Small BART for summarization of texts in russian language.

Ignatov Dmitry

30 May 2020

Abstract

This article will discuss the possibilities of a 26 times reduced BART architecture for the summarization task in Russian. The research code used is publicly available in the GitHub repository here: https://github.com/IgnatovD/ruBart.

1 Introduction

The BART architecture has shown excellent results in a wide range of tasks and has become the most advanced for the summarization task. To achieve such results, a large model was used that required a huge amount of computing resources. In this study, a reduced copy of BART will be considered. Consider its capabilities for the problem of summarizing the Russian text.

2 Related Work

The problem of summation is traditionally divided into two techniques: extractive and abstractive. There were also works on combining these two approaches. We solve the problem of attracted summation, so we will consider previous work in this direction.

Attention-based encoder was first used in 2015. Their study was published in an article entitled "A model of neural attention for generalizing an abstract sentence.".

A little later, based on this article, Nallapati et al. (2016) used a more powerful sequence-to-sequence model, and Miao and Blunsom (2016) used a variational auto-encoder to output in this model and apply it to the problem of compressing sentences.

Nallapati et al. (2016b) also created a new dataset for summarizing level articles called CNN / Daily Mail.

Vinyals et al., (2015) created the Pointer network, a model that, based on attention, implements a mechanism for copying the most important units from the input sequence. This work laid the foundation for many other studies Gu et al. (2016), See et al. (2017) and Paulus et al. (2017) in which the copy approach was used to solve abstract summation problems.

Chen et al. (2016) and See et al. (2017) struggled with the problem that the model distorts the facts and repeats the same words.

Paulus et al. (2017) introduce a neural network model with a novel intra-attention that attends over the input and continuously generated output separately, and a new training method that combines standard supervised word prediction and reinforcement learning (RL).

Fan et al. (2017) apply convolutional sequenceto-sequence model and design several new tasks for summarization.

Liu et al. (2017) achieve high readability score on human evaluation using generative adversarial networks.

An important job was done by Cibils et al. (2018). Most practitioners point out that models for abstract summarization still use large parts of the source text in the summary, which often makes them look like mining structures. In order to diversify the output of a neural network, a beam search was suggested. Article is called "Diverse Beam Search for Increased Novelty in Abstractive Summarization"

Zhang et al. (2018), Celikyilmaz et al. (2018), Cohan et al. (2018), Chu et al. (2018), Liao et al. (2018), Lebanoff et al. (2018) they investigated the problem of summarization of large texts, as well as summarization at the level of several documents.

Lin et al. (2018), Hardy et al. (2018), Gao et al. (2018), Weber et al. (2018), Kodaira et al. (2018), Li et al. (2018), Fan et al. (2018) solved the problem that the model often suffers from repetition and semantic irrelevance.

Chang et al. (2018), Ma et al. (2018) chinese language has its own characteristics, a group of researchers studied the problem of summarization in Chinese.

Kurniawan et al. (2018) the team introduced a new benchmark dataset for indonesian text summarization.

Chen et al. (2018) the team proposed an accurate and fast summation model that first selects the main sentences and then rewrites them abstractly.

Ramakanth Pasunuru, Mohit Bansal (2018) proposed a reinforced learning approach with two new reward features: ROUGESal and Entail, in addition to the basic coverage level.

Gehrmann et al. (2018) this work proposes a simple technique use a data-efficient content selector to over-determine phrases in a source document that should be part of the summary.

Paulus et al. (2018) the team offers two methods to increase the abstraction level of the generated resumes. First, the decoder is disassembled into the context network, which extracts the corresponding parts of the source document, and a pre-prepared language model, which includes preliminary knowledge about the generation of the language. Secondly, the team offers a novelty metric that is optimized directly through the study of politics in order to stimulate the generation of new phrases.

Kim et al. (2018) introduced a new MMN model and collected a Reddit TIFU dataset consisting of 120 thousand posts from the Reddit online forum, which was used to solve the abstract summarization problem.

