Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference “Dialogue 2020”

Moscow, June 17–20, 2020

# PRETRAINING AND AUGMENTATION IN NAMED ENTITY RECOGNITION TASK FOR CYBERSECURITY DOMAIN IN RUSSIAN[1]

**Tikhomirov M. M.** (tikhomirov.mm@gmail.com),
**Loukachevitch N. V.** (louk_nat@mail.ru),
**Sirotina A. Yu.** (overnastuhed@yandex.ru),
**Dobrov B. V.** (dobrov_bv@mail.ru)

Lomonosov Moscow State University, Moscow, Russia

The paper presents the results of applying the BERT representation model in the named entity recognition task for the cybersecurity domain in Russian. Several variants of the model were investigated. The best results were obtained using the BERT model, trained on the target collection of information security texts. This model achieved results, which were 15 percentage points of F1-macro measure greater than results of CRF, the best method in previous experiments for the same task and data. We also explored a new form of data augmentation for the task of named entity recognition.

**Key words:** Cybersecurity, Named entity recognition, Pretraining, Augmentation

**DOI:** 10.28995/2075-7182-2020-19-724-735

[1] The research was supported by Russian Science Foundation (project No. 20-11-20166). Computational experiments were carried out using the equipment of the shared research facilities of HPC computing resources at Lomonosov Moscow State University.

# ПРЕДОБУЧЕНИЕ И АУГМЕНТАЦИЯ В ЗАДАЧЕ ИЗВЛЕЧЕНИЯ ИМЕНОВАННЫХ СУЩНОСТЕЙ ПО ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

**Тихомиров М. М.** (tikhomirov.mm@gmail.com),
**Лукашевич Н. В.** (louk_nat@mail.ru),
**Сиротина А. Ю.** (overnastuhed@yandex.ru),
**Добров Б. В.** (dobrov_bv@mail.ru)
МГУ имени М. В. Ломоносова, Москва, Россия

## 1. Introduction

Automatic named entity recognition (NER) is one of the basic tasks in natural language processing. The NER methods are usually tested on well-known datasets such as CONLL-2003 for English and some other European languages [18]. For Russian, such known datasets are Gareev's dataset [7], Persons-1000 [21], Collection3 [16], FactRuEval [2]. The majority of well-known datasets consist of news documents with three types of named entities labeled: person (people's names), organization (names of organizations), location (places, mostly geographical objects). For these types of named entities, the state-of-the-art NER methods usually give impressive results.

Nevertheless, if some other types of texts are being processed or some other types of named entities are being extracted, various difficulties arise. In such cases, one has to establish new principles of annotation and to ensure that these principles are applied consistently. However, even this being done, one can still face such a problem as insufficient amount of entities of a certain type, which leads to decrease of recognition quality.

In this paper we discuss the NER task in the cybersecurity domain [19]. Several additional types of named entities for this domain were annotated if compared to general datasets such as software programs, devices, technologies, hackers, and malicious programs (vulnerabilities). The most important entities for this domain are names of malicious software and hackers. However, the annotated dataset contains a modest number of entities of these types. This could be explained by the fact that usually names of viruses and hackers are not known at the time of an attack and are revealed later.

To improve NER quality in such conditions, we suggest using BERT transformers [5] as well as an automatic dataset augmentation method, by which we mean extending a training dataset with sentences containing automatically labeled named entities.

Our paper's contribution is as follows:

- We study how quality of a NER system changes depending on variants of the BERT model used. We experimented with the following models: a multilingual model, a model fine-tuned on Russian data, and a model fine-tuned on cybersecurity texts. We compare these results with the CRF-model that previously achieved the best performance on the cybersecurity dataset.

- We introduce a new method of dataset augmentation for NER tasks and study the parameters of the method.

The remainder of the paper is organized as follows. **Section 2** presents related work. **Section 3** describes the labeled data in the cybersecurity domain used in the study. **Section 4** presents the BERT-based models and the augmentation approach specially intended for NER tasks. Section 5 describes the results of the experiments.

