Bull. Nov. Comp. Center, Comp. Science, 45 (2021), 1-14

Information extraction from news texts using a joint deep learning model

Leonid Anisiutin, Tatiana Batura, Nikita Shvarts

Abstract. This paper describes the experiments for the task on information extraction from news texts in Russian in a setting with a wide variety of types of entities and relations. We have adapted the SpERT model which uses the BERT network as a core for the joint extraction of entities and relations. The results obtained for the named entity recognition are quite good and comparable with the results on English datasets. However, the hypothesis that the joint extraction of entities and relations improves the quality of extraction has not been confirmed. The results showed that using a full context instead of a local context improves the quality of extraction of both entities and relations.

Keywords: natural language processing, information extraction, language models, SpERT model, entity recognition, relation classification, machine learning

Introduction

In the modern world, the amount of text information is rapidly increasing. To work with large amounts of data effectively, it is necessary to improve and create new methods for extracting information from texts. This paper studies the methods for the automatic extraction of entities and classification of relations from texts in Russian. As a rule, modern approaches based on deep learning methods solve these two problems separately.

Named Entity Recognition (NER) is a subtask of information extraction aiming to find named entities mentioned in a natural language text and classify them into a predefined set of categories, such as the names of people, organizations, locations, expressions of time, date, etc. Named entities are understood as a word or several words denoting an object. One entity often has a whole variety of representations in the form of words, which gives the variability of solutions to the problem. All sorts of terms can be referred to unambiguous or "hard" correspondences, for example, the names of biological species and substances.

Usually, the NER task is split into two separate tasks: determining the boundaries of an entity and entity classification by the type. As a rule, the first stage is simplified to a segmentation problem: names are defined as the continuous intervals of words without nesting, which is a fairly strong simplification. However, there are more advanced methods for finding nested and overlapping entities, which is a more challenging task.

A formal statement of the entity recognition task is as follows. Let us denote by e_i the tokens of an input text. Let a sequence of tokens be $s = (e_i, e_{i+1}, \ldots, e_{i+k})$. There is a predefined set of labels

$$E \cup \{none\},$$

for example, $\{person, profession, age, country, \ldots\} \cup \{none\}$, where $\{none\}$ is a special label indicating that the sequence does not belong to any class. The aim is to build a classifier that matches a label with an arbitrary sequence of tokens:

$$\hat{y}_s : s \to E \cup \{none\}.$$

Relation extraction (RE) is the task of finding and classifying semantic relations between entities from a predetermined set of types. Formally, the problem is formulated as follows. Let the set of labels for relation types be

$$R \cup \{no\_relation\},$$

for example, $\{work\_as, point\_in\_time, takes\_place, \ldots\} \cup \{no\_relation\}$, where $\{no\_relation\}$ is a label indicating that there is no relation between two entities. The task is to build a classifier that matches a label to a pair of entities (s_i, s_j):

$$\hat{y}_r : (s_i, s_j) \to R \cup \{no\_relation\}.$$

There are various approaches to extracting relations; in essence, they are divided into two approaches. Firstly, this is an ontology-based approach, when researchers try to formalize the relations between entities through some conceptual schemes and certain rules. An example of such an approach is Gene Ontology, a bioinformatics project dedicated to the creation of a unified terminology for annotating genes and gene products of all biological species. Secondly, these are the increasingly popular end-to-end methods for identifying relations and their semantics based on neural network techniques. This approach shows good results in terms of speed and flexibility [1].

The earlier methods solve the two problems separately [2-4]. Recently, however, the approach using **joint models** (NER and RE) has become increasingly popular [5, 6]. This approach enhances the mutual influence of the expressions for relations and entities in a text, since knowledge about relations can be useful in entity recognition and vice versa. The results of experiments conducted on English datasets with a small number of the entity and relation types are promising. Still not enough attention is paid to the research focused on working with the Russian language and in a setting with a wide variety of the types of entities and relations. For example, the papers [7, 8] describe the experiments with RE methods on Russian datasets with 7 and 11 relation types. This number of relations is not complete and not enough to reflect the semantics of different statements accurately.