Koupaee et al. (2018) In this article, they present WikiHow, a dataset of over 230,000 pairs of articles and summaries, extracted and created from an online knowledge base written by various human authors.

Al-Sabahi et al. (2018) a bidirectional attentional encoder-decoder and bidirectional beam search model was used for abstract summation.

Shi et al. (2018) In 2018, the seq2seq model was very popular. This article reviews the approaches taken to improve in 2018 and introduced the open source NATS library.

Ailem et al. (2019) this article proposes a new decoder in which the summary is generated by matching both the input text and the hidden topics of the document

Wenbo et al. (2019) this paper presents a concept pointer network for improving these aspects of abstractive summarization.

Chadha et al. (2019) proposed DR.SAS, which uses the Actor-Critic (AC) algorithm to study the dynamic distribution of self-control over tokens to reduce redundancy and generate factual and coherent resumes to improve the quality of summation.

Subramanian et al. (2019), Liao et al. (2019) they investigated theproblem of summarization of large texts, as well as summarization at the levelof several documents.

Lebanoff et al. (2019) While recent work in abstractive summarization has resulted in higher scores in automatic metrics, there is little understanding on how these systems combine information taken from multiple document sentences. In this paper, the researchers analyze the outputs of five state-of-the-art abstractive summarizers, focusing on summary sentences that are formed by sentence fusion.

MacAvaney et al. (2019) they used the automatic generation of accurate resumes from clinical reports, which could save a doctor's time, improve overall coverage and reduce errors.

Hoang et al. (2019) they are studying the issue of effectively adapting transformer architecture for tasks in a specific field. In this work, they offer two solutions for the effective adaptation of the prepared transformer language models as text adders: embedding the source code and adaptive subject training.

Han et al. (2019), Duan et al. (2019), Gui et al. (2019) they used various approaches related to the mechanism of attention.

Baan et al. (2019) data scientists are exploring the understanding of multifaceted attention in abstract generalization.

Gabriel et al. (2019) the research team introduces collaborative generator-discriminator networks (Co-opNet), a general framework for abstractive summarization and new dataset, Scientific Abstract SummarieS (SASS).

Gliwa et al. (2019) this paper introduces the SAMSum corpus and showing that model-generated summaries of dialogues achieve higher ROUGE scores than the model-generated summaries of news.

Narayan et al. (2019) set a new task, consisting of one sentence and summary, answering the question "What is this article about?". They collect a large dataset based on articles from the British Broadcasting Corporation (BBC).

Sharma et al. (2019) presents a novel dataset, BIGPATENT, consisting of 1.3 million records of U.S. patent documents along with human written abstractive summaries

Yang Liu, Mirella Lapata (2019) in this paper, they showcase how BERT can be usefully applied in text summarization and propose a general framework for both extractive and abstractive models. GitHub

Shi et al. (2019) In this article, researchers present an open source toolkit, namely LeafNATS, for training and evaluating various sequence-based model sequences for the NATS task of generalizing neural abstraction text, and for deploying pre-trained models in real-world applications.

Li et al. (2019) and Wang et al. (2019) they represent a new models Diverse Convolutional Seq2Seq Model(DivCNN Seq2Seq) and Bi-directional Selective Encoding with Template (BiSET), created specifically for abstractive summarization tasks.

Elgezouli et al. (2020) BERT Fine-tuning For Arabic Text Summarization.

Scialom et al. (2020) data scientists introduced a new approach for decoding sequences, discriminatory match search (DAS). Inspired by generative adversarial networks (GANs), where the discriminator is used to improve the generator, their method differs from the GAN in that the generator parameters are not updated during training, and the discriminator is used only to control the generation of the sequence during output.

Matsumaru et al. (2020) There is a known problem in that the model generates headings that are not relevant to the article. Researchers have suggested that this problem lies in the data set. After checking, they found that there are many examples where the headline does not fit the article. They filtered the data set and improved the quality of the header generation.

Aksenov et al. (2020) Data Scientists are exploring the possibilities of BERT for the task of generalization.

