# Text Summarization

Petcu Mircea, Sandor Cristian*
University of Bucharest, Romania

## 1. Introduction

In this paper, Wepropose an abstractive text summarization method for Romanian language based on Machine Learning. The main requirement in Machine Learning represents the large amount of data required to train the mod els. Hence, Wecreated a data set that consists of the news and their summaries from online press publications, with which Wetrained an architecturebased model Transformer. The problem with current language models is that the number of tokens that can be pro cessed at once is limited due to memory and computational power considerations. Thus, it is often used to truncate the text to a specific number of tokens, losing information. So, Wepro pose a preprocessing method with TextRank to preserve those more important sentences in the truncation process.

## 2. Dataset and preprocessing

**Dataset:**
In developing my summarization methods proposal, Weutilized a dataset from Readerbench in combi nation with my own samples, which were col lected from various news publications in Romania (b365.ro, buletin.de/bucuresti.ro, alephnews.ro) . My part of the dataset was collected automatically with python scripts. To avoid the potential multitude of unknown tokens, Wecleaned the dataset of emojis from the text using the 'demoji' module in Python. Particularly, the data extracted from alephnews.ro had a high proportion of HTML symbols. Wedecided to eliminate all HTML symbols except for "quot;", which Wereplaced with quotation marks to indicate the beginning of a quote. Weremoved any new line markers and eliminated redundant spaces.
In one experiment, Wetried to evaluate the impact of TextRank (Mihalcea and Tarau, 2004) on the objective of summarization. The idea was to preprocess the document intended for summarization to fit the text within a context size (e.g., 512 tokens, 768 to kens). The advantage of this preprocessing method is that it can reduce training time and memory consumption while maintaining the same performance or even improving it.
We applied tokenization on the entire dataset to highlight the relationship between the number of words and the number of tokens. The number of words is highly correlated with the number of to kens, for both the document to be summarized and for the summary. The Pearson correlation (?), calculated for these two entities, is over 0.99. We analyzed the distribution of the ratio between the number of tokens and the number of words, which has an average of approximately 1.38 tokens per word, and the standard deviation is just over 0.08. This rule also applies to a smaller sample. Although the standard deviation tends to increase with the reduction in the number of selected examples, it follows the bellshaped curve of the normal distribution, and the average is centered in all cases at approximately 1.38. Reiterating the points men tioned earlier, the ratio is well accumulated near the average, from which We conclude that We can rely on the number of words instead of the number of tokens in the case of selecting examples to be reduced. Therefore, We set the threshold from where Westart selecting at 371 in the case of the 512 to ken context and 556 in the case of the 768 token context. Although TextRank is considered an algorithm for extractive summarization, We will use it to reduce text size, specifically applying it to the entire text while preserving all sentences and storing them in the order of importance determined by the algorithm. With the sentences sorted by their importance according to TextRank, We can begin the process of text reduction. To avoid the need to tokenize each sentence individually to count the tokens, We choose to rely on the number of words. As mentioned in the previous subsection, there is a close relationship between the number of tokens and the number of words, so We set the threshold where We intend to reach with the documents at 371 words, because We have an average of 1.38 tokens per word, the actual threshold being in fact 512 to kens, similarly for the case of the 768 token context. Therefore, We add sentences in order of importance until the number of words in the selected text exceeds 371, respectively 556 words. Having reduced the text, We sort it according to the order of the sentences in the original text, to preserve its logic.

# Text Summarization

Petcu Mircea, Sandor Cristian*
University of Bucharest, Romania

## 3. Models

**TextRank experiment**
For the textrank experiment, the model We have chosen for conducting the experiments is an MBart Large CC25 (Liu et al., 2020) , which was pretrained on 25 languages for machine translation but can also be used for monolingual summarization. The model has the significant advantage of being pretrained on almost 63 GB of data in Romanian, and it has a fairly large number of parameters ( 680 milions of parameters), making it suitable for more complex tasks such as abstractive text sum marization.
A major impediment to training is the large memory requirement of the training data and the rela tively large size of the chosen model. Because of this, Wetrain the model for abstractive text summarization with QLORA (Dettmers etal., 2023) to be able to use a batch size of 4. Additionally, We use gradient checkpointing and gradient accumulation for at least 4 training steps, eventually resulting in a batch of at least 16 examples, which provides more stability in the training process.
In all the experiments, We chose to quantize the model to normal float 4, to use double quantization, and selected brainfloat16 (Kalamkar et al., 2019) as the data type for computations for numerical stability and the speed of calculations. Addition ally, Weused a dropout for LoRA (Hu et al., 2021) of 0.01, rslora (Kalajdzievski, 2023), and initial ized the LoRA adaptation matrices as specified in the original paper (the first matrix from a Gaussian distribution and the second one initialized with zeros).
The best results, from the perspective of the loss function, were obtained with a rank of 32 and an alpha of 64, with a weight decay of 0.01. We used a batch of 4 with 4 steps of accumulation and cosine annealing for scheduling the learning rate, which starts at 0.0001. We allocated 400 training steps for warming up the learning rate, to stabilize the model and accelerate convergence with AdamW. With all these hyperparameters, Wetrained 4 models:

