

# Narrabat — a Prototype Service for Stylish News Retelling

I.I. Dolgaleva <[iidolgaleva@edu.hse.ru](mailto:iidolgaleva@edu.hse.ru)>

I.A. Gorshkov <[iagorshkov@edu.hse.ru](mailto:iagorshkov@edu.hse.ru)>

R.E. Yavorskiy <[ryavorsky@hse.ru](mailto:ryavorsky@hse.ru)>

*Faculty of Computer Science, Higher School of Economics,  
20 Myasnitskaya, Moscow, 101000, Russia*

**Abstract.** Nowadays, news portals are forced to seek new methods of engaging the audience due to the increasing competition in today's mass media. The growth in the loyalty of news service consumers may further a rise of popularity and, as a result, additional advertising revenue. Therefore, we propose the tool that is intended for stylish presenting of facts from a news feed. Its outputs are little poems that contain key facts from different news sources, based on the texts of Russian classics. The main idea of our algorithm is to use a collection of classical literature or poetry as a dictionary of style. The facts are extracted from news texts through Tomita Parser and then presented in the form similar to a sample from the collection. During our work, we tested several approaches for text generating, such as machine learning (including neural networks) and template-base method. The last method gave us the best performance, while the texts generated by the neural network are still needed to be improved. In this article, we present the current state of Narrabat, a prototype system rephrasing news we are currently working on, give examples of generated poems, and discuss some ideas for future performance improvement.

**Keywords:** natural language processing; information extraction; natural language generation; tomita parser; neural networks

**DOI:** 10.15514/ISPRAS-2017-29(4)-23

**For citation:** Dolgaleva I.I., Gorshkov I.A., Yavorsky R.E. Narrabat — a Prototype Service for Stylish News Retelling. *Trudy ISP RAN/Proc. ISP RAS*, vol. 29, issue 4, 2017, pp. 325-336. DOI: 10.15514/ISPRAS-2017-29(4)-23

## 1. Introduction

### 1.1 The main idea

In the era of information explosion demand for news aggregation services is always high. Classical news services like Yandex News or Google News are on the market for a long time, but their format is too restricted to satisfy all potential audiences. The motivation for Narrabat, a new news service, is to retell news in a stylish way similar to the writings of great writers and poets so as to promote consumers loyalty and to increase the revenue of news portals, for instance, from contextual advertising.

The goal of the study is to develop a methodology of rewriting news texts in a specified style and to implement it as a service. To provide a new insight into retelling news, we build an architecture of Narrabat that is rather straightforward: retrieve news from the providers, extract facts, reproduce the facts in a new form. The realization of the proposed architecture might require handling two important issues. Firstly, it is necessary to process the news and extract the main information from it. At this point, it is essential to realize what kind of unstructured data will be marked as key information. Secondly, we need to generate text in a predefined style considering extracted key words.

To make precise the scope of the study, we explore the methods of retelling the news texts in more capturing manner and build a system that today has no parallel in the integrated marketing communications in news sphere.

The paper presents the current state of the retelling service implementation we are still working on. A well-established result is that we have constructed a prototype system that is capable of producing the poem from the news text. It is to be hoped that in the not too distant future, the findings of the current research will be applied to real regularly updated news feed as a service, possibly, as a chat-bot.

The plan of the paper is the following: in section 2 we present an algorithm for producing poems from the news. In section 3 the current results are presented. Finally, section IV describes the work still to be done.

### 1.2 Related work

Recent years have seen the rapid growth in the number of studies devoted to the extraction of information and natural language generation. Insofar as retelling news is concerned to these two subject areas, it would be wise to cover both of them in the paper.

Nowadays, state-of-the-art approaches of fact extraction go far beyond the earliest systems, where the patterns are found referring to rules of grammar [1], [2]. However, an involvement of highly qualified experts in the field or linguists is believed to be a significant drawback of these approaches. Some of them are briefly recalled in the next few paragraphs.

The next coherent idea about highlighting the facts from the text was to propose an algorithm that was able to be trained independently or "almost independently", namely, using active learning techniques [3], [4].

As the task of the researches became more complicated, and the need to distinguish an implicitly expressed meaning occurred, the aforementioned approaches lose its efficiency. And the researches shifted their attention to generative models [5] and conditional models [6].