The paper is structured as follows. Section 1 provides an overview of the existing methods for solving the assigned tasks and analyzes their advantages and disadvantages. Section 2 contains a detailed description of the SpERT model, which is based on the Transformer architecture and has been adapted for Russian language processing. The experiments were carried out on a Russian dataset of news texts marked with 29 types of entities and 49 types of relations. The collection size is about 900 documents. Section 3 describes this

dataset and model settings. Section 4 compares the results of experiments with pre-trained models (Russian and multilingual), taking into account the full and local contexts. The developed software components can be used in further research.

1. Related work

Traditionally, models for the extraction of entities and relations are used separately for the detection of entities and for the classification of relations. For both tasks, neural network approaches have firmly taken the leading positions. Various approaches to the classification of relations have been investigated, in particular, on the basis of recursive neural networks (RNNs) and convolutional neural networks (CNNs). In addition, models based on the Transformer architecture were used to classify relations [9, 10]: the input text is passed through the model once, and the resulting embeddings are classified, but these methods work with the pre-labeled named entities. In contrast, the method used in our work does not rely on marked-up entities and jointly discovers entities and relations in one pass.

In [11], the joint extraction of entities and relations is considered as a problem of filling a table, where each cell corresponds to a pair of words in a sentence. The diagonal of the table is filled with the BILOU-tags of the tokens themselves. BILOU (Begin, Intermediate, Last, Other, Unigram) is a text tagging format enabling entity extraction. Other cells are filled with the relations between the corresponding pairs of tokens. Relations are predicted by matching the last words of entities. The table is populated with relation types by minimizing the scoring function based on several features such as parts of speech and entity classes. To find the optimal solution for filling tables, beam-search is used with the support of several trajectories.

In another work, [12], the joint extraction of entities and relations is also considered as the task of filling a table. However, unlike [11], the authors use a bidirectional RNN to label each pair of words.

A slightly different approach is considered in [13]. The authors use a composite model for the joint extraction of entities and relations. First, the bidirectional LSTM network labels the entities according to the BILOU schema. Second, a tree-structured bidirectional RNN operates on a parsing tree between a pair of entities to predict the type of relation. Something similar was discussed in [14], which uses a BILOU-based combination of a bidirectional LSTM and CNN to extract a high-level representation of the input sentence feature. Since NER is performed only for the most likely relations, this approach predicts fewer labels as compared to the table-filling approaches. In [4], the input tokens are first encoded using a bidirectional LSTM network. Then another LSTM network operates on each encoded word representation and outputs the entity boundaries (similar to the BILOU scheme) along with their relation type. The simultaneous association of one entity with several others is not considered.

In [5], the authors also use a bidirectional LSTM network to encode each word of a sentence. They use character embeddings along with Word2Vec-derived embeddings as input representations. Entity boundaries and tags are retrieved using a conditional random field (CRF). Unlike [4], the approach considered in [5] also allows the correct handling of the cases when one entity is associated with several relations at the same time. One of the

latest works is [15], in which the model was built on the basis of BERT and trained under the problem of answering questions reduced to the extraction of relations for certain questions. The main disadvantage of this approach can be attributed to the need to formulate these questions.

In [16], the authors present the most promising approach from our point of view, which is called SpERT (Span-based Joint Entity and Relation Extraction with Transformer Pre-training). The model used in this approach combines the advantages of the BERT model, such as the ability to build context-sensitive embeddings, and span-based approach for entities extraction and relation classification using light-weight classifiers. This allows us to overcome several limitations revealed in the works presented above. Therefore, the SpERT model has been chosen as the basis for our study and is described in detail below.

2. Model description

The transformer-based networks, such as BERT [17], GPT [18], TransformerXL [19] or RoBERTa [15], have recently attracted a lot of attention in the natural language processing research community. These models use a multi-head self-attention mechanism as a key mechanism for tracking mutual influence between tokens [20]. In this way, context-sensitive embeddings can be obtained which disambiguate homonyms and better express semantic and syntactic information. Note that word embeddings are context-sensitive if the embeddings in different sentences are different for the same word.