Pilault et al. (2020) interesting study on how an untrained, randomly initialized encoder shows the same results as a trained encoder.

Separately, it is worth highlighting the works of Daniil Gavrilov, Pavel Kalaidin, Valentin Malykh (2019), who used the Universal Transformer architecture and the new RIA dataset in Russian. This work will be taken as a basis.

3 Model Description

The BART architecture is used in a reduced size. The model was taken from the Transformers library. The main characteristics of the model and the differences are listed in Table 1.

Dimension		
Name	**Base**	**Small**
hidden	1024	256
decoder ffn dim	4096	1024
encoder ffn dim	4096	1024
max position embeddings	1024	1024
Number of layers		
encoder layers	16	4
decoder layers	16	4
encoder attention heads	16	16
decoder attention heads	16	16
num hidden layers	12	4
Other		
all parameters	406 291 456	15 579 136
vocab size	52000	30000
tokenized	BPE	BPE

Table 1: Base: a model with default settings, like the original model. Small: the model used in this study.

4 Dataset

The model was trained from scratch. To do this, it was necessary to use two different data sets: one for pre-training from scratch, the other for fine-tuning.

4.1 Russian Wikipedia

This dataset was used for pre-training from scratch. Wikipedia makes database backups. Data is publicly available and can be downloaded. The loaded data is raw HTML markup.

The Wikiextractor library was used to process Wikipedia. The script takes the raw data and processes it. In addition, a script was used from Alexander Veysov, which translates the data into csv format.

Number of sentences	19 621 082
Uncompressed size	15Gb

4.2 RIA

It was planned to use this data set to fine-tune the model for the task of summarization text. The dataset was compiled by Gavrilov, Daniil and Kalaidin, Pavel and Malykh, Valentin.

Russian news agency "Rossiya Segodnya" provided them with a dataset (RIA) for research purposes. It contains news documents from January, 2010 to December, 2014.

Number of news articles	1 003 869
Mean title length	9.5 w
Mean text length	315.6 w
Train dataset	983 869 ar
Test dataset	20 000 ar

Table 2: w - words, ar - articles

This dataset was used in the article Daniil Gavrilov, Pavel Kalaidin, Valentin Malykh (2019), which was taken as a baseline.

5 Experiments

The model training was planned to be divided into two stages: pre-training and fine-tuning. In the original article, it was shown that masked LM (MLM) is a good method of pre-training for the task of summarization. Therefore, it was decided to pre-training using the MLM method.

BART base	
Pre-training method	**Xsum**
w/ Token Masking	**7.08**
w/ Token Deletion	6.90
w/ Text Infilling	6.61
w/ Document Rotation	17.14
w/ Sentence Shuffling	10.93
w/ Text Infilling + Sentence Shuffling	6.62

Table 3: In the original article, testing took place on the XSum dataset.

To encode words into a numerical representation, a byte level BPE tokenizer from the Tokenizers library was used. The tokenizer was trained from scratch on 1/4 of the Russian Wikipedia dataset. The text has not been converted to lowercase. As practice has shown, this amount of data for training was not enough, and it is better to train the tokenizer from the very beginning using a larger amount of data.

5.1 Metrics

In this study, the ROUGE metric was used to evaluate the quality of the model for the summarization problem. ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation. It is essentially of a set of metrics for evaluating automatic summarization of texts as well as machine translation. It works by comparing an automatically produced summary or translation against a set of reference summaries (typically human-produced). The following is an example of calculating the metric for ROGUE-1 precision and ROGUE-1 recall.

$$ROUGE-1_{precision} = \sum_{i=1}^{N} \frac{number\ of\ overlapping\ words}{total\ words\ in\ system\ summary}$$

$$ROUGE-1_{recall} = \sum_{i=1}^{N} \frac{number\ of\ overlapping\ words}{total\ words\ in\ reference\ summary}$$

Using recall and precision, you can also calculate the F-measure.

$$F_{\beta} = (1+\beta^2) \cdot \frac{precision \cdot recall}{(\beta^2 \cdot precision) + recall}$$

The ROUGE-L metric was also used. ROUGE-L – measures longest matching sequence of words using LCS. An advantage of using LCS is that it does not require consecutive matches but in-sequence matches that reflect sentence level word order. Since it automatically includes longest in-sequence common n-grams, you don’t need a predefined n-gram length.