## 2. Related Works

### 2.1. Named Entity Recognition in Information-Security Domain

The information extraction task in cybersecurity domain has been discussed in several works. However, most works consider information extraction only from structured or semi-structured English texts. For instance, Bridges at al. [3] used training corpora consisting of Microsoft Bulletins and National Vulnerability Database descriptions mainly. The training corpus presented in [9] does contain unstructured blog posts, but those comprise less than 10% of the corpus.

The proposed NER systems are based on such methods as principle of Maximum Entropy [3], Conditional Random Fields (CRF) [12], [9]. Gasmi et al. [6] explored two different NER approaches: the CRF-model and a neural network (NN) based model LSTM-CRF (as suggested by Lample et al. [13]). The NN-based model combined bidirectional LSTM, the word2vec representation as a source of pre-trained word embeddings and CRFs as an output layer.

In [19], the Sec_col[2] cybersecurity corpus for Russian named entity recognition was described. The corpus contains unstructured texts, it was collected from journal articles, news reports, and forum posts. All these data can provide additional details on cybersecurity problems. The authors compared different models for cybersecurity NER including CRF and several variants of neural networks.

### 2.2. Using BERT in Named Entity Recognition

The state-of-the-art models for named entity recognition utilize various contextualized vector representations. One such a popular model is BERT [5]. BERT is an implementation of a statistical language model based on deep neural networks; the task of the BERT pretraining is to predict the word in a given place in the text. The BERT architecture consists of a 12-layer transformer-encoder that forms contextualized token representations, thus converting a sequence of tokens into a sequence of vectors.

Using BERT made it possible to achieve better results in various natural language processing tasks [5], including named entity recognition. Such results are due to the high information content of vector representations, which, unlike static vector representations, such as word2vec [14], depend on the context. In addition, an important

[2] https://github.com/LAIR-RCC/InfSecurityRussianNLP

point is the use of transfer learning techniques. BERT is pretrained on a large amount of unlabeled data on the language modeling tasks, and then it is finetuned for a specific task.

Initially, BERT is multilingual, trained on multilingual data. The paper [11] describes an approach to further training of the multilingual model on the Russian-language data (Russian Wikipedia and the Russian news corpus). The new model, called RuBERT, showed an improvement in quality in three NLP tasks in Russian in comparison with previous results and multilingual BERT. The use of RuBERT in the NER task on the Russian dataset Collection3 [16] also gave a significant improvement [4].

In 2019, the named entity recognition shared task for Slavic languages was organized [17]. Most participants and the winner used BERT as the main model. An interesting detail of this competition was that there was a significant imbalance among the types of entities in the data. For example, the entity type “product” (PRO) was annotated only for 8% of all entities in the Russian data. The results of extracting this type of entities were significantly lower than for other entities, which raises the question of improving the quality of rare entity recognition in unbalanced datasets.

### 2.3. Methods of Data Augmentation

Methods of data augmentation for natural language processing are mainly discussed for such tasks as machine translation and automatic text classification. The simplest augmentation method is to replace source words with their synonyms from manual thesauri (for example, WordNet [15]) or with words that are close to the source words according to a distributional model trained on a large text collection [24].

In [10] it was claimed that synonyms may not fit into the context, therefore the replacement words should be those that are the most probable according to a language model.

The authors of [22] used four simple augmentation techniques for the classification tasks: replacing words with their synonyms (WordNet), occasional word insertion, occasional word deletion and occasional word order changing. This method was applied to five datasets for the text classification task. Quality evaluation was presented for RNN and CNN neural networks . The average improvement of 0.8% for F-score was achieved. The study showed that all four operations contributed to the obtained improvement.

In this paper we discuss a specialized method of data augmentation for named entity recognition. We obtain additional annotated data by inserting named entities in appropriate sentences and contexts.

## 3. Data

We use a renewed version of Sec_col corpus [19] as a training dataset for the NER task. The final corpus contains 861 unstructured texts (more than 400,000 tokens), which are posts and comments extracted from several sources on cybersecurity.

The set of corpus labels includes four general types: PER (for persons excluding hackers), ORG (for organizations excluding hacker groups), LOC, and EVENT;

and five domain-specific types such as PROGRAM (for computer programs excluding malware), DEVICE (for various electronic devices), TECH (for technologies having proper names), VIRUS (for malware and vulnerabilities), and HACKER (for single hackers and hacker groups). The corpus was pre-annotated automatically, then multi-pass manual annotation took place. The annotation principles are described in detail in [19]. The quantitative characteristics for each tag are presented in **Table 1**.

