# Towards Russian Text Generation Problem Using OpenAI's GPT-2

Oleksii Shatalov, Nataliya Ryabova

*National University of Radio Electronics, Nauky av., 14, Kharkiv, 61000, Ukraine*

**Abstract**

This work is devoted to Natural Language Generation (NLG) problem. The modern approaches in this area based on deep neural networks are considered. The most famous and promising deep neural network architectures that are related to this problem are considered, in particular, the most popular free software solutions for NLG based on Transformers architecture with pre-trained deep neural network models GPT-2 and BERT. The main problem is that the main part of already existing solutions is devoted to the English language. But there are few models that are able to generate text in Russian. Moreover, the text they generate often belongs to a general topic and not about a specific subject area. The object of the study is the generation of a contextually coherent narrow-profile text in Russian. Within the framework of the study, a model was trained for generating coherent articles of a given subject area in Russian, as well as a software application for interacting with it.

**Keywords 1**

Natural Language Generation, Natural Language Processing, Transformers Architecture, Deep Learning, Transfer Learning, GPT-2

## 1. Introduction

The current rate of growth of content is so great that organizations are beginning to fail to keep up with their own set of speeds. Editors and copywriters do not have time to create new texts from scratch, think over ideas for new publications so that they are original. Hiring a large staff of additional staff can significantly increase the costs of the company, which will lead to lower profits. The second option for solving the problem is to reduce or maintain the speed of content formation, which will also give negative results in the future, since the company will be a loser in comparison with competitors. One of the options for solving the problem is the use of the latest artificial intelligence (AI), machine learning and deep learning technologies for such a task as well as others related to Natural Language Processing (NLP) problems. Deep neural networks and their training has become a real breakthrough in solving basic AI problems, including NLP [1, 2, 3, 4]. This area of AI is rapidly developing, there are separate areas within deep learning, such as generative deep learning, reinforcement learning, within which new modern models of deep neural networks are being developed that can solve traditionally complex AI problems faster and, most importantly, more efficiently [4]. The impressive results of deep neural networks are certainly achieved thanks to modern information technologies, such as large-scale machine learning libraries TensorFlow, PyTorch with API for Python language [5, 6, 7, 8, 9].

The main component of many neural language understanding and generating models is pretrained word representation, proposed in [9, 10]. Word embeddings are the basis of deep learning for NLP. Word embeddings (word2vec, GLoVe) are often pretrained on the text corpus from co-occurrence statistics. But learning highquality representations in many cases is challenging task. Word representations are applied in a context free manner. So, the solution of this problem is train contextual representations on text corpus. In the paper [12] authors introduced a new type of deep

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: oleksii.shatalov@nure.ua (O. Shatalov); nataliya.ryabova@nure.ua (N. Ryabova)
ORCID: 0000-0002-7267-6718 (O. Shatalov); 0000-0002-3608-6163 (N. Ryabova)

contextualized word representation, they used vectors from bidirectional LSTM (Long Short-Term Memory recurrent networks). This language model authors called ELMo (Embeddings from Language Models) representations. So we can see that deep neural networks models are the most up-to-date and constantly evolving approach for solving many problems of NLP and NLG.

The rest of the paper is organized in the following way. The state of research and recent advances in deep learning for natural language generation are reviewed in Section 2. In Section 3 general description of the GPT model is given and test runs of the original model are described. Section 4 is devoted to searching Russian language resources with a large database of articles on technological topics, development of software script and formation of dataset. Section 5 contains detailed description of the experiments, taking into account all technical details of the implementation. Experiments include two stages: model learning and model training. The most interesting parts of experiments include train the model to generate whole texts and to generate article titles. In Section 6 experimental results are analyzed. In Section 7 the integration of the models with web application considered. The main characteristics of the proposed web application are described. Conclusions and perspectives for future work are discussed in Section 8.

## 2. Related Works

The neural text generation problem is analized in many works, for example [4, 13]. Authors describe the classic and recently proposed neural text generation models. The development of Recurrent Neural Networks Language Models (RNNLMs) discussed in detail with three training paradigms: supervised learning, reinforcement learning, adversarial training. In 2017, a new simple neural network architecture was proposed, called transformer, based solely on the attention mechanism, without recurrence, i.e. sequential calculations [14]. Transfer learning technology allows you to retrain ready-made models. For Today, this technology is the most promising in deep learning and is used in the most advanced neural models for the generation of natural language texts [15]. The next step was to demonstrate that language models begin to learn NLP tasks without any explicit supervision when trained on a new dataset of millions of webpages called WebText [16, 17]. Authors are researchers from OpenAI and they demonstrate how work their largest model GPT-2 (Generative Pre-Trained Transformer). This is a 1.5B parameter Transformer that achieves state of the art results on 7 out of 8 tested language modeling datasets in a zero-shot setting but still underfits WebText.