5.2 Experiment setup for pre-training

Unfortunately, the BART article does not describe which hyperparameters were used in pre-training. Since BART is an ideological continuation of BERT, it was decided to focus on the hyper parameters for pre-training mentioned in the BERT article.

Parameters		
	BART small	**BERT base**
Learning rate	1e-4	1e-4
L2 weight decay	1e-2	1e-2
β_1	0.9	0.9
β_2	0.999	0.999
Batch size	64	256
Max number of tokens	128	512
Tokens in batch	≈8K	≈131K
Number of steps	294 089	1 000 000
Activation function	Gelu	Gelu
Optimizer	AdamW	AdamW
Schedule	LambdaLR	LambdaLR
Learning rate warmup	3 065	10 000
The training loss	MLM	MLM + NSP
Computing resources	1 GPU NVidia P100	16 TPU chips total
Computation time (hours)	≈ 28	≈ 96

Table 4: Data on pre-training parameters is taken from the original BERT article. The optimizer and scheduler is taken from the Transformers library.

5.3 Pre-training

During the training, I encountered two problems:

1. At a certain point in time, the loss function assumed nan values.

2. The loss function had poor convergence.

The loss function started taking nan values at a certain point in time. It was experimentally established that this problem is random. Most likely, it arose due to the fact that the tokenizer was trained on insufficient data. Because of this, it encoded some data incorrectly.

At the first training, it was noticed that the loss function does not converge very well. Since there was still a problem that the loss function takes nan values, the hypothesis arose that this is due to the fact that the model takes large weights. It was necessary to assign in the optimizer a larger penalty for large weights. Initially, the original optimizer Adam from PyTorch was used. Then I decided to use the optimizer AdamW from the Transformer library, because there another approach is used to calculate the L2 norm. Replacing the optimizer 3 times improved the result!

Data for pre-training the model was submitted not reduced to lower case. After about 128K steps, data began to be transferred in lower case. This was done in order to prepare the model for working with lowercase texts, since the RIA dataset for the summarization task was in lowercase.

