our dataset; when the input text contains many numbers and special characters such as colons, semicolons, dashes, exclamation marks etc., the translation degrades in quality. Sometimes it adds words which do not appear in the input text at all, and sometimes it discards these characters and generates a nonsensical paraphrase.

Along with this, we also had the following problems:

- The model occasionally tries to translate names (and fails miserably).
- The generated paraphrase is sometimes the same as the original text.
- The generated paraphrase sometimes has an entirely different meaning than the original.

We plan on addressing these problems in the next submission.

4. Proposed models

4.1 LSTM

4.2 T5

The T5 or text-to-text model consists of a single transformer encoder-decoder architecture, where both the encoder and decoder are composed of multiple layers of self-attention and feedforward neural network blocks. The model is trained in a text-to-text fashion, where the input and output sequences are represented as text strings, and the task is framed as a text-to-text translation problem.

The T5 model uses a shared weight strategy, where the same set of weights are used for both the encoder and decoder, as well as for all the different tasks that the model is trained on. This allows the model to efficiently share knowledge across different tasks and datasets, and enables the transfer of pre-trained knowledge to new tasks with minimal additional training.

In addition to the shared weight strategy, the T5 model also incorporates several other techniques to improve its performance, including:

- Pre-training on large-scale unsupervised tasks to learn general-purpose language representations
- Fine-tuning on task-specific supervised datasets to adapt the model to specific tasks
- Data augmentation techniques to increase the amount of training data available
- Multi-task training, where the model is trained on multiple tasks simultaneously to further improve its ability to transfer knowledge across tasks.

Since this model is based on text-to-text predictions, it can be easily used for paraphrasing tasks. For our implementation, we used a pre-trained t5 model that was trained on Slovenian text corpus [7]

5. Methodolgy

5.1 Tokenization

To use paraphrases as input for any natural language processing model, we need to first tokenize the sentences into individual words or sub-words. Once the paraphrases are tokenized using a suitable tokenizer, they can be represented numerically and used as input for the model. The numerical representation can be in the form of a sequence of integers or one-hot encoded vectors, depending on the specific requirements of the model. For tokenizing paraphrases, we can use various approaches, depending on the specific requirements of the model and the task at hand. In our case, we use two different tokenizers. For LSTM model we used *BLANK* tokenizer and for T5 model we used already pretrained tokenizer from HuggingFace[7] since it a good practice to use the same tokenizer as the one that was used for pre-training the model.

Model	Adequacy	Fluency	Diversity
Transformers	90.7	-2.6	11.0
LSTM	NaN	NaN	NaN

Table 1. Table with average scores of metrics for 21 samples that different models scored.

Model	Adequacy	Fluency	Diversity
Transformers	100	0	0
LSTM	NaN	NaN	NaN

Table 2. Table with modus scores of metrics for 21 samples that different models scored.

5.2 Transformers model training

TODO explain the T5 training

5.3 LSTM model training

TODO explain the LSTM training

5.4 Metrics

In this section, we will describe metrics we used. We used automatic techniques as well as manual. For manual we based on techniques that Federmann et al., 2019 used. For automatic we based on metrics that Shen et al., 2022 [4] introduced.

5.4.1 Manual metrics

As manual metrics we used following metrics:

- **Paraphrase adequacy**: We measure semantic similarity between the original sentence and it's candidating paraphrase. We measure it on a scale from 0 to 100 where 0 is not adequate and 100 is perfectly adequate.
- **Paraphrase fluency**: When measuring fluency of candidating paraphrase we compare it with original sentence and select which one is more fluent. We measure it on a scale from -50 to 50 where -50 denotes that original sentence is much more fluent than it's candidating paraphrase, 50 the opposite and 0 neither of them.
- **Paraphrase diversity**: When measuring diversity of the original and candidating paraphrase we assign it a score from 0 to 100 where 0 means that sentences are identical and 100 that they are well diverse. We pay attention to more meaningful changes.

When measuring with manual metrics we used 21 simple sentences in Slovene. We gathered them from the news, which was also the source of the Gigafida dataset that we used for training models.

5.4.2 Automatic metrics

TODO explain the manual metrics

6. Results

.

In 13 cases, the adequacy stayed at 100. In most of the cases it was because the sentence did not change. In those cases also the fluency and diversity resulted in 0. In 2 cases the paraphrase was more fluent than the original, and in 4 cases the original was more fluent. 2 sentences were more fluent because our model made incoherent paraphrases that were essentially just double copies of the original one. In 7 cases the paraphrases were at least a bit diverse and in 3 cases sentences were very diverse. Table 2 shows us that in most cases paraphrases did not change it's original meaning and were very as fluent as original and not diverse at all. It is that way mainly because in most cases, proposed paraphrases were only copies of original sentences. Table 1 shows us that in most of the original meaning stayed the same and that paraphrases were only slightly less fluent than original sentences and also that paraphrases only slightly differ to original sentences.