Recently some pretrained high-capacity neural language models have become increasingly important in natural language processing and generation. There are such deep neural networks as ELMo [12], BERT [18, 19, 20], GPT-2,3 [15, 21, 22]. They are able to predict the next word in a sequence or some masked word anywhere in a given sequence. BERT (Bidirectional Encoder from Transformers) is neural network from Google, which demonstrated the best results on a number of NLP tasks (machine translation, text analysis, chat bots, answering questions, etc.). Google has released pre-trained models of BERT, but they suffer from a lack of documentation. In fact, BERT from Google is an improved GPT network from OpenAI (bidirectional instead of unidirectional), also on the transformer architecture. BERT is the best in almost all popular NLP benchmarks. Unlike BERT, another popular generative pretrained transformer GPT-2 is created for generating samples of synthetic text with a completely logical narrative, if you give it any beginning [15]. So GPT-2 is available for testing and experimenting with it. Therefore, many researchers and practitioners in AI, NLP and deep learning are trying to solve their problems of generating texts using GPT-2. For example, copywriters and editors will be more focused on editing texts, or writing them on ready-made topics that the model will provide. For such tasks, the Transformers architecture is used, which is able to perceive the context by processing the token chain at once. It should also be noted that many trained models will be required in their subject areas, since at the moment there are no models that could equally successfully generate coherent text in several non-related branches of human activity at once. The multilingualism of the model requires a similar remark.

Thus, it was decided to train a model of a non-profit company OpenAI called GPT-2, namely, a medium-sized model with 300 million parameters, generating texts in Russian about information technologies, blockchain and artificial intelligence. To train the model, the Transfer learning will be used, which allows additional training of ready-made models [21]. This technology will be easy to

apply to the chosen GPT-2 [22]. For such training with so many parameters, a fairly extensive dataset in Russian is required. The preparation of the dataset will also be described in the article.

## 3. GPT model overview

The GPT model is designed to predict the next word in the text, forming, when repeating the operation, a complete coherent text with meaning, context, logic of presentation and completeness of thought. It was originally designed to answer user questions.

According to the developer, a non-profit company OpenAI, the product they offer can be used in the future to help in speech recognition, articles and publications editing, keeping control of the storytelling quality.

The creation of GPT (General Purpose Technology, and later Generative Pretrained Transformer) models began with the announcement and release of GPT-1 in 2018. At that time, of course, it was a breakthrough in the field of text generation and the use of Transfer learning technology, which was new at that time. But, nevertheless, due to the fact that it was trained on a small amount of data, its work left much to be desired. With the announcement of the first generation of GPT, the foundation was laid for continued research in this area.

GPT-2, trained on more than 40 GB of data from 8 million web pages, impressed its own developers so much that the company initially released only a beta version, citing the malicious use of their brainchild to generate fake news, spam, and more. The release of GPT-2 took place in 2019, immediately after the release, the work on GPT-3 has begun.

The 3rd generation GPT made a closed API for the same reason – the possibility of using it to harm. We can also assume that the new model has a sufficiently large amount of data that does not allow it to be disseminated in the way we are used to. Among other things, a fairly large amount of money was spent on the creation of GPT-3, on the order of several million dollars. And this is one of the key factors that does not allow teaching it as effectively even with the knowledge of the mathematical apparatus of the structure of the model.

### 3.1. Test runs of the original model

When starting and initializing the model, there were several questions to be answered:

- What weights to use to work with the model
- What is the maximum length of the generated text
- What is the concept of "temperature" and how it affects the generation
- Consumed resources

To work with the model, the weights of the PyTorch library are used, as well as a special configuration file and an encoder model for storing tokens. The maximum length of the generated text can be any, however the model context window is 1024 tokens. "Temperature" is a parameter that is adjusted during the generation of text by the model, it shows the degree of "madness" of the text, that is, how far the model can deviate from the examples set during training. Average consumed resources were calculated in the middle of Google Colaboratory after 100 launches of each of the original models. The results can be seen in Table 1.

**Table 1**

Consumed resources

| Number of parameters | Occupied disk space, GB | RAM, GB | GPU memory, GB |
|---|---|---|---|
| GPT-2 124M | 0.5 | 2.48 | 2.37 |
| GPT-2 355M | 1.42 | 3.46 | 4.14 |
| GPT-2 774M | 3.1 | 6.09 | 8.28 |
| GPT-2 1558M | 6.23 | 9.98 | 10.69 |

When the generation is started, an initial phrase is sent to the input, through which the context of the generated text is set. The minimum size is 1 token, the maximum is 1023 tokens. Based on

observations, the larger the volume of the input phrase, the longer it takes to generate the text, it should also be noted that the increase in time is not linear.

There were also several launches of all models with different input phrases of the "information technology" subject area. In Figure 1 below, you can see an example of the generated text for the input phrase "The future of machine learning".

```
The future of machine learning in machine learning is looking bright.

Machine learning is a new field with many interesting new applications. The first is mac
hine learning for advanced tasks. The second is machine learning for problem solving.

Machine learning is one of the most popular and most advanced applications of machine le
arning. It is used to identify and solve complex problems in a variety of contexts. It c
an also help to solve complex problems in a variety of situations.

Machine learning is very important to the development of new technologies.
====================
The future of machine learning is also shaping up to be fascinating. In the future, we m
ight see how machines can be used to solve puzzles, create data, and even manipulate dat
a in a real-time fashion.

This is a fascinating time for machine learning. In the past, there were many technologi
es that were designed to solve very real-world problems, but these were developed only f
or humans. Now, there are many tools and frameworks that can be used to solve real-world
problems.

Machine learning is
====================
```

**Figure 1**: An example of the generated text by the original model GPT-2 124M

The pre-trained model works reasonably well in English, generating grammatically correct texts while maintaining context. The 1.5 billion parameter model is expected to have more coherent text than the 124 million parameter model, and is about the same as the 774 million parameter model. But to run a larger model, more resources are needed, and they work longer.