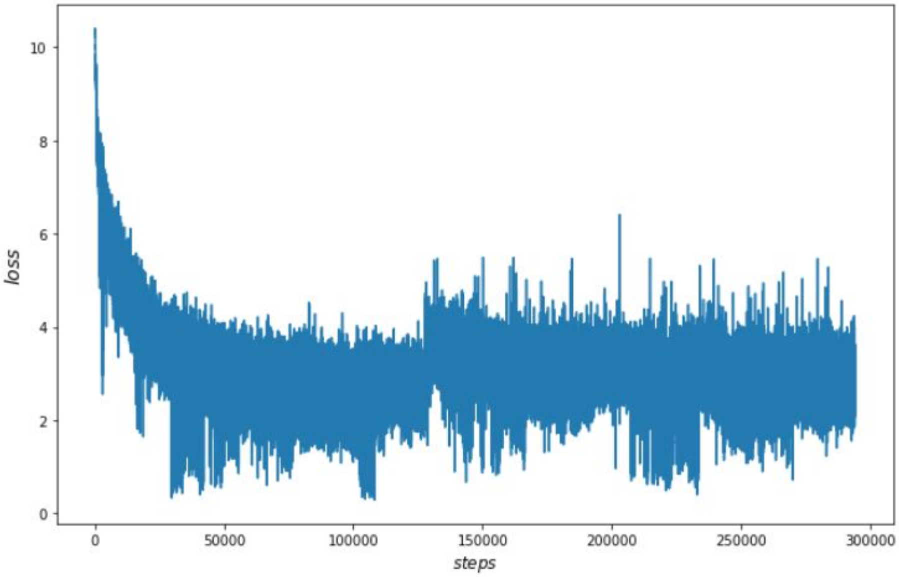


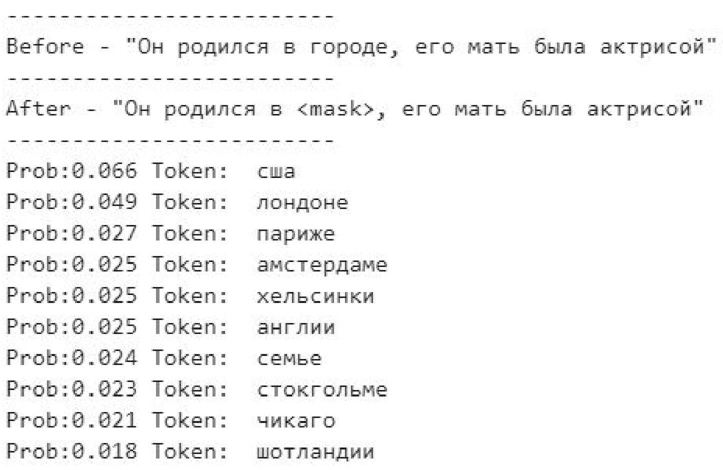
Figure 1: Loss function graph.

During training, every 2000 steps measured the average value of the loss function.

$$mean\ loss = \frac{1}{S}\sum_{i=1}^{S} loss$$

At the end of the pre-training, the average loss function was 2.72.

At this stage, I decided to see if the model learned to predict words. An example can be seen in Figure 2. It also generates text, for the most part repeating the input.



```
------------------------
Before - "Он родился в городе, его мать была актрисой"
------------------------
After - "Он родился в <mask>, его мать была актрисой"
------------------------
Prob:0.066 Token:  сша
Prob:0.049 Token:  лондоне
Prob:0.027 Token:  париже
Prob:0.025 Token:  амстердаме
Prob:0.025 Token:  хельсинки
Prob:0.025 Token:  англии
Prob:0.024 Token:  семье
Prob:0.023 Token:  стокгольме
Prob:0.021 Token:  чикаго
Prob:0.018 Token:  шотландии
```

Figure 2: What the model predicts in place of a disguised token.

5.4 Experiment setup for fine-tune

In an original BART article, it was mentioned that the model trained on a large batch equal to 8000. Such a large batch does not fit on the GPU. To simulate learning with a large batch, a technique called “gradient accumulation” was applied.

Parameters	
Learning rate	1e-4
L2 weight decay	0
β_1	0.9
β_2	0.999
Batch size	16
Gradient accumulation to	8000 bs
Max number of tokens	512
Number of steps	195
Activation function	Gelu
Optimizer	AdamW
Schedule	LambdaLR
Learning rate warmup	3
Computing resources	1 GPU NVidia P100
Computation time (hours)	$\approx$ 12

Table 5: bs - batch size

5.5 Fine-tune

At the fine tuning stage, there was no problem with the values of the nan in the loss function. This was due to the fact that on Wikipedia there were symbols from other languages that the tokenizer did not see during training and could not encode. Therefore, the tokenizer must be trained on the entire dataset for pre-training.