Shedding light on the text generation approaches, the first things that arises is that text in natural language may be generated via predetermined rules [7], [8], when a set of templates is composed to map semantics to utterance. This approach is supposed to be conventional one. These systems are believed to be simple and easy to control, however, at the same time, no scalable due to limited number of rules, and, consequently, output texts.

Furthermore, utilization of statistical approaches in sentence planning are still based on hand-written text generators, whether choosing the most frequent derivation in context-free grammar [9] or maximizing the reward in reinforcement learning [10]. By the way, further researches are aimed at minimizing human participation and rely on learning sentence planning rules from labelled corpus of utterances [11], which also require a huge markup by linguists.

The next set of approaches in natural language generation is based on corpus-driven dependencies. The systems in this direction imply the construction of class-based n-gram language model [12] or phrase-based language model [13]. Moreover, a significant number of researchers utilize active learning in order to generate texts [14], [15].

The use of neural network-based approaches in natural language generation is still relatively unexplored. Although, there are studies that present the high-quality recurrent neural network-based language models [16], [17] that are able to model arbitrarily long dependencies. In addition, it is worth emphasizing that the usage of Long Short-term Memory (LSTM) network may try to solve the vanishing gradient problem [18] such as in [10].

## 2. Data and Method

### 2.1 The news sources

In this framework, we utilize short news texts that were extracted from Russian-language informational portal "Yandex.News". The collection of news consists of 330 texts on different topics, for instance, society, economy, policy, to name but a few (ultimately, 22 topics). This collection of news texts was composed of texts on diverse topics wilfully so as to consider all lexical, syntactic and morphological particularities of each of the themes in order to create universal system of text processing and generation.

Every text in the collection comprises no more than three sentences except a title. It is worth emphasizing that the format of short texts leads itself well with highlighting the main information from the text. It follows from the fact that every sentence is quite informative to extract key knowledge by means of rule-based approach.

### 2.2 Fact extraction

To provide basic information from the news, we propose to extract a kind of extended grammatical basis of the sentences. To that end, we use Tomita-parser [19] that allows to extract structured data (facts) from text in natural language. The tool is much more flexible and effective in key information detection and extraction than, for example, metric tf-idf since it allows to retrieve finite chains of words from all the positions in the sentence, not only successive words.

Open-source Tomita-parser, in contrast to similar non-commercial fact extraction software, accounts for specificity of work with the Russian language and has more or less detailed documentation. The tool was implemented by developers of Yandex on the basis of GLP-parser [20], which utilizes context-free grammars, dictionaries of keywords and interpreter.

To get a new insight into extracting the meaning of the texts, a dictionary (gazetteer) and grammar was compiled. As mentioned before, we suggest that the main idea of the sentence is fixed in common basis of the sentence, a kind of analogue of the grammatical basis. Given the opportunity to construct Russian-language sentences with the inversion, the grammar consists of the two main rules:

$$S \rightarrow \textit{Subject Predicate} \mid \textit{Predicate Subject}$$

Every non-terminal derives a string of words dependent on the root words, namely, for Subject it may be adjective and for Predicate it may be addition or adverb.

After the required string of words is found, Tomita-Parser transforms it into fact and represents it in the result collection of labelled texts, which, in turn, is prepared for text retelling.

### 2.3 Poems collection

To teach our system the poetry style we have used writings of Alexander Blok [21] and Nikolay Nekrasov [22] retrieved from Maksim Moshkov on-line library Lib.Ru [23]. We have chosen to utilize particularly these poets as their poems possess artistic and rhythmic harmony, and clearly traceable metrical feet. In further work, we plan to expand the collection of poetry by Agniya Barto, Athanasius Fet and Fedor Tyutchev.

### 2.4 Learn and produce methods

Besides the method that is described above, we tested another ways of generating word sequences, such as neural networks. For example, we trained a network with LSTM-layer which was expected to generate poems, using a huge dataset of

Pushkin's poems from [24]. (LSTM for generating poems was successfully applied in [25], [26], [27]). The result we got was a bit insufficient due to low computational power of our computer and small network size. Further implementations with additional layers increased the quality of generated poems, but it is still being trained, so we are not ready yet to present its results.