An interesting feature is that the network itself was able to generate, albeit non-existent, but valid links. Sometimes it can get stuck – repeating the same phrase.

In Russian, a network of any size works very poorly – this is due to the fact that they were taught mainly in English. You can verify this by looking at Figure 2, the text " Все, что я могу сказать об этом действии, это то, что" was fed to the model input.

There are several analogs – models, pre-trained on texts in Russian and capable of generating coherent texts of general topics. But it often loops, it is not able to generate an adequate text on a specialized topic because there were no corresponding texts in the training dataset.

Text generation is carried out in a style which is closed to the works of fiction of classical literature.

It is also worth noting that in some cases the model recognizes the text as lyrics and begins to supplement the text with white verse.

There is a noticeable improvement in the use of words in context, and also there are no missing words that do not exist in the explanatory dictionary.

At the same time, there is a noticeable improvement in the use of punctuation marks in comparison with the previous experiment, words and the generation of meaningful text, as well as specific characters (for example, "?" And "!").

For further research, it was decided to take the so-called "average" model of 355 million parameters.

```
��се, что я могу сказать об этом действии, это то, что не моей сказать, что я могу ска
зать об этом действии, это то, что не моей сказать об э
====================
��се, что я могу сказать об этом действии, это то, что в ходя действии часто.

Как вороты в ходя действии да контрой наши полезным фото и советск
====================
��се, что я могу сказать об этом действии, это то, что я могу сказать об этом действи
и, это то, что я могу сказать об этом действии, это то, ч
====================
��се, что я могу сказать об этом действии, это то, что и всегда не примеров дают вам м
астерии.

Купить от эти что пришло не пришло версия действи
====================
��се, что я могу сказать об этом действии, это то, что вы компью на вести, который то
статьях об сказать на вести, который то статьях об сказать
====================
```

**Figure 2**: An example of generating text in Russian by the original model

## 4. Formation of a dataset

As we understood from the text above, for the model for correct work it's required to train it on a sufficiently large amount of text. Thus, the primary task before training the model was the search for Russian-language resources with a large database of articles on technological topics. After the research, a small list of them was formed with an approximate number of articles on the portal. The list can be found in Tables 2, 3 and 4.

**Table 2**
List of portals for receiving thematic texts "Technologies"

| Source name | Approximate number of publications |
|---|---|
| Populyarnaya Mekhanika | 44000 |
| Hightech | 10000 |
| TJ | 5900 |
| Rusbase | 3000 |
| Techliga | 1000 |

**Table 3**
List of portals for receiving thematic texts "Machine learning"

| Source name | Approximate number of publications |
|---|---|
| Habr | 8700 |
| 3D News | 800 |
| ITC | 500 |
| Robotics | 300 |
| Korrespondent | 200 |
| IZ | 200 |
| VC | 200 |

**Table 4**
List of portals for receiving thematic texts "Cryptocurrencies"

| Source name | Approximate number of publications |
|---|---|
| ForkLog | 22000 |
| ProBlockchain | 21000 |
| RBK | 20000 |
| bits.media | 1600 |
| VC | 1200 |
| Habr | 900 |

Due to the number of articles and the approximate amount of text in them, it was decided to form a dataset based on articles from the "Populyarnaya Mekhanika" and "ForkLog" portals.

For further work, a software script was written that unloaded a monthly archive of portals and saved it as HTML pages. For a higher speed of the program, this problem was solved by parallel programming. It allowed to increase the speed of page retrieval by 8 times.

After the pages have been swapped out, they should be processed. We tried several options for processing the dataset. We got the heading under the <h1> tag and all remaining text from each document, except for unnecessary related information, such as embedded Twitter posts, time of article creation, author, tags, etc. Then we implemented:

- Distribution of all articles in different text files
- Distribution of all articles in 1 text file
- Distribution of all articles in different text files with their separation with special service tokens
- Distribution of all articles in 1 text file with their separation with special service tokens
- Distribution of all articles in different text files with separation of articles and article titles with special service tokens
- Distribution of all articles in 1 text file with separation of articles and article titles by special service tokens

As a result, the formation of 1 text file with the separation of articles and article titles with special service tokens is a processing option that has shown the greatest efficiency both in comparison with the training time and in further practical applicability. The separation of the title and the article was done for a reason: it is planned to train a separate model only for generating titles. An example of the appearance of the text can be seen in Figure 3. Inserting tokens is necessary both for separating the constituent parts of the article, and for the model to understand the boundaries of finding one context. Also, in the future, it is the service tokens that will allow us to separate the logically complete parts of the generated text from each other.