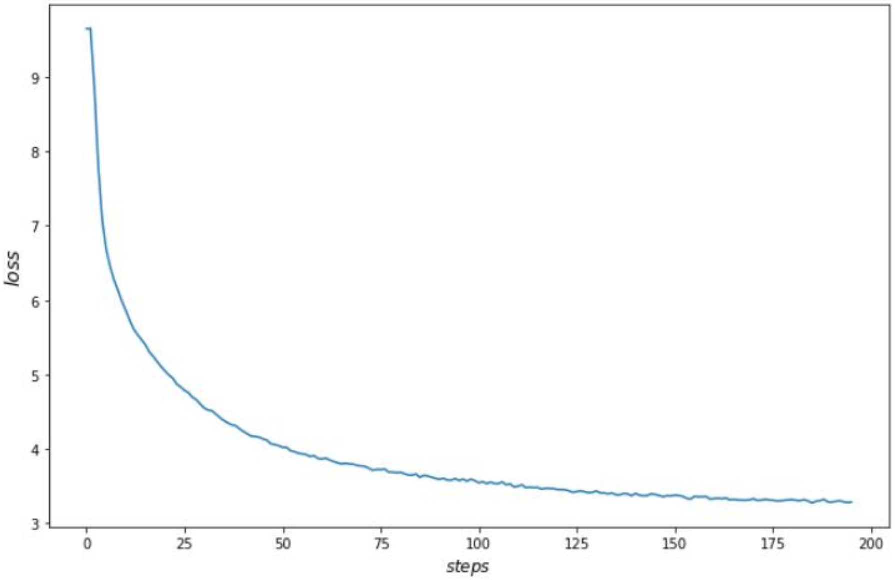
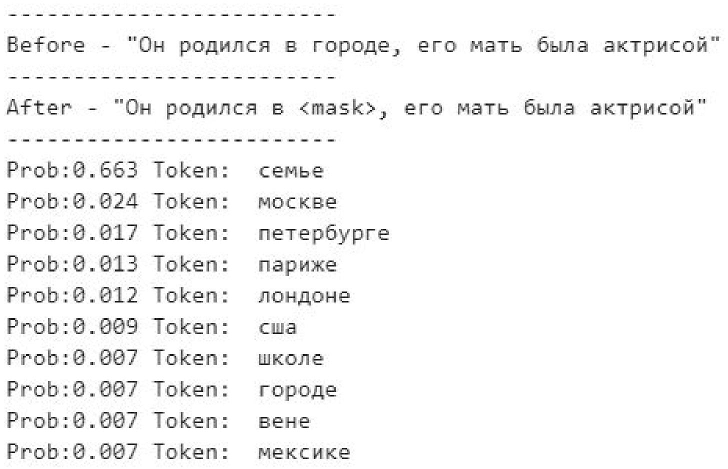


Figure 3: Loss function graph.

It is also interesting to see how the probabilities and model predictions for the MLM problem have changed, after training the model for the task of generalizing the text. It can be seen that the model was rebuilt for the new information received.



```
------------------------
Before - "Он родился в городе, его мать была актрисой"
------------------------
After - "Он родился в <mask>, его мать была актрисой"
------------------------
Prob:0.663 Token:  семье
Prob:0.024 Token:  москве
Prob:0.017 Token:  петербурге
Prob:0.013 Token:  париже
Prob:0.012 Token:  лондоне
Prob:0.009 Token:  сша
Prob:0.007 Token:  школе
Prob:0.007 Token:  городе
Prob:0.007 Token:  вене
Prob:0.007 Token:  мексике
```

Figure 4: Predictions after fine tuning.

6 Results

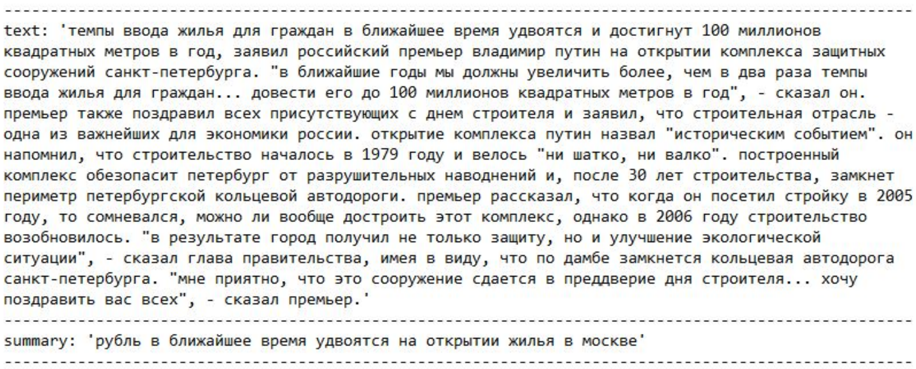
For the experiment, 20,000 random texts were selected. Using the ROUGE metric, the model was evaluated 5 times and the average value was taken. The beam search parameter was set to 10.