**Table 1:** Tag distribution

| Type on entity | Description | Number of entities |
|---|---|---:|
| ORG | organizations (not including hacker groups) | 3,791 |
| PRO-GRAM | programmes (software products and their parts: codes, procedures) | 3,497 |
| TECH | technologies (named methods and approaches) | 2,962 |
| LOC | locations (geographical locations) | 1,376 |
| PER | persons (names of people that are not hackers) | 1,015 |
| DEVICE | devices (various electronic devices and computer programs) | 539 |
| VIRUS | viruses (malicious software and vulnerabilities) | 480 |
| EVENT | events | 301 |
| HACKER | hackers (individual hackers and hacker groups) | 60 |

According to the table, one of the labels, HACKER, is severely underrepresented in the dataset. One of the reasons for that could be that at the time when an attack happens, hackers are unknown, therefore their names are not mentioned. Another important type of label, VIRUS, is represented better than HACKER, but its frequency is still lower than for the other tags.

## 4. Models Used in Cybersecurity NER

### 4.1. BERT Models

As part of this study, we evaluated BERT in the cybersecurity NER task with the following pretrained weights:

- multilingual-bert-base model (BERT),
- model trained on Russian general data RuBERT,
- RuCyBERT, which was obtained by additional training RuBERT on information-security texts.

Training RuCyBERT was similar to training RuBERT [11], but without creating a new vocabulary. To do this, the pretraining procedure was launched on 500K cybersecurity texts with the initialization of all weights from RuBERT. The training lasted 500k steps with batch size 6.

All three models have the same architecture: transformer-encoder [20] with 12 transformer blocks, 12 self-attention heads and $H = 768$ hidden size. The models are fine-tuned for 6 epochs, with B = 16 batch size, with learning rate 5e-5 and T = 128 maximum sequence length. When forming input for the model, only the first token of a word gets a real word label, the remaining tokens get a special label X. At the prediction step, the predicted label of the first token is chosen for the whole word.

## 4.2. Training Data Augmentation

The important classes of named entities in the cybersecurity domain are names of viruses and hackers (including hacker groups). The Sec_col collection, however, includes a quite small number of hackers' names. This could be due to the fact that names of many hackers and hacker groups are not known, therefore many texts related to cybersecurity include only unnamed descriptors (such as *hacker, hacker group, hacker community*).

Analysis of some extra texts revealed that additional manual annotation is not a reasonable solution to the problem, as most texts mention almost the same well-known hacker groups and their attacks. During the analysis, a new augmentation approach for the NER task was proposed. The core idea of the NER augmentation is as follows: in most contexts where an entity descriptor is mentioned, some other variants of mentions are possible. For Russian, such variants can be: 1) a descriptor followed by a name or 2) just the name alone. The first above-indicated variant of entity mentioning is language-specific, depends on language-specific grammar rules. Consequently, we could augment the collection by adding names after descriptors or by replacing descriptors with names.

Tables 2 and 3 show the examples of the proposed augmentation procedure in English translation. In the first pair of sentences, the descriptors were replaced with the names; in the second pair of sentences, the names were inserted after the descriptors *хакер ('hacker', hacker)* and *зловред ('zlovred', malware)*. It should be noted that the sentences are given in translation into English, and for English the correct insertion of a name is before a descriptor. In parentheses, we give fragments with initial Russian augmentation.

**Table 2:** Augmentation examples for HACKER

| Original | Modified |
|---|---|
| Replacement | |
| The absence of vulnerabilities on the site and its willingness to resist **hacker** attacks is an important issue, but often stubbornly ignored by site owners. | The absence of vulnerabilities on the site and its willingness to resist **Pwn2Own** attacks is an important issue, but often stubbornly ignored by site owners. |
| Insertion | |
| And the number of installed software protection tools against **hackers** is lower—71% of those who installed a firewall. | And the number of installed software protection tools against **Sandworm hackers** (*хакеры Sandworm* in Russian) is lower—71% of those who installed a firewall. |