Transformer networks are usually pre-trained on large datasets on two tasks simultaneously: firstly, it is a language modeling task, and secondly, binary classification (whether two sentences are next to each other). The resulting models can then be transferred to target tasks with a relatively small amount of training data, resulting in a high level of performance in many natural language processing tasks, such as answering questions [21] or contextual emotion detection [22].

The SpERT architecture [16] uses the BERT network as a core for the joint extraction of entities and relations. The main idea of the approach is to allocate spans: any subsequence (or interval) of tokens is a candidate for an entity and any pair of spans can be involved in a relation. This model performs a full search on all of these hypotheses. An advantage of the approach based on the allocation of spans, in contrast, for example, to the BIO-scheme is that in the BIO-scheme each word has exactly one label, and it cannot be part of multiple overlapping or nested entities, which is, generally speaking, a significant oversimplification. Unlike the previous work based on the BIO / BILOU scheme [5], the span-based approach can identify nested and overlapping entities. Since the transformer models such as BERT are computationally expensive, an important advantage of SpERT is that it performs only one forward pass for each input sentence, and thereby decision making is a computationally easy procedure based on the resulting word embeddings. Unlike other recent approaches, for example [23], SpERT has simple and shallow classifiers for entities and relations. This facilitates effective learning and allows full search across all spans. However, one of the limitations of the approach is the inability to

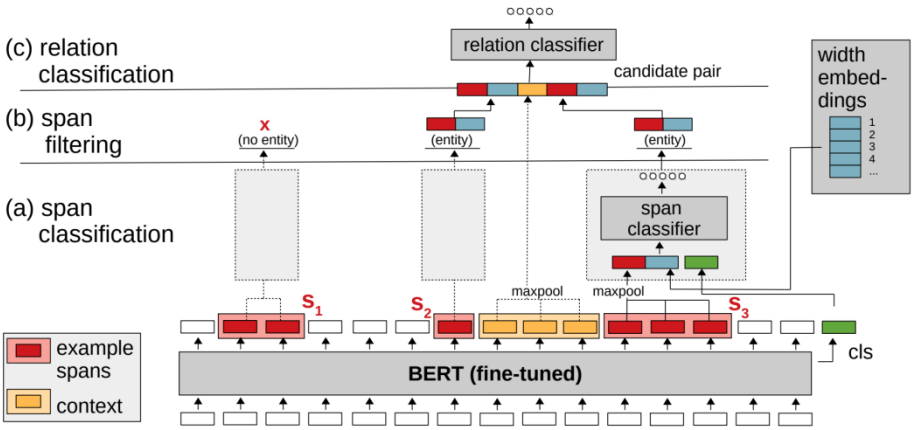


Figure 1. The original SpERT model architecture presented in [16]

process the relations in different sentences, since the model works with one sentence at a time.

Let us describe the SpERT model in more detail. We decided to reuse entirely the model presented by the authors in [16] without changes, since the original approach shows promising results. Nonetheless, we would like to highlight the main principal components of this approach. The schema of the approach towards joint entity and relation extraction using SpERT is shown in Figure 1.

The model input is a list of tokens obtained from words as a result of the Byte-pair encoding (BPE) technique, which breaks words into fragments, which are more frequently used. This procedure is an important step, especially for morphologically rich languages such as Russian, because it leads to a significant reduction in the model vocabulary and avoids problems with the words not represented in the training set.

After processing the input sequence of tokens by the BERT model, its output is a sequence of embeddings ($e_1, e_2, \ldots, e_n, c$), where ($e_1, e_2, \ldots, e_n$) are token embeddings, and c is an embedding of the overall sentence context. Thus, for the sentence of length n the output length is $n + 1$. All possible continuous subsequences of a length less than 10 are selected from the sequence of token embeddings as candidates for entities. Next, an aggregate transform is applied to each candidate sequence (in our case max-pooling). However, the aggregation loses spatial information about the length of the candidate and, in order to save this information, a special embedding is concatenated, which characterizes the length of the sequence. Also, in order to classify entities correctly, the classifier needs a context, so the context vector c is concatenated to the resulting vector.

The embedding obtained at the previous step goes to the input linear classifier of relations, the output of which is the probability distribution of the classes of entities, including the technical class, which means that the candidate is not an entity. The class with the highest probability is assigned to the candidate. Candidates falling into the "non-entity" class are filtered and do not participate in the further process.