After processing and downloading pages from portals, it was already possible to accurately form representations by the number of articles in the dataset. The data is in Table 5.

**Table 5**
Data on the volume of the dataset

| Dataset (source name) | Number of articles |
|---|---|
| Populyarnaya Mekhanika | 51772 |
| ForkLog | 22080 |

```
<|strtfrt|><|strtfttl|>ВВС США хочет установить боевые лазеры на Lockheed AC-130<|ndfttl
|>

Глава подразделения сил специального назначения ВВС США генерал-лейтенант Брэдли Хейтолд
заявил, что одной из задач оборонной промышленности США должно стать оснащение боевыми л
азерами каждой летающей артбатареи Lockheed AC-130 к 2020 году.
Лазер мощностью в 120 киловатт должен весить не более 2,3 тонны, будет использоваться ка
к с оборонительными, так и с наступательными целями и станет самым мощным лазерным оружи
ем, стоящим на вооружении США.
Его первой целью будет защита неповоротливых и крупных летающих батарей от ракет типа «з
емля-воздух». Также будущее лазерное оружие должно фокусироваться для поражения целей уж
е на земле. Подобная пушка может взрывать боеприпасы врага и поджигать топливо, разрезат
ь машины, радары и ракетные установки, рассекая их лазерным лучом.
В отличие от другого типа вооружений мощность лазера можно увеличить или уменьшить в зав
исимости от ситуации, давая самолету больше выбора при поражении целей.
Одним из кандидатов для выполнения этого оборонного заказа является компания General Ato
mics, чей лазер имеет мощность в 75 киловатт и весит 2,3 тонны, поэтому инженерам за сле
дующие пять лет придется понять, как сгенерировать и сохранить еще 45 киловатт мощности,
при этом не увеличив вес установки.<|ndfrt|>
<|strtfrt|><|strtfttl|>Робот стоит первым в очереди за iPhone 6S

iPhone 6S и iPhone 6S Plus появились в продаже сегодня, но очереди желающих приобрести н
овый телефон стали появляться за неделю до события. Одна любительница продукции Apple из
Австралии решила подойти с выдумкой к вопросу, не стала, по примеру многих, ночевать в п
алатке перед магазином, а отправила вместо себя робота.
Устройство, занявшее одно из первых мест в очереди за новым iPhone – это iPad, на которо
м в реальном времени демонстрировалось лицо самой предприимчивой покупательницы Люси Кел
ли. Планшет был укреплен на вершине шеста, а тот в свою очередь крепился к дистанционно
 управляемому колесу. Находясь дома или на работе, Люси могла видеть то, что видел робо
т, болтать с другими людьми в очереди и контролировать движения своего робота.
Робот простоял в очереди два дня, и опередил его только предприниматель Линдси Хармер, к
оторый жил в палатке перед магазином 17 дней до того, как новые iPhone поступили в прода
жу. Сегодня утром Люси Келли получила заветный смартфон, по пути став местной знаменитос
тью.
На фото Линдси Хармер вместо с роботом Люси Келли<|ndfrt|>
```

**Figure 3**: An example of the appearance of a dataset for the whole text of an article

## 5. Experiments to explore model training

After a series of experiments, which were described above, studies were carried out on various devices on the basis of which the training took place.

### 5.1. Choosing the main device for computing during training

Test training of models was carried out on 3 computing devices:

- CPU
- GPU
- TPU

On average, the learning rate on a GPU is 2 times higher than the learning rate on the CPU, and the same rate of learning on the TPU is higher than that on the GPU. Also, the Google Colab environment

offers TPUv2-8 for use, which means a possible division of training into 8 threads, which, in theory, will increase model training by 16 times compared to a GPU. Table 6 shows the elapsed training time for 1 epoch on different devices, based on measurements made during the experiments.

**Table 6**
Time spent on 1 training epoch

| Computing devices | Time, s |
|---|---|
| CPU | 65.1 |
| GPU | 31.6 |
| TPU | 2.3 |

Thus, after several test runs on different devices and receiving data on the elapsed time, it was decided to configure the server with a connection to Google Colud TPUv2-8.

## 5.2. Model training

As mentioned above, we decided to train 2 models: one only for generating text titles, and the second for generating whole texts, including the title. First, a study was carried out according to the second model.

### 5.2.1. Train the model to generate whole texts

In total, about 300 experiments were carried out with models of this type. We changed the markup of the texts, the learning rate, tried a different number of articles from one source or another. Ultimately, about 80% of the models suffered from "looping": a part of the text (most often it was a phrase or a sentence) was repeated several times in the text, making it impossible to supplement the content. A clear example of this can be seen in Figure 4.

```
Децентрализованные биржи обеспечивают прозрачность данных в публичных сетях, обеспечивая
безопасность данных и прозрачность транзакций.
Децентрализованные биржи обеспечивают прозрачность данных в публичных сетях и прозрачнос
ть транзакций.
Децентрализованные биржи обеспечивают прозрачность данных в публичных сетях и прозрачнос
ть транзакций.
Децентрализованные биржи обеспечивают прозрачность данных в публичных сетях и прозрачнос
ть транзакций.
Децентрализованные биржи обеспечивают прозрачность данных в публичных сетях и прозрачнос
ть транзакций.
Децентрализованные биржи обеспечивают прозрачность данных в публичных сетях и прозрачнос
ть транзакций.
```