- A version that has a context of 512 tokens obtained by right truncation.
- A version that has a context of 768 tokens obtained by right truncation.
- A version that has a context of 512 tokens obtained by preprocessing with TextRank to about 371 words and right truncation to 512 tokens.
- A version that has a context of 512 tokens obtained by preprocessing with TextRank to about 556 words and right truncation to 768 tokens.

**Supervized finetuning** Considering that the MBART Large model presented in the previous chapter has achieved quite good results, We will choose it as a candidate in this process. In addition to this, We are also train ing a FlanT5 Large model (Chung et al., 2022), pretrained on the Romanian language. FlanT5 Large represents a slightly different architecture compared to MBART mainly due to the positional representations, which are, this time, relative, [ref] not absolute as in the case of MBART. To achieve better performance than in the previous chapter with MBART, We choose a context of 1024 tokens for it, a context in which the majority of the data from the training set fits. For FlanT5, being a larger model, We cannot train the model with this context in a reasonable time, so We reduce the context to 512 tokens
For model evaluation, We used beam search decoding with a number of beams of 5 and a length penalty of 1.5, as well as greedy decoding. For both, We set the model to repeat a sequence of 5 n grams multiple times, with the minimum number of generated tokens set to 50 and a maximum of 250. The results can be observed in Table 1 and 2.
As a baseline, We used the RoGPT2 Base model trained on abstractive summarization in the Romanian language on a smaller dataset in [ref].

**Long document inference** Additionally, We also experimented with inference on documents much longer than those in the train ing set by preprocessing with TextRank to fit into the set context. For TextRank preprocessing, We used the same method described earlier. We evaluated on a dataset of nearly 1000 examples, each with over 1800 words (words, numbers, punctuation marks). We tested 3 methods:

- RoGPT2 Base (baseline) with greedy decoding
- MBART Large with 1024 context size with greedy decoding
- MBART Large with 1024 context size with beam search decoding

The results can be observed in Table 5 and 6.
**Example:**
Document: "Astăzi, 14 maWe2024, avariile rețeleWede termoficare din BucureștWeafectează un număr impresionant de imobile din oraș. La acest moment, sistemul funcționează normal în proporție de doar 88%, fiind afectate 1123 de blocurWedin toate sectoarele.Chiar dacă există probleme în toate sectoarele Capitalei, cel maWeafectat este Sectorul 6, unde există un total de 576 de blocurWecare întâmpină probleme în ceea ce privește apa caldă de consum".
Summary: "Un număr de 1123 de blocurWedin BucureștWeau rămas fără apă caldă din cauza unor avariWela rețeaua de termoficare".
Reference summary:"Sute de miWede bucureștenWesunt nevoițWesă facă dușurWerecWedin cauza unor noWeavariWeTermoenergetica care îWelasă pe oamenWefără apă caldă. Peste 1000 de blocurWedin Capitală sunt afectate pe problemele actuale ale rețelei"

# Abstractive Text Summarization in Romanian Language

Petcu Mircea
University of Bucharest, Romania
mircea.petcu@s.unibuc.ro

## 4. Results

| Model | Decoding Algorithm | ROUGe1 | ROUGe2 | ROUGeL | ROUGeLsum |
|---|---|---|---|---|---|
| RoGPT2 Base (baseline) | Greedy | 31.26 | 14.79 | 22.94 | 27.51 |
| | Beam Search | 33.26 | 16.0 | 24.0 | 29.3 |
| MBART Large 1024 | Greedy | 37.77 | 19.41 | 27.59 | 33.33 |
| | Beam Search | **38.3** | **19.96** | **27.94** | **33.75** |
| FlanT5 Large | Greedy | 37.28 | 19.28 | 27.49 | 33.12 |
| | Beam Search | 38.08 | 19.89 | 27.84 | 33.73 |