Table 1 presents the example of quatrain generated by the first version of our neural networks:

Table 1. The poem produced by neural networks

Narrabat output v.00	Ко в жаме стрък иреланье, И сталили пореланье И по почаль в сореннем По сеанно переланий.
----------------------------	--

On the Table I it could be seen that although the poem consists of non-existent Russian-language words, the strings of characters in words virtually resemble real words in their structure. The second thing to sharpen the issue addressing the table is that three out of four strings in the quatrain have the same number of syllables (while the fourth line has only one syllable less). The makings of the rhythms, as well, are evident. Given all the above, we treat the neural networks as a paramount direction for our further research.

## 2.5 Current version of the algorithm

Apart from training neural networks to generate poems, we are so far to seek the most conspicuously well-turned poem generator. To that end, we use template-base method described below.

First, in order to break words into syllables, we utilize an improved version of an algorithm of P. Hristov in the modification of Dymchenko and Varsanofiev [28] that comprises a set of syllabication rules that are applied sequentially. Then syllables of potentially matching subjects and predicates are compared using the following heuristic:

- The number of syllables must coincide.
- Vowels inside syllables have priority over consonants.
- The last syllable has priority over the other.

Search for the similar sentences returns pieces of classical writings, which are used then as templates for the resulting text generation. The output poems ought to be sought in the Section 3.

## 3. Results

Below is an example produced by current release (v.01) of our Narrabat system. We start from a news description and extract subject and predicate, see Table 2.

Table 2. Original news

Source	Общегородской субботник пройдет в следующую субботу, 15 апреля.
Extracted basis	Subject = общегородской субботник Predicate = пройдет

The same is done for all sentences in the collection, see example in Table 3.

Table 3. Original piece from the collections

Source	А виноградные пустыни, Дома и люди — всё гроба. Лишь медь торжественной латыни Поет на плитах, как труба.
Extracted basis	Subject = медь торжественной латыни Predicate = поет

The implemented similarity measure allows us to figure out that the subjects and the predicates are quite similar, see Table 4. Notice the same number of syllables and almost identical endings.

Table 4. Example of a similar pairs match

Subjects	Predicates
медь тор-жест-вен-ной ла-ты-ни	по-ет
об-ще-го-род-ской суб-бот-ник	прой-дет

Now we can replace the matching pairs, see Table 5 for an example of the resulting poem.

Table 5. The final result of the algorithm

Narrabat output v.00	А виноградные пустыни, Дома и люди — всё гроба. Лишь <b>общегородской субботник</b> <b>Пройдет</b> на плитах, как труба.
----------------------------	---

One can see that the resulting text keeps subject and predicate from the original fact and at the same time the inserted fragment smoothly fits the style of the poem and do not destroy its structure.

All readers are able to have a closer look at the details of implementation of our Narrabat system and access the source code that is open and available on GitHub [29].

#### 4. Conclusion and future research directions

In the paper we have proposed a prototype of system that is capable of retelling the news as poems that resembles style of great writers.

In the course of the work we discovered that the collection of poems have to be drastically enlarged in order to generate high-quality poems. Given the exploration, Nikolay Nekrasov has demonstrated more mapping potential in our tasks as he wrote more common sentences than Alexander Blok.

Moreover, the aforementioned metrics of mapping the subjects and predicates from news and poems does not cover all cases to be universal, for instance, in further releases it may take into account rhyme explicitly. Although the first results presented above are somehow promising, still a lot is on the to do list:

- Improve the quality fact extraction by extending the parsing rules.
- Use available dictionaries of accentuation to take into account the rhythmic structure of a sentence.
- Apply machine learning techniques to better grasp the style of a sample writing.
- Extend the algorithm to cover other parts of sentences, namely, objects and complements.