**Table 3:** Augmentation examples for VIRUS

| Original | Modified |
|---|---|
| Replacement | |
| Almost 30% are seriously concerned about this issue, another 25% believe that the danger of **spyware** is exaggerated, and more than 15% do not consider this type of threat to be a problem at all. | Almost 30% are seriously concerned about this issue, another 25% believe that the danger of **Remcos** is exaggerated, and more than 15% do not consider this type of threat to be a problem at all. |
| Insertion | |
| The **malware** described above is unique and can create big problems for both an individual and the whole company. | The **Locker malware** (*Зловред Locker* in Russian) described above is unique and can create big problems for both an individual and the whole company. |

The suggested augmentation includes two subtypes: inner and outer. The inner augmentation involves extracting sentences that contain relevant descriptors within the existing training data. If a sentence meet augmentation restrictions, then the descriptor is replaced with a name or a name is added after the descriptor with equal probability. In both cases, we require that the descriptor must not be followed by a labeled named entity and it must not be preceded by words that agree with the descriptor in gender, number or case, such as adjectives, participles, ordinal numbers, and others.

For the outer augmentation, we look for sentences with relevant descriptors in a collection of unannotated cybersecurity texts. There also must not be any evident named entities (words starting with a capital letter) in a window of certain width around the descriptor. As for this purpose an unannotated collection is used, we do not know the classes of potential named entities, thus we have to exclude sentences with such entities. Besides, we also require the absence of adjectives before the descriptor. The selected sentences also undergo the procedure of inserting a name after a descriptor or replacing the descriptor with a name with equal probability.

The augmentation has been implemented for two types of named entities: malicious software (VIRUS label) and hackers (HACKER label). 24 virus descriptors and 6 hacker descriptors were used. By means of inner augmentation, 262 additional annotated sentences for viruses and 165 annotated sentences for hackers were created.

The outer augmentation can be of an unlimited size. In this paper we study how the size of the outer augmentation affects the NER quality.

Inserted named entities are obtained in the following way. We took a large cybersecurity text collection and used it to extract names and sequences of names that follow target descriptors. We created the frequency list of extracted names and chose those names for which frequency was higher than a certain threshold (5). Then we excluded the names that appeared in the annotated training collection and belonged to classes that are different from the target class. The rest of the names were randomly used for insertion into the augmented sentences.

## 5. Experiments

We compare several variants of the BERT model on the NER task for information security domain. In addition, the results of using augmentation of the labeled data are investigated.

The CRF method was chosen as the baseline model for comparison, since in previous experiments with the Sec_col collection, this method showed better results than several variants of neural networks that are usually used for the NER task (BiLSTM with character embeddings) [19]. The CRF model utilizes the following features: token embeddings, lemma, part of speech, vocabularies of names and descriptors, word clusters based on their distributional representation, all these features in window 2 from the current token, tag of the previous word. The detailed description of CRF features, vocabularies, and implementation is given in [19]. Also, for comparison, the LSTM-CRF model based on Flair[3] realization was added. The LSTM-CRF model used fasttext embeddings[4] and the first capital letter feature for training.

Table 4 shows the classification results for four models for all labels used, as well as the averaged macro and micro F-measures. It can be seen that the use of the multilingual-bert-base (BERT in the table) gives better results than the CRF model for all types of named entities. The use of the pretrained models on the Russian data (RuBERT) and information security texts (RuCyBERT) gives a significant improvement over previous models. The LSTM-CRF model with the described features showed weak results, therefore, did not participate in further experiments.

**Table 4:** Results of basic models

| | LSTM-CRF | CRF | BERT | RuBERT | RuCyBERT |
|---|---|---|---|---|---|
| DEVICE | 13.92 | 31.78 | 34.04 | 43.13 | **46.77** |
| EVENT | 28.79 | 42.70 | 60.38 | 64.49 | **67.86** |
| HACKER | 5.70 | 26.58 | 42.69 | 52.43 | **61.03** |
| LOC | 83.10 | 82.30 | 90.00 | **91.28** | 90.01 |
| ORG | 62.82 | 68.15 | 76.10 | **78.95** | 78.58 |
| PER | 58.71 | 67.10 | 80.99 | 84.32 | **84.56** |
| PROGRAM | 44.22 | 62.15 | 63.15 | 64.77 | **66.57** |
| TECH | 47.14 | 60.65 | 67.08 | 67.60 | **69.24** |
| VIRUS | 14.39 | 40.90 | 40.21 | 46.92 | **54.72** |
| F-micro | 53.12 | 63.95 | 69.37 | 71.61 | **72.74** |
| F-macro | 39.87 | 53.59 | 61.63 | 65.99 | **68.82** |
| F-macro std | 2.63 | — | 1.52 | 0.93 | **0.86** |