RIA						
Model	**R-1-f**	**R-1-r**	**R-2-f**	**R-2-r**	**R-L-f**	**R-L-r**
First Sentence	24.08	**45.58**	10.57	21.30	16.70	**41.67**
Encoder-Decoder	39.10	38.31	22.13	**21.75**	36.34	36.34
UT w/ smoothing	39.31	37.10	21.82	20.66	36.32	35.37
UT	**39.75**	37.62	**22.15**	21.04	**36.81**	35.91
BART small	24.50	23.84	10.69	10.44	24.15	22.57

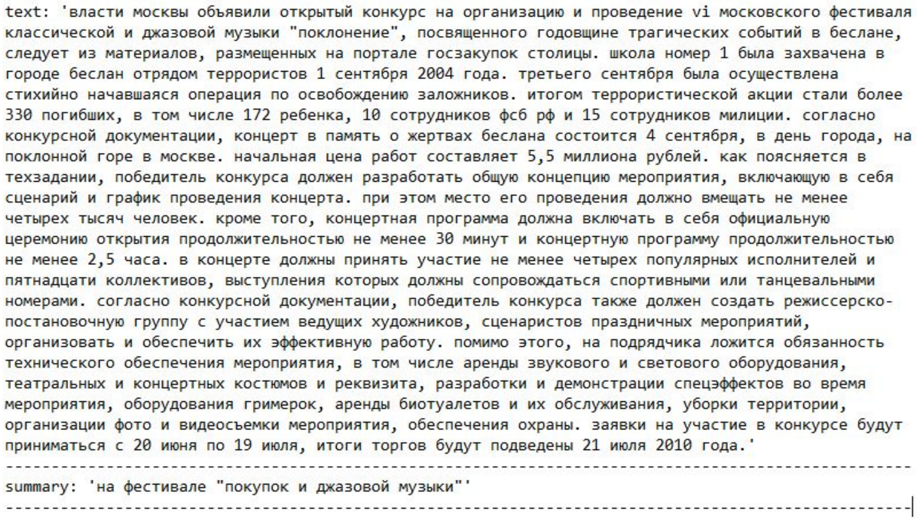
Table 6: UT - Universal Transformer

Examples of summarization that the model made.