#### References

- [1]. Douglas E Appelt, Jerry R Hobbs, John Bear, David Israel, and Mabry Tyson. Fastus: A finite-state processor for information extraction from real-world text. In *IJCAI*, volume 93, pages 1172–1178, 1993.
- [2]. R Mooney. Relational learning of pattern-match rules for information extraction. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, volume 328, page 334, 1999.
- [3]. François Mairesse, Milica Gašić, Filip Jurčićek, Simon Keizer, Blaise Thomson, Kai Yu, and Steve Young. Phrase-based statistical language generation using graphical models and active learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1552–1561. Association for Computational Linguistics, 2010.
- [4]. Aidan Finn and Nicolas Kushmerick. Active learning selection strategies for information extraction. In *Proceedings of the International Workshop on Adaptive Text Extraction and Mining (ATEM-03)*, pages 18–25, 2003.
- [5]. Kristie Seymore, Andrew McCallum, and Roni Rosenfeld. Learning hidden markov model structure for information extraction. In *AAAI-99 workshop on machine learning for information extraction*, pages 37–42, 1999.
- [6]. Adwait Ratnaparkhi. Learning to parse natural language with maximum entropy models. *Machine learning*, 34(1-3):151–175, 1999.
- [7]. Adam Cheyer and Didier Guzzoni. Method and apparatus for building an intelligent automated assistant, March 18 2014. US Patent 8,677,377.

- [8]. Hugo Gonçalo Oliveira and Amílcar Cardoso. Poetry generation with poetryme. In *Computational Creativity Research: Towards Creative Machines*, pages 243–266. Springer, 2015.
- [9]. Anja Belz. Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models. *Natural Language Engineering*, 14(04):431–455, 2008.
- [10]. Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke, and Steve Young. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. *arXiv preprint arXiv:1508.01745*, 2015.
- [11]. Amanda Stent and Martin Molina. Evaluating automatic extraction of rules for sentence plan construction. In *Proceedings of the SIGDIAL 2009 Conference: The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 290–297. Association for Computational Linguistics, 2009.
- [12]. Adwait Ratnaparkhi. Trainable approaches to surface natural language generation and their application to conversational dialog systems. *Computer Speech & Language*, 16(3):435–455, 2002.
- [13]. François Mairesse and Steve Young. Stochastic language generation in dialogue using factored language models. *Computational Linguistics*, 2014.
- [14]. Gabor Angeli, Percy Liang, and Dan Klein. A simple domain-independent probabilistic approach to generation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 502–512. Association for Computational Linguistics, 2010.
- [15]. Ravi Kondadadi, Blake Howald, and Frank Schilder. A statistical nlg framework for aggregated planning and realization. In *ACL (1)*, pages 1406–1415, 2013.
- [16]. Tomáš Mikolov, Stefan Kombrink, Lukáš Burget, Jan Černocký, and Sanjeev Khudanpur. Extensions of recurrent neural network language model. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pages 5528–5531. IEEE, 2011.
- [17]. Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernocký, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Interspeech*, volume 2, page 3, 2010.
- [18]. Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.
- [19]. Yandex LLC. Tomita-parser tool to extract structured data from texts. <https://tech.yandex.ru/tomita/>. Accessed: 2017-04-10.
- [20]. Masaru Tomita. Lr parsers for natural languages. In *Proceedings of the 10th International Conference on Computational Linguistics and 22nd annual meeting on Association for Computational Linguistics*, pages 354–357. Association for Computational Linguistics, 1984.
- [21]. Alexander Blok. *Sobranie sochinenij v 8 tomah [Collected Works in 8 volumes]*. Gosudarstvennoe izdatel'stvo hudozhestvennoj literatury [State Publishing House of Fiction], Moscow, 1960-1963 (in Russian).
- [22]. Nikolaj Nekrasov. *Polnoe sobranie stihotvorenij N. A. Nekrasova v 2 tomah [Complete collection of poems by N.A. Nekrasov in 2 volumes]*. Tipografija A. S. Suvorina [Printing house of AS Suvorin], Sankt-Peterburg, 1899 (in Russian).
- [23]. Lib.ru: Library of Maksim Moshkov. <http://lib.ru/>. Accessed: 2017-04-10.