Since models based on neural networks due to random initialization can give slightly different results from run to run, the results in the tables for all BERT models are given as averaging of four runs. The last row of Table 4 indicates (F-macro std)

[3] https://github.com/flairNLP/flair

[4] araneum_none_fasttextcbow_300 from https://rusvectores.org/ru/models/

the standard deviation of the results from the mean. It can be seen that the better the model fits the data, the better the results are, and the standard deviation decreases.

The following tables show the use of the proposed data augmentation approach to extract two types of named entities HÀCKER and VIRUS with inner and outer augmentations. For the outer augmentation, options for adding 100, 200, 400, 600 augmented sentences for each entity types (HÀCKER and VIRUS) were considered. However, the outer augmentation of 600 sentences gave a stable decrease in the results for all models, and therefore these results are not given in the tables. The "mean F1" column shows the averaging of the values of the F1 measure over all types of entities. The best achieved results are in bold. The results improving the basic results (without augmentation) are underlined.

Table 5 presents the results of applying augmentation to the CRF model. All types of the augmentation improved the results of extracting target entities. The best augmentation was inner augmentation, which gave an increase in the average quality of extracting named entities HACKER and VIRUS named entities by 10 percentage points (almost a third). Macro F1 measure for all types of entities (last column) is also significantly improved.

**Table 5:** CRF with augmentation

| | HACKER_VIRUS | | | macro |
|---|---|---|---|---|
| | P | R | F1 | F1 |
| base (no augmentation) | **66.31** | 24.21 | 33.73 | 53.59 |
| inner | 42.08 | **47.31** | **43.58** | **57.39** |
| outer 100 | 47.36 | 32.63 | 38.20 | 54.98 |
| outer 200 | 48.12 | 35.36 | 40.21 | 55.18 |
| outer 400 | 40.58 | 35.27 | 36.97 | 54.21 |

Table 6 shows the performance of the augmentation procedure for the multilingual BERT base. The table shows how unstable the multilingual BERT model behaves, demonstrating a very high standard deviation on the two types of entities that interest us. Any variant of augmentation reduces the standard deviation, which, however, remains quite high (column F1 std). Two models of outer augmentation increase the quality of extraction of target entities while significantly reducing the standard deviation compared to the original model.

**Table 6:** BERT with augmentation

| | HACKER_VIRUS | | | | macro | |
|---|---|---|---|---|---|---|
| | P | R | F1 | F1 std | F1 | F1 std |
| base (no augmentation) | **46.43** | 38.14 | 41.45 | 7.23 | 61.63 | 1.52 |
| inner | 36.81 | 45.44 | 39.92 | 3.53 | 61.26 | 0.86 |
| outer 100 | 39.13 | 44.96 | 41.04 | **2.18** | 62.02 | **0.55** |
| outer 200 | 39.32 | 48.24 | **42.51** | 4.33 | **62.21** | 0.74 |
| outer 400 | 40.23 | 45.97 | **42.53** | 4.59 | 62.12 | 1.08 |

Table 7 presents the results of the RuBERT model, trained on the Russian data. The results are significantly higher than for the previous model, the standard deviation is lower. And in this model, the augmentation in all cases reduces the standard deviation of F measures for target and all types of entities. The results on the target entities increased with outer augmentation of 200 sentences for both entities. Also, for some reason, the outer augmentation only with viruses positively influenced the extraction of both of them (100 and 200 sentences). The study of this phenomenon is planned to continue.