**Figure 4**: An example of model "looping"

This loop is caused for several reasons:

- Model overfitting
- A small value of the "temperature" parameter, which is responsible for the probability threshold for predicting the next word (accordingly, if the temperature is too high, then everything that is generated will be incoherent text), is set during text generation
- A small "window" for choosing the most probable words, also set during text generation

During the tests, the most optimal values of the parameters above were formed:

- The number of epochs at which the generated text is human-readable is 1000
- The value of the "temperature" parameter was set to 0.8, since at lower values the model began to "loop", and at higher values – to generate incoherent text

- The value of the "window" for taking the most probable subsequent words by the model was set to 40

Also, during training, it was customary to save and test the models every 100 epochs with a small step decrease at small epoch values. Experiments on them showed that the model has not yet learned how to normally generate text for exactly the topic that was laid down in the dataset, and it began to make progress in the latter after the 800th era of training.

Due to the fact that the volume of the generated text was quite small, and the chosen subject area is assistance to editors and copywriters, it was decided to filter the dataset by the number of words. 3000 was taken as the extreme value. Thus, the number of articles in the dataset was reduced to 31686. The results of testing the model confirmed our guess: the articles became longer and the coherence of the text inside them increased. An example can be seen in Figure 5.

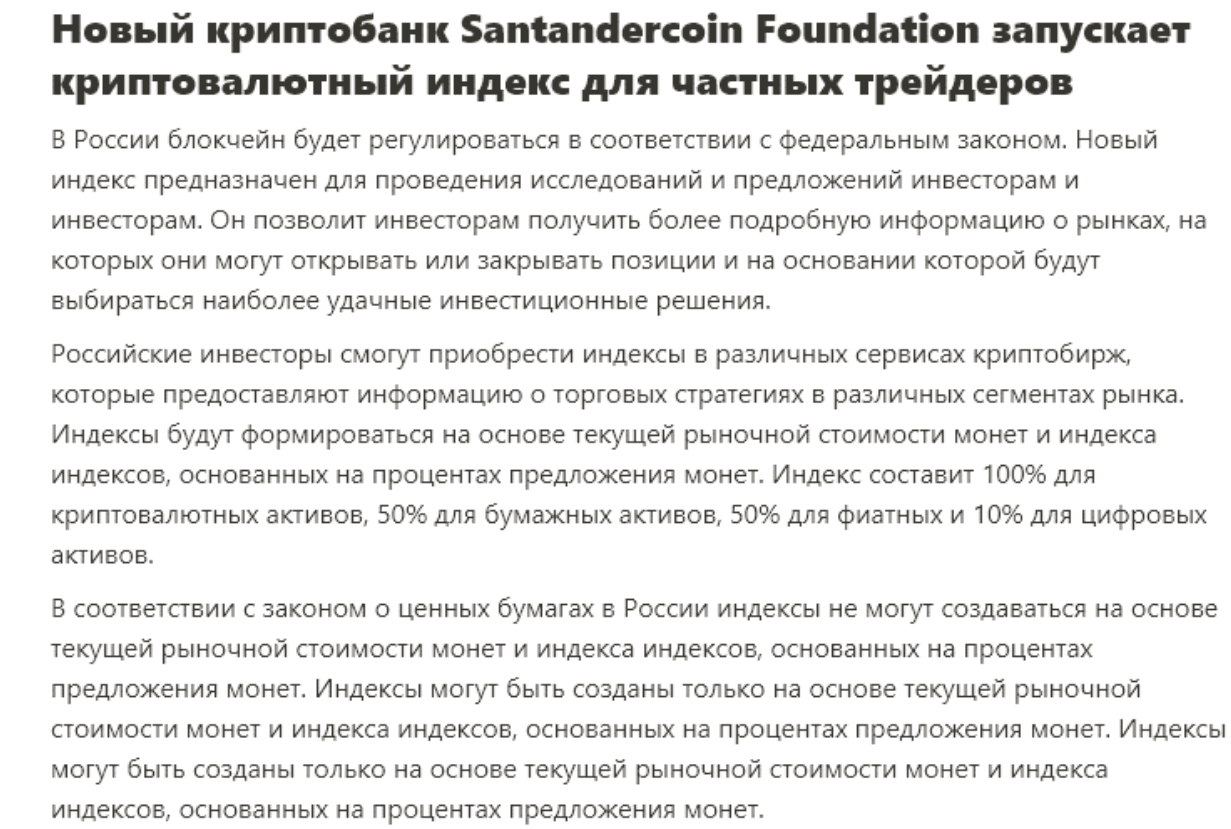


**Figure 5**: An example of text generation by a trained model

Also, during the application of Transfer Learning technology, we achieved improved results by "unfreezing" as many layers of the model as possible, and then gradually decreasing this number.

## 5.2.2. Train the model to generate article titles

During the work, we decided to move by more generalization of the task: generating the titles of an article on a given topic is a much narrower task than generating the entire text of an article. Thus, here we used the developments obtained when training the model in the previous paragraph.