- [24]. Alexander Pushkin. *Sobranie sochinenij v desyati tomah. Tom vtoroj. Stihotvoreniya 1823-1836* [Collected works in ten volumes. Volume 2. Poems of 1823-1836]. [State Publishing House of Fiction], Moscow, 1959—1962 (in Russian).
- [25]. Anna Rumshisky, Peter Potash, Alexey Romanov. Ghostwriter: Using an lstm for automatic rap lyric generation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1919–1924, 2015.
- [26]. Rui Yan. i, poet: Automatic poetry composition through recurrent neural networks with iterative polishing schema. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*, pages 2238–2244, 2016.
- [27]. Yejin Choi, Marjan Ghazvininejad, Xing Shi and Kevin Knight. Generating topical poetry. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1183–1191, 2016.
- [28]. Yura Batora. Algorithm for splitting words into syllables. <https://sites.google.com/site/fofantapp/project-updates/hyphenation>. Accessed: 2017-04-10.
- [29]. Rostislav Yavorskiy, Irina Dolgaleva, Ilya Gorshkov. Narrabat. <https://github.com/onobot/allbots/tree>. Accessed: 2017-04-10.

## Narrabat — прототип сервиса для пересказа новостей в формате стихотворений

И.И. Долгалева <[iidolgaleva@edu.hse.ru](mailto:iidolgaleva@edu.hse.ru)>

И.А. Горшков <[iagorshkov@edu.hse.ru](mailto:iagorshkov@edu.hse.ru)>

Р.Э. Яворский <[ryavorsky@hse.ru](mailto:ryavorsky@hse.ru)>

Факультет компьютерных наук, Высшая Школа Экономики,  
101000, Россия, Москва, ул. Мясницкая, 20

**Аннотация.** В интернете все большую популярность приобретают СМИ, отказывающиеся от общепринятого формального способа изложения новостей и делающие акцент на креативности предоставляемого контента. Яркими примерами могут послужить паблик "Лентач" из социальной сети "ВКонтакте", сопровождающий каждую новость мемами, и ресурс "КАКТАМ?", оборачивающий заголовки в намеренно сверхэмоциональную форму. Мы решили реализовать инструмент Narrabat, пересказывающий новости в еще одном необычном стиле. Его задача - преобразовывать новостные ленты, взятые из сторонних источников, в небольшие стихотворения, отражающие ключевые события новостных сюжетов. В качестве основы для генерации стихов используется большая коллекция русской классики (состоящая из, к примеру, произведений Блока и Некрасова). Одним из главных достоинств выбранной нами формы пересказа и созданного инструмента в частности является то, что, при всей оригинальности вывода, процесс его генерации полностью автоматизирован, в отличие от сервисов, описанных выше. Инструмент работает в несколько этапов: сначала происходит выделение фактов из заголовков выгруженных новостей при помощи Tomita Parser, после чего факты передаются в модуль, отвечающий за генерацию стихотворения. По ходу работы мы использовали несколько подходов для генерации стихотворений, такие, как алгоритмы, построенные на правилах, и машинное обучение, включая нейронные сети. На данном этапе наилучший результат дал первый метод, однако работа по обучению нейронной сети ведется до сих пор. В данной статье мы опишем текущие результаты работы, приведем примеры сгенерированных стихотворений, а также перечислим направления для дальнейшего улучшения инструмента.

**Ключевые слова:** обработка естественного языка; извлечение информации; генерация текста; томиита парсер; нейронные сети

**DOI:** 10.15514/ISPRAS-2017-29(4)-23

**Для цитирования:** Долгалева И.И., Горшков И.А., Яворский Р.Э. Narrabat — прототип сервиса для пересказа новостей в формате стихотворений. *Труды ИСП РАН*, том 29, вып. 4, 2017 г., стр. 325-336 (на английском языке). DOI: 10.15514/ISPRAS-2017-29(4)-23