Table 1: The ROUGE metrics the version of the MBart Large model with context of 1024 tokens, RoGPT base finetune on abstractive summarization with context of 724 tokens and FlanT5 Large with context of 512 tokens

| Model | Decoding Algorithm | Precision (%) | Recall (%) | F1-Score (%) |
|---|---|---|---|---|
| RoGPT2 Base (baseline) | Greedy | 68.25 | 69.61 | 68.80 |
| | Beam Search | 69.08 | 70.56 | 69.73 |
| MBART Large 1024 | Greedy | **72.40** | 72.74 | 72.50 |
| | Beam Search | 72.14 | **73.04** | **72.53** |
| FlanT5 Large | Greedy | 72.07 | 72.52 | 72.22 |
| | Beam Search | 71.90 | 73.08 | 72.42 |

Table 2: The BERT Score metrics for the version of the MBart Large model with context of 1024 tokens, RoGPT base finetune on abstractive summarization with context of 724 tokens and FlanT5 Large with context of 512 tokens

| Model | Decoding Algorithm | ROUGe1 | ROUGe2 | ROUGeL | ROUGeLsum |
|---|---|---|---|---|---|
| context 512 | Greedy | 34.49 | 16.29 | 24.49 | 29.75 |
| | Beam Search | 35.56 | 17.38 | 25.33 | 30.91 |
| context 512 TextRank | Greedy | 34.63 | 16.31 | 24.61 | 29.84 |
| | Beam Search | 35.63 | 17.31 | 25.3 | 30.93 |
| context 768 | Greedy | 34.8 | 16.44 | 24.68 | 30.06 |
| | Beam Search | **35.76** | **17.59** | 25.49 | **31.18** |
| context 768 TextRank | Greedy | 34.86 | 16.56 | 24.79 | 30.11 |
| | Beam Search | 35.74 | 17.56 | **25.51** | 31.15 |

Table 3: The ROUGE metrics for each trained version of the MBart Large model: with a context of 512 tokens, with a context of 512 tokens with TextRank preprocessing, with a context of 768 tokens, and with a context of 768 tokens with TextRank preprocessing.

| Model | Decoding Algorithm | Precision (%) | Recall (%) | F1-Score (%) |
|---|---|---|---|---|
| context 512 | Greedy | 72.73 | 70.61 | 71.58 |
| | Beam Search | **71.88** | 71.32 | 71.54 |
| context 512 TextRank | Greedy | 72.61 | 70.75 | 71.59 |
| | Beam Search | 71.73 | 71.42 | 71.51 |
| context 768 | Greedy | 72.75 | 70.82 | 71.7 |
| | Beam Search | 71.82 | **71.48** | 71.59 |
| context 768 TextRank | Greedy | 72.76 | 70.84 | **71.71** |
| | Beam Search | 71.86 | 71.47 | 71.6 |

Table 4: The BERT Score metrics for each trained version of the MBart Large model: with a context of 512 tokens, with a context of 512 tokens with TextRank preprocessing, with a context of 768 tokens, and with a context of 768 tokens with TextRank preprocessing.

| Model | Decoding Algorithm | ROUGe1 | ROUGe2 | ROUGeL | ROUGeLsum |
|---|---|---|---|---|---|
| RoGPT2 Base (baseline) | Greedy | 19.58 | 4.54 | 13.62 | 16.6 |
| MBART Large 1024 | Greedy | **24.84** | 5.32 | 14.61 | 20.91 |
| | Beam Search | 24.81 | **5.78** | **15.09** | **21.3** |

Table 5: The ROUGE metrics for the trained version of the MBart Large model with context of 1024 tokens and RoGPT base finetune on abstractive summarization with context of 724 tokens on long document summarization

| Model | Decoding Algorithm | Precision (%) | Recall (%) | F1-Score (%) |
|---|---|---|---|---|
| RoGPT2 Base (baseline) | Greedy | 63.44 | 61.48 | 62.36 |
| MBART Large 1024 | Greedy | 65.76 | **65.22** | **65.45** |
| | Beam Search | **65.82** | 65.16 | **65.45** |

Table 6: The BERT Score metrics for the trained version of the MBart Large model with context of 1024 tokens and RoGPT base finetune on abstractive summarization with context of 724 tokens on long document summarization

## 5. Conclusion and Future Work

In conclusion, Wecan state that this project has brought me the satisfaction of making a contribution in this field by creating a dataset of respectable size and training models capable for this task. I learned many new things, the most important of which seemed to be the creation of the dataset, training with QLORA, and the sampling methods for generating summaries.
As future work, Wecan mention further adjustment with Direct Preference Optimization to human preferences, experimenting with other measures of similarity between sentences from TextRank, attempting training on a text distribution that does not have a bias towards having information at the beginning, and certainly, training on a larger dataset with larger, more powerful models.