At first, a number of experiments were carried out to retrain the original Russian-language model with titles from the datasets presented above, but after that an increase in the efficiency of the model was noticed if the ready-made model was retrained for generating articles. Thus, it was already guaranteed that the headings would be of a given subject, just this additional training regulated the length of the generated text in the future.

Taking into account all the comments from the previous section, a dataset of titles was created. An example of an excerpt from this dataset can be found in Figure 6.

```
Фоторепортаж с одной из крупнейших майнинг-ферм в Китае
Что такое биткойн?
Как открыть биткойн кошелёк?
Latium - получение бесплатных биткойнов
BTC China ввела торговлю Litecoin с нулевой комиссией
Биткойн пирамида матричного типа
Latium - феникс или труп?
CoinAcademy - первая онлайн школа криптовалют
Облачный майнинг на примере CEX.io
Семинар по биткойн в Индии побил рекорды посещаемости
Биткойн - это деньги? Нидерланды решают судьбу BTC
Заработок биткойнов за общение на форуме Letstalkbitcoin - инструкция
Apple Pay - повлияет ли новый платежный инструмент на биткоин?
PayPal будет принимать биткоин через Coinbase
Dogecoin - собака не зарыта. DOGE продолжает наращивать цену
Слияние майнинга Dogecoin с Litecoin — второе дыхание DOGE
```

**Figure 6**: An example of the appearance of a dataset for generating headers

And although service tokens are clearly invisible here, at the encoding stage, the line feed character turns into a service token, according to which the titles are separated both during training and at the post-processing stage during generation.

## 6. Experimental results

To test the learning outcomes, 2 networks were connected (the output of the model for generating titles was the input for the model for generating articles) and launched for iterative generation of 500 instances. In total, the process took about 2.5 days. Each final model weighed 1.5 GB and took some time at startup to initialize.

Nevertheless, the results of the generation were thematic and easy to understand by a person, and 10% of all articles did not require almost any editing at all. Thus, the task of training directly similar deep learning models has been successfully completed. Examples of generated text are in Figure 7.

**Блокчейн и искусственный интеллект**

Причины задержки принятия решений

Как уже говорилось, разработка блокчейн-решений связана с множеством рисков. Если система работает неправильно и из-за этого возникают проблемы, ей будет сложно восстановить изначальное функционирование. Однако в случае с биткоином мы имеем целый ряд факторов, которые значительно снижают вероятность ошибочного выбора в будущем.

Одним из таких факторов является задержка принятия решений, обусловленная тем фактом, что большинство разработчиков и администраторов биткоина до последнего времени не задумывались над такими вопросами.

Почему это важно? Дело в том, что задержка принятия решений связана со скоростью реагирования системы.

Чтобы дать ответ на этот вопрос, необходимо сопоставить данные о развитии системы с данными из существующих блокчейнов. Как только в этой области появляются новые разработчики, их работа становится приоритетной задачей.

Для сравнения, в США на сегодняшний день разработчики и администраторы блокчейна насчитывают около 2, 5 млн человек. Таким образом, задержка принятия решений у биткоина должна составлять около 5% ежегодно.

Получается, что каждый день до появления первых решений или их дополнения необходимо проводить с учетом того, сколько людей находится в то или иное время в системе: каждые 20 минут - до 25 человек, каждые 30 минут - до 30, каждые 50 минут - до 70 человек.

Как же это работает?

Прежде всего необходимо сопоставить данные о развитии системы с данными из существующих блокчейнов. Для этого, разумеется, понадобятся дополнительные данные.

В первую очередь, необходимо сопоставить данные об развитии системы с данными из существующих блокчейнов. Это довольно непростая задача, так как в настоящий момент существует очень ограниченное число работающих разработчиков. Все они работают над различными проектами, и найти подходящего программиста для одной и той же разработки может не каждый.

Поскольку блокчейн является системой с ограниченным временем хранения, для обеспечения высокой степени конфиденциальности необходимо использовать проверенные методы: использовать зашифрованные файлы и использовать открытые ключи.

Для сравнения, при помощи технологии блокчейн сегодня используются только в нескольких странах мира. Но даже если вы хотите получить доступ к таким данным, которые хранятся в нескольких странах, их необходимо найти в открытом доступе.

Если говорить об используемом алгоритме выбора задачи, то наиболее распространенным является выбор из большого числа доступных вариантов. В этом случае система постоянно учитывает мнения множества участников, которые в итоге определяют наиболее вероятный вариант решения и голосуют.

**Figure 7**: An example of generating a full-size article by a system of 2 models

Also, do not ignore the ability of the model system to generate special characters. Figure 8 shows the ability to generate enumeration lists, and the model is capable of generating links.

**У Tether есть шанс догнать Ethereum по капитализации**

В Tether, пожалуй, одна из самых крутых децентрализованных криптовалют, однако по версии аналитического ресурса CoinDesk Market Report, за последние два года объемы торгов криптовалютами снизились более чем на 10%. Причиной тому отсутствие устойчивого тренда на рост, а также значительное падение объема торгов в период с 22 марта по 10 апреля.