**Table 7:** RuBERT with augmentation

| | HACKER_VIRUS | | | | macro | |
|---|---|---|---|---|---|---|
| | P | R | F1 | F1 std | F1 | F1 std |
| base (no augmentation) | 53.65 | 47.38 | 49.67 | 4.65 | 65.99 | 0.93 |
| inner | 45.01 | **55.74** | 48.87 | 3.48 | 65.92 | 0.68 |
| outer 100 | 47.46 | 53.29 | 49.38 | 3.10 | 65.88 | 0.79 |
| outer 200 | 47.83 | 55.34 | 50.71 | 2.96 | 66.24 | **0.59** |
| outer 400 | 45.57 | 53.45 | 48.46 | **2.36** | 65.77 | 0.67 |
| outer viruses 100 | **57.14** | 51.67 | **53.79** | 3.05 | **66.85** | 0.64 |
| outer viruses 200 | 55.33 | 52.55 | 53.34 | 3.90 | 66.68 | 0.77 |

Table 8 presents the results of the RuCyBERT model, trained on the information-security texts. The basic quality of this model is much higher, and there is no improvement from the augmentation. The augmentation on average reduces the standard deviation of F-measure, which leads to the fact that the performance of models with augmentation and the basic model is comparable.

It can be also seen from Tables 5–8 that in almost all experiments the proposed augmentation significantly increases recall, but decreases precision.

**Table 8:** RuCyBERT augmentation

| | HACKER_VIRUS | | | | macro | |
|---|---|---|---|---|---|---|
| | P | R | F1 | F1 std | F1 | F1 std |
| base (no augmentation) | **61.33** | 55.89 | 57.87 | 3.75 | 68.82 | 0.86 |
| inner | 52.51 | **62.57** | 56.03 | 2.54 | 68.61 | 0.53 |
| outer 100 | 50.78 | 59.69 | 53.79 | 2.36 | 67.78 | **0.43** |
| outer 200 | 52.82 | 59.61 | 54.82 | 3.94 | 68.06 | 0.74 |
| outer 400 | 52.42 | 61.31 | 55.64 | **2.16** | 67.93 | 0.71 |

## 6. Conclusion

In this paper we present the results of applying BERT to named entity recognition for cybersecurity Russian texts. It is shown that the multilingual model performs better than the CRF model, which uses a substantial number of token features. Further tuning of the model (first on the Russian data and then on the cybersecurity collection) has significantly improved the NER quality. The highest macro F-score shown by BERT model (RuCyBERT) is 15 percent higher than macro F-score of the CRF model.

For each model, we have also presented a new form of augmentation of labeled data for the NER task, that is adding names after or instead of a descriptor of a certain type. A considerable improvement is recorded for relatively weak CRF and multilingual BERT models. For the fine-tuned models, the quality has barely grown. Nevertheless, if in some cases it is impossible to fine-tune BERT on a specialized collection, the presented augmentation for named entities could be of great use while extracting named entities of non-standard types. Besides, the proposed augmentation approach can be used in automated creation of a domain-specific NER annotated dataset from general datasets such as CONLL-2003, or Collection3. The described Sec_col collection and the trained RuCyBERT model can be obtained from the repository[5].

## References

1. *Bahdanau, D., Cho, K., Bengio, Y.* (2014), Neural machine translation by jointly learning to align and translate, available at https://arxiv.org/abs/1409.0473.
2. *Bocharov, V., Starostin, A., Alexeeva, S., Bodrova, A., Chunchunkov, A., Dzhumaev, S., Efimenko, I., Granovsky, D., Khoroshevsky, V., Krylova, I., Nikolaeva, M., Smurov, I., Toldova, S.* (2016), FactRuEval 2016: Evaluation of Named Entity Recognition and Fact Extraction Systems for Russian, Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialog 2016" [Komp'yuternaya Lingvistika i Intellektual'nye Tekhnologii: Trudy Mezhdunarodnoy Konferentsii "Dialog 2016"], Moscow, pp. 702–720.
3. *Bridges, R., Jones, C., Iannacone, M., Testa, K., Goodall, J.* (2013), Automatic labeling for entity extraction in cyber security, available at https://arxiv.org/abs/1308.4941.
4. *DeepPavlov* documentation, http://docs.deeppavlov.ai/en/master/. Last accessed 25 Dec 2019
5. *Devlin, J., Chang, M. W., Lee, K., & Toutanova, K.* (2018), Bert: Pre-training of deep bidirectional transformers for language understanding, available at https://arxiv.org/abs/1810.04805.
6. *Gasmi, H., Bouras, A., Laval, J.* (2018), LSTM Recurrent Neural Networks for Cybersecurity Named Entity Recognition, ICSEA-2018, Vol 11.
7. *Gareev, R., Tkachenko, M., Solovyev, V., Simanovsky, A., Ivanov, V.* (2013), Introducing baselines for Russian named entity recognition, International Conference on Intelligent Text Processing and Computational Linguistics, Springer, Berlin, Heidelberg, pp. 329–342.
8. *Howard, J., Ruder, S.* (2018), Universal language model fine-tuning for text classification, available at https://arxiv.org/abs/1801.06146.
9. *Joshi, A., Lal, R., Finin, T., Joshi, A.* (2013), Extracting cybersecurity related linked data from text, 2013 IEEE Seventh International Conference on Semantic Computing, Irvine, CA, USA , pp. 252–259.