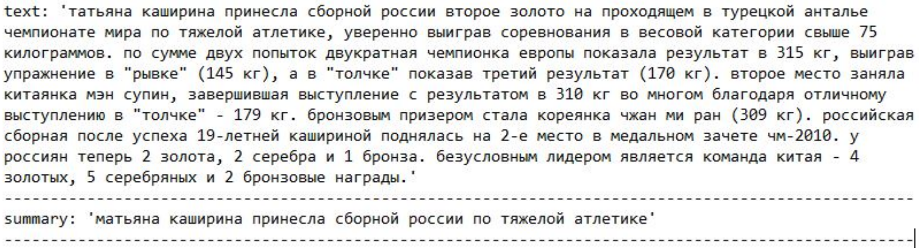
The model does not work well with long texts. The shorter the text, the better the result.



```
------------------------------------------------------------------------------------------------
text: 'темпы ввода жилья для граждан в ближайшее время удвоятся и достигнут 100 миллионов
квадратных метров в год, заявил российский премьер владимир путин на открытии комплекса защитных
сооружений санкт-петербурга. "в ближайшие годы мы должны увеличить более, чем в два раза темпы
ввода жилья для граждан... довести его до 100 миллионов квадратных метров в год", - сказал он.
премьер также поздравил всех присутствующих с днем строителя и заявил, что строительная отрасль -
одна из важнейших для экономики россии. открытие комплекса путин назвал "историческим событием". он
напомнил, что строительство началось в 1979 году и велось "ни шатко, ни валко". построенный
комплекс обезопасит петербург от разрушительных наводнений и, после 30 лет строительства, замкнет
периметр петербургской кольцевой автодороги. премьер рассказал, что когда он посетил стройку в 2005
году, то сомневался, можно ли вообще достроить этот комплекс, однако в 2006 году строительство
возобновилось. "в результате город получил не только защиту, но и улучшение экологической
ситуации", - сказал глава правительства, имея в виду, что по дамбе замкнется кольцевая автодорога
санкт-петербурга. "мне приятно, что это сооружение сдается в преддверие дня строителя... хочу
поздравить вас всех", - сказал премьер.'
------------------------------------------------------------------------------------------------
summary: 'рубль в ближайшее время удвоятся на открытии жилья в москве'
------------------------------------------------------------------------------------------------
```



```
text: 'власти москвы объявили открытый конкурс на организацию и проведение vi московского фестиваля
классической и джазовой музыки "поклонение", посвященного годовщине трагических событий в беслане,
следует из материалов, размещенных на портале госзакупок столицы. школа номер 1 была захвачена в
городе беслан отрядом террористов 1 сентября 2004 года. третьего сентября была осуществлена
стихийно начавшаяся операция по освобождению заложников. итогом террористической акции стали более
330 погибших, в том числе 172 ребенка, 10 сотрудников фсб рф и 15 сотрудников милиции. согласно
конкурсной документации, концерт в память о жертвах беслана состоится 4 сентября, в день города, на
поклонной горе в москве. начальная цена работ составляет 5,5 миллиона рублей. как поясняется в
техзадании, победитель конкурса должен разработать общую концепцию мероприятия, включающую в себя
сценарий и график проведения концерта. при этом место его проведения должно вмещать не менее
четырех тысяч человек. кроме того, концертная программа должна включать в себя официальную
церемонию открытия продолжительностью не менее 30 минут и концертную программу продолжительностью
не менее 2,5 часа. в концерте должны принять участие не менее четырех популярных исполнителей и
пятнадцати коллективов, выступления которых должны сопровождаться спортивными или танцевальными
номерами. согласно конкурсной документации, победитель конкурса также должен создать режиссерско-
постановочную группу с участием ведущих художников, сценаристов праздничных мероприятий,
организовать и обеспечить их эффективную работу. помимо этого, на подрядчика ложится обязанность
технического обеспечения мероприятия, в том числе аренды звукового и светового оборудования,
театральных и концертных костюмов и реквизита, разработки и демонстрации спецэффектов во время
мероприятия, оборудования гримерок, аренды биотуалетов и их обслуживания, уборки территории,
организации фото и видеосъемки мероприятия, обеспечения охраны. заявки на участие в конкурсе будут
приниматься с 20 июня по 19 июля, итоги торгов будут подведены 21 июля 2010 года.'
------------------------------------------------------------------------------------------------
summary: 'на фестивале "покупок и джазовой музыки"'
------------------------------------------------------------------------------------------------
```



```
text: 'татьяна каширина принесла сборной россии второе золото на проходящем в турецкой анталье
чемпионате мира по тяжелой атлетике, уверенно выиграв соревнования в весовой категории свыше 75
килограммов. по сумме двух попыток двукратная чемпионка европы показала результат в 315 кг, выиграв
упражнение в "рывке" (145 кг), а в "толчке" показав третий результат (170 кг). второе место заняла
китаянка мэн супин, завершившая выступление с результатом в 310 кг во многом благодаря отличному
выступлению в "толчке" - 179 кг. бронзовым призером стала кореянка чжан ми ран (309 кг). российская
сборная после успеха 19-летней кашириной поднялась на 2-е место в медальном зачете чм-2010. у
россиян теперь 2 золота, 2 серебра и 1 бронза. безусловным лидером является команда китая - 4
золотых, 5 серебряных и 2 бронзовые награды.'
------------------------------------------------------------------------------------------------
summary: 'матьяна каширина принесла сборной россии по тяжелой атлетике'
------------------------------------------------------------------------------------------------
```

7 Conclusion

As we can see, the BART model, reduced by 26 times, shows poor results for the summation problem. To achieve better results, this architecture requires more parameters and processing power.