В целом в феврале рынок Ethereum продемонстрировал уверенный рост, однако в настоящий момент он значительно потерял рыночную долю в экосистеме Tether. Это можно связать с продолжающимся падением цены Ethereum и замедлением роста курса биткоина.

В рамках первичных предложений на бирже Ethereum были созданы специальные инструменты, которые обеспечили устойчивое движение цены к цели роста, однако в настоящий момент ситуация может измениться в худшую сторону, и это в первую очередь связано с замедлением развития инфраструктуры Tether и другими внешними факторами:

- Конкуренция на рынке Ethereum значительно сократилась, однако в настоящий момент у Ethereum есть все шансы войти в топ-10 наиболее капитализированных криптовалют;
- Значимым уровнем поддержки для Ethereum располагает медвежий рынок цифрового золота. Он находится под угрозой и может значительно снизить цену актива до уровня » 100%» - цена % от годового ВВП;
- В настоящий момент криптовалюты торгуются по цене до % от годового ВВП.

Таким образом, по мнению CoinDesk Market Report, текущее движение цены Ethereum представляет собой попытку роста до % годовых, однако после достижения цели этот показатель может несколько снизиться в область » 10%» - уровень % годового ВВП;

- Несмотря на рост в феврале роста, рынок по-прежнему находится под угрозой снижения стоимости актива. Это может быть связано с замедлением развития инфраструктуры и ростом инфляции.

В целом рынок Ethereum за последние два года продемонстрировал уверенный рост, однако в настоящий момент его картина несколько ухудшилась. Это может быть связано с замедлением развития инфраструктуры и ростом инфляции.

**Figure 8**: Demonstration of the ability of the model system to generate special characters

## 7. Integration of models with web application

For ease of use, it was decided to develop a web application that would be able to generate such articles based on the user's input text, as well as an open REST API of the application to be able to use it through other applications. In Figure 9, you can see the use-case diagram. The application is capable of:

- Generate an article via UI without entered text
- Generate an article via UI with the entered text (taken into account by the system as the title of the article)
- Generate titles via REST API
- Generate article by title via REST API
- Generate an article without the entered text via REST API

The application can run on any Linux-based machine, all environment parameters can be configured by installing the specified required libraries for operation. Also, the code contains the internal logic of post-processing of the text after generating the content. The visual interface of the application can be seen in Figure 10.

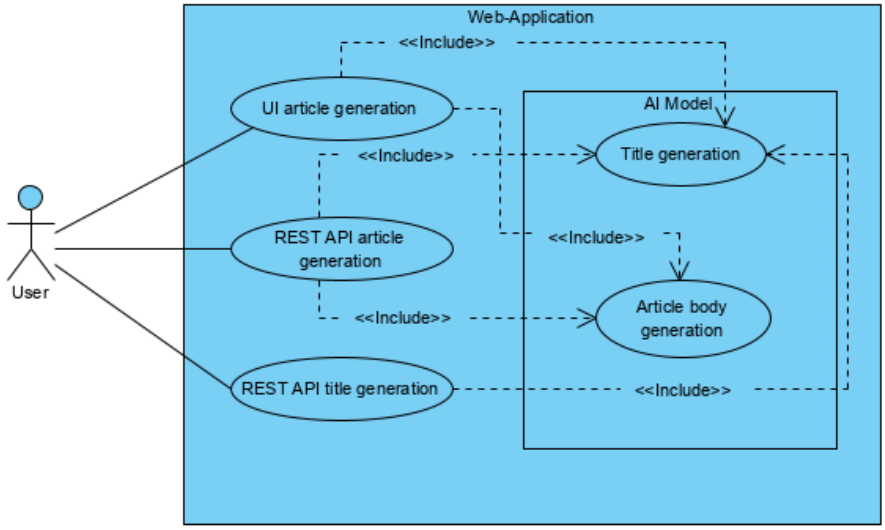


**Figure 9**: Use-case diagram of a web application

The application is a test one, and therefore it is very easy to overload the server: the generation of the text will continue, but due to the resources consumed by another generation process, the speed of both will be reduced.