[5] https://github.com/LAIR-RCC/InfSecurityRussianNLP

10. *Kobayashi, S.* (2018), Contextual augmentation: Data augmentation by words with paradigmatic relations, 2018 Conference of the North American Chapter of the Assoc. for Computational Linguistics, NAACL-2018, New Orleans, pp. 452–457.
11. *Kuratov, Y., Arkhipov, M.* (2019), Adaptation of Deep Bidirectional Multilingual Transformers for Russian Language, available at https://arxiv.org/abs/1905.07213.
12. *Lafferty J., McCallum A., Pereira F.* (2001), Conditional random fields: Probabilistic: models for segmenting and labeling sequence data, International Conference on Machine Learning ICML-2001.
13. *Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.* (2016), Neural architectures for named entity recognition, available at https://arxiv.org/abs/1603.01360.
14. *Mikolov, T., Chen, K., Corrado, G., & Dean, J.* (2013), Efficient estimation of word representations in vector space, available at https://arxiv.org/abs/1301.3781.
15. *Fellbaum, Ch.* (1998), WordNet: An Electronic Lexical Database, MIT-press
16. *Mozharova, V. , Loukachevitch, N.* (2016), Combining knowledge and CRF-based approach to named entity recognition in Russian, International Conference on Analysis of Images, Social Networks and Texts, Springer, Cham, pp. 185–195.
17. *Piskorski, J., Laskova, L., Marcinczuk M., Pivovarova, L., Priban P., Steinberger, J., Yangarberger, R.* (2019), The Second Cross-Lingual Challenge on Recognition, Normalization, Classification, and Linking of Named Entities across Slavic Languages, 7th Workshop on Balto-Slavic Natural Language Processing BSNLP-2019, Florence, pp. 63–74.
18. *Sang, E., Meulder, F.* (2003), Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition, Proc. of the 7th conference on Natural language learning at HLT-NAACL 2003, Vol 4., pp. 142–147.
19. *Sirotina, A., Loukachevitch, N.* (2019), Named Entity Recognition in Information Security Domain for Russian, Proceedings of RANLP-2019, Varna, pp. 1115–1122.
20. *Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. & Polosukhin, I.* (2017), Attention is all you need, Advances in neural information processing systems, pp. 5998–6008.
21. *Vlasova, N. A., Suleymanova, E. A., and Trofimov, I. V.* (2014), The message about Russian collection for named entity recognition task [Soobshchenie o russkoyazychnoj kollekcii dlya zadachi izvlecheniya lichnyh imen iz tekstov], Proceedings of computational and cognitive linguistics TEL, Kazan, pp. 36–40.
22. *Wei, J. W., Zou, K.* (2019), Eda: Easy data augmentation techniques for boosting performance on text classification tasks, Conference on Empirical Methods in Natural Language Processing EMNLP-2019, Hong Kong, pp. 6381–6387.
23. *Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., … & Klingner, J.* (2016), Google's neural machine translation system: Bridging the gap between human and machine translation, available at https://arxiv.org/abs/1609.08144.
24. *Yang Wang, W., Yang, D.* (2015), That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using #petpeeve tweets, 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, pp. 2557–2563.