## Список литературы

- [1]. Douglas E Appelt, Jerry R Hobbs, John Bear, David Israel, and Mabry Tyson. Fastus: A finite-state processor for information extraction from real-world text. In *IJCAI*, volume 93, pages 1172–1178, 1993.
- [2]. R Mooney. Relational learning of pattern-match rules for information extraction. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, volume 328, page 334, 1999.
- [3]. François Mairesse, Milica Gašić, Filip Jurčićek, Simon Keizer, Blaise Thomson, Kai Yu, and Steve Young. Phrase-based statistical language generation using graphical models and active learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1552–1561. Association for Computational Linguistics, 2010.
- [4]. Aidan Finn and Nicolas Kushmerick. Active learning selection strategies for information extraction. In *Proceedings of the International Workshop on Adaptive Text Extraction and Mining (ATEM-03)*, pages 18–25, 2003.
- [5]. Kristie Seymore, Andrew McCallum, and Roni Rosenfeld. Learning hidden markov model structure for information extraction. In *AAAI-99 workshop on machine learning for information extraction*, pages 37–42, 1999.
- [6]. Adwait Ratnaparkhi. Learning to parse natural language with maximum entropy models. *Machine learning*, 34(1-3):151–175, 1999.
- [7]. Adam Cheyer and Didier Guzzoni. Method and apparatus for building an intelligent automated assistant, March 18 2014. US Patent 8,677,377.
- [8]. Hugo Gonçalo Oliveira and Amílcar Cardoso. Poetry generation with poetryme. In *Computational Creativity Research: Towards Creative Machines*, pages 243–266. Springer, 2015.
- [9]. Anja Belz. Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models. *Natural Language Engineering*, 14(04):431–455, 2008.
- [10]. Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke, and Steve Young. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. *arXiv preprint arXiv:1508.01745*, 2015.
- [11]. Amanda Stent and Martin Molina. Evaluating automatic extraction of rules for sentence plan construction. In *Proceedings of the SIGDIAL 2009 Conference: The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 290–297. Association for Computational Linguistics, 2009.
- [12]. Adwait Ratnaparkhi. Trainable approaches to surface natural language generation and their application to conversational dialog systems. *Computer Speech & Language*, 16(3):435–455, 2002.
- [13]. François Mairesse and Steve Young. Stochastic language generation in dialogue using factored language models. *Computational Linguistics*, 2014.
- [14]. Gabor Angeli, Percy Liang, and Dan Klein. A simple domain-independent probabilistic approach to generation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 502–512. Association for Computational Linguistics, 2010.
- [15]. Ravi Kondadadi, Blake Howald, and Frank Schilder. A statistical nlg framework for aggregated planning and realization. In *ACL (1)*, pages 1406–1415, 2013.
- [16]. Tomáš Mikolov, Stefan Kombrink, Lukáš Burget, Jan Černocký, and Sanjeev Khudanpur. Extensions of recurrent neural network language model. In *Acoustics, Speech and Signal*

- Processing (ICASSP), 2011 IEEE International Conference on, pages 5528–5531. IEEE, 2011.
- [17]. Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernocký, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Interspeech*, volume 2, page 3, 2010.
- [18]. Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.
- [19]. Yandex LLC. Томита-парсер. <https://tech.yandex.ru/tomita/>. Дата обращения 10.04.2017.
- [20]. Masaru Tomita. Lr parsers for natural languages. In *Proceedings of the 10th International Conference on Computational Linguistics and 22nd annual meeting on Association for Computational Linguistics*, pages 354–357. Association for Computational Linguistics, 1984.
- [21]. Александр Блок. Собрание сочинений в 8 томах. Государственное издательство художественной литературы, Москва, 1960-1963.
- [22]. Николай Некрасов. Полное собрание стихотворений Н.А. Некрасова в 2 томах. Типография А. С. Суворина, Санкт-Петербург, 1899.
- [23]. Lib.ru: Библиотека Максима Мошкова. <http://lib.ru/>. Дата обращения 10.04.2017. .
- [24]. Александр Пушкин. Собрание сочинений в десяти томах. Том второй. Стихотворения 1823-1836. 1823-1836.
- [25]. Anna Rumshisky, Peter Potash, Alexey Romanov. Ghostwriter: Using an lstm for automatic rap lyric generation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1919–1924, 2015.
- [26]. Rui Yan. i, poet: Automatic poetry composition through recurrent neural networks with iterative polishing schema. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*, pages 2238–2244, 2016.
- [27]. Yejin Choi, Marjan Ghazvininejad, Xing Shi and Kevin Knight. Generating topical poetry. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1183–1191, 2016.
- [28]. Yura Batora. Algorithm for splitting words into syllables. <https://sites.google.com/site/fofantapp/project-updates/hyphenation>. Дата обращения 10.04.2017.
- [29]. Rostislav Yavorskiy, Irina Dolgaleva, Илья Горшков. Narrabat. <https://github.com/onobot/allbots/tree>. Дата обращения 10.04.2017.