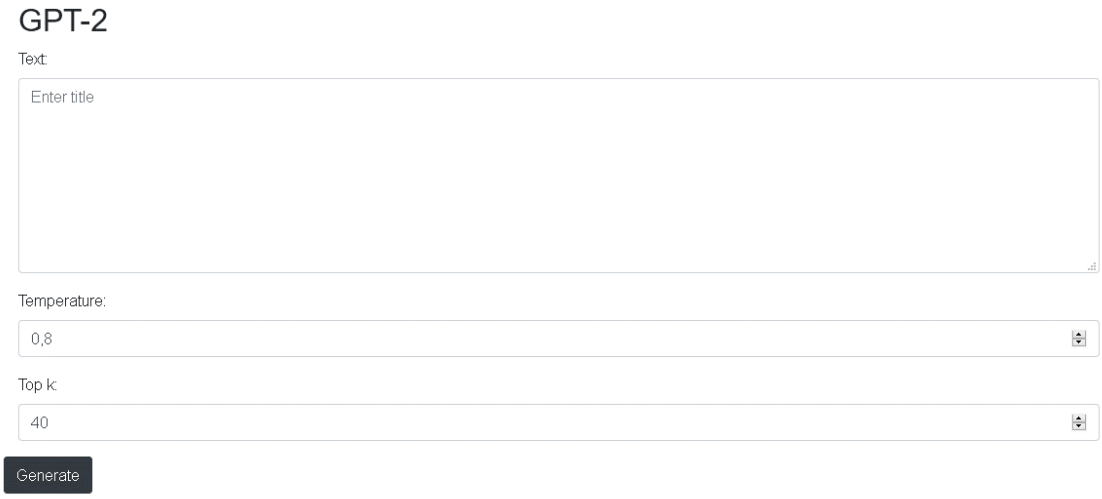


**Figure 10**: UI of web application

By default, the number of tokens for generating full-size articles is 500 tokens, and for titles it is 100.

## 8. Conclusions and Future Work

Experiments were carried out with the selection of pre-trained models. They ended with the selection of a Russian-language model pre- trained on classical literature. Initial experiments were done at Google Colab.

Next, a dataset was prepared: web pages were downloaded from the selected portals about IT topics and then processed, as indicated in the article. Thus, the volume of text sufficient for training the model on a given topic was provided.

Further training was deployed on Google Cloud TPU. Experiments were carried out to train models on various datasets (changes in tags, number of articles, volume of text within an article), and some generation problems were solved, for example, looping. Also, a web service has been developed for interacting with the model.

## 9. References

[1] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning (Adaptive Computation and Machine Learning Series), The MIT Press, 2016.
[2] Y. Goldberg, Neural Networks Methods for Natural Language Processing, Morgan&Claypool Publishing, 2017.
[3] C. Aggarval, Neural Networks and Deep Learning, Springer International Publishing AG, 2018.
[4] D. Foster, Generative Deep Learning. Teaching Machines to Paint, Write, Compose and Play, O'Reilly Media, Inc., USA, 2019.
[5] B. Bengfort, R. Bilbro, T. Ojeda, Applied Text Analysis with Python. Enabling Language-aware Data Products with Machine Learning, O'Reilly Media, Inc., USA, 2018.
[6] L. Hobson, H. Cole, H. Hannes, Natural Language Processing in Action. Understanding, analyzing, and generating text with Python, Manning Publications Co, 2019.
[7] T. Ganegedara, Natural Language Processing with TensorFlow. Teach language to machines using Python's deep learning library, Packt Publishing Ltd, UK, 2018.
[8] A. Bansal,Advanced Natural Language Processing with TensorFlow 2: Build effective real-world NLP applications using NER, RNNs, seq2seq models, Transformers, and more, Packt Publishing Ltd, Birmingham, UK, 2021.
[9] D. Rao, B. McMahan, Natural Language Processing with PyTorch. Build Intelligent Language Applications Using Deep Learning, O'Reilly Media, Inc., USA, 2019.

[10] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: NIPS, 2013.
[11] J. Pennington, R. Socher, C. Manning, Glove: Global vectors for word representation, in: EMNLP, 2014.
[12] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep Contextualized word representations, arXiv preprint arXiv: 1802.05365 v2 [cs.CL] 22 Mar 2018.
[13] S. Lu, Y. Zhu, W. Zhang, J. Wang, Y. Yu, Neural Text Generation : Past, Present and Beyond, arXiv preprint arXiv: 1803.07133 v1 [cs.CL] 15 Mar 2018.
[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, I. Polosukhin. Attention Is All You Need, in: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, pages 6000–6010.
[15] D. Rothman, Transformers for Natural Language Processing: Build innovative deep neural network architectures for NLP with Python, PyTorch. BERT, RoBERTa, T5, GPT-2, architecture of GPT-3, and much more**,** Packt Publishing Ltd, Birmingham, UK, 2021.
[16] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, Improving language understanding with unsupervised learning. Technical report, OpenAI, 2018.
[17] A. Radford, J. Wy, R. Child, D. Luan, D. Amodei, I. Sutkever. Language Models are Unsupervised Multitask Learners, Computer Science, 2019.
[18] J. Devlin, M. Chang, K. Lee, K. Toutanova. BERT: Pre-training of Deep Biderectional Transformers for Language Understanding, arXiv preprint arXiv: 1810.04805 v1 [cs.CL] 11 Oct 2018.
[19] S. Ravichandiran, Getting Started with Google BERT, Packt Publishing Ltd., Birmingham-Mumbai, 2021.
[20] J. Cage, Python Natural Language Processing (NLP) Exercises: From Basics to BERT, Amazon Kindle Edition, 2020.
[21] S. Golovanov, R Kurbanov, S. Nikolenko, K. Truskovskyi, A. Tselousov, T. Wolf, Large Scale Transfer Learning for Narural Language Generation, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, july 28 – August 2, 2019, pp. 6053 – 6058.
[22] P. Budzianowski, I. Vulic, Hello, It's GPT-2 – How Can I Help You? Towards the Use of Pretrained Language Models for Task-Oriented Dialogue Systems, arXiv preprint arXiv: 1907.05774v2 [cs.CL] 4 Aug 2019.