7. Links

- https://viri.cjvt.si/gigafida/
- https://aclanthology.org/D19-5503.pdf
- https://www.kaggle.com/datasets/aemreusta/paraphrased-articles-using-gpt3
- https://github.com/afader/oqa
- https://slovenscina.eu/prevajalnik

8. Further improvements

Altogether, there are a few issues that we will resolve until next submission and hopefully further improve our paraphrasing results. As already mentioned, our method of translation based paraphrasing doesn't produce the best paraphrasing dataset, with many of them being only copies of original sentences. We also found out that many of them doesn't have any unique meaning that could be paraphrased or contain only numbers without any text. So to make training more efficient, we will filter our dataset from negative examples and use alternative ways of creating more paraphrases. Additionally, we will also increase the size of our dataset from around 7000 examples to at least 100 000, thus improving our model.

We'll also try to create a successful LSTM model to compare results from different models and further fine tune the existing one - T5.

References

[1] Jianing Zhou and Suma Bhat. Paraphrase generation: A survey of the state of the art. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5075–5086, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[2] Gigafida.

[3] Christian Federmann, Oussama Elachqar, and Chris Quirk. Multilingual whispers: Generating paraphrases with translation. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 17–26, Hong Kong, China, November 2019. Association for Computational Linguistics.

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

[5] Nataša Logar, Tomaž Erjavec, Simon Krek, Miha Grčar, and Peter Holozan. Written corpus ccGigafida 1.0, 2013. Slovenian language resource repository CLARIN.SI.

[6] 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. Slovenian language resource repository CLARIN.SI.

[7] lala.

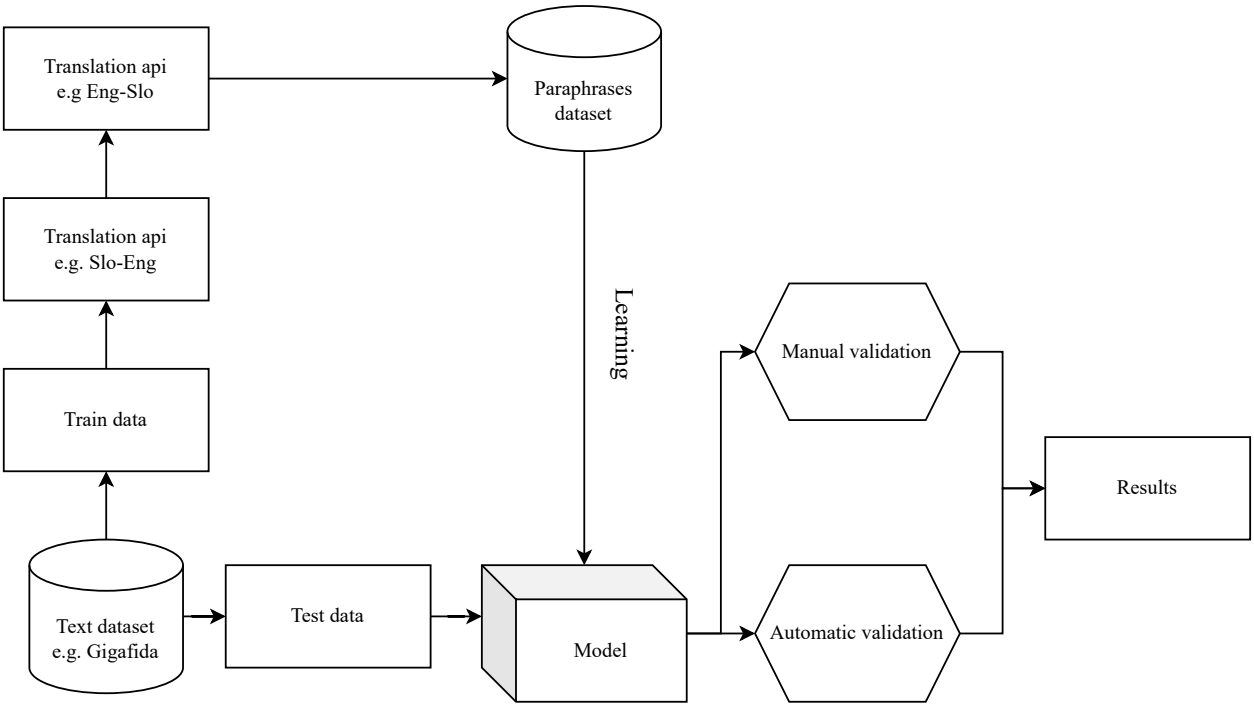

Figure 1. Visualization of a basic pipeline. Detailed visualization of a pipeline we will use to make a model which generates paraphrases.