Further, all relations are considered in pairs. For each pair, its embedding is concatenated with the embedding of the local context. Under the local context, we mean all words lying between entities. A local context embedding is obtained by using the same aggregation function as in the entity extraction stage (max-pooling, in our case). The resulting vector goes as an input to the linear classifiers. In this approach, we have as many linear classifiers as there are relation classes. The sigmoidal function is used as the activation function of the last layer, and if the value is greater than a certain threshold of confidence α, we assume that there is a relation between the given pair of entities.

The weights of the classifiers act as training parameters, and additional training of the basic BERT model is performed. The loss function is the sum of two terms: $L = L_{entities} + L_{relations}$, where the first is responsible for the classification errors of entities and the second, for the errors in the classification of relations.

When training classifiers, we take positive examples from the labeled data and select negative examples randomly from the remaining unlabeled data. That is, for the entity classifier, a fixed number of random entity candidates are considered as negative examples; for the relation classifier, negative examples are selected from entities not related by any relation.

3. Experiments

3.1. Data description

The experiments were carried out on the recently released dataset NEREL [24]. It consists of news texts in Russian labeled with 29 types of entities and 49 types of relations. NEREL contains more than 13,000 sentences with 56K annotated named entities and 39K annotated relations. Tables 1 and 2 show the top 10 most frequent types of entities and relations, respectively.

Table 1. Statistics for the top 10 most frequent types of entities in the corpus

Сущность	Количество
PERSON	9707
PROFESSION	9093
ORGANIZATION	7102
EVENT	108
DATE	4877
COUNTRY	4475
CITY	2205
NUMBER	1978
AGE	1207
ORDINAL	996

Table 2. Statistics for the top 10 most frequent types of relations in the corpus

Отношение	Количество
WORKPLACE	3352
WORKS AS	3072
PARTICIPANT IN	2896
ALTERNATIVE NAME	2698
HEADQUARTERED IN	1115
POINT IN TIME	1098
TAKES PLACE IN	1084
ORIGINS FROM	1041
LOCATED IN	797
AGE IS	678

An example of labeled data from a corpus is shown in Figure 2.

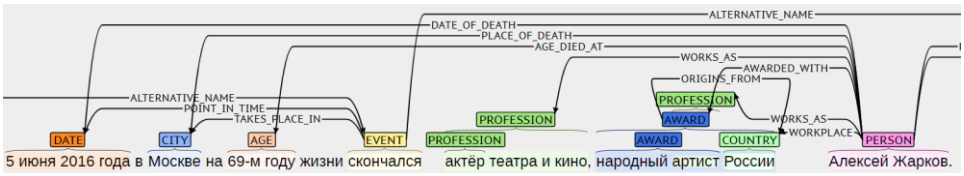


Figure 2. An example of a sentence with markup of entities and relations

However, quite a large number of relations are relations between entities from different sentences that cannot be taken into account within the framework of the chosen model. We have excluded them from consideration, and therefore only 23 types of relations were taken for experiments (the number of relations is 8731; the number of named entities is 53627).

3.2. Model setting description

All calculations were performed on a computer with an NVIDIAGeForce RTX 2060 SUPER graphics card. The programming language was Python, the framework for neural networks training was PyTorch.

Experiments were carried out on two pre-trained models: the Russian-language DeepPavlov “rubert-base-cased” model and multilingual “bert-base-multilingual-cased” model. The parameters of experiments for the Russian-language RuBERT model were as follows: the batch size was 5; the number of negative examples in a negative sampling for entities was 100 and for relations, 150; the learning step was 5; and the learning step reduction factor was 0.01. For the multilingual model, only the size of the batch differed (was equal to 2) because it was larger and the previous size did not fit into memory. The optimizer algorithm was AdamW (Adam with Decoupled Weight Decay Regularization).

The first model is one of the best models of feature extractors for Russian, and we added the second model because the corpus has foreign words, which might affect the quality.

4. Results

4.1. Model quality assessment metrics

We used standard metrics (precision, recall and F1-measure) to evaluate the quality of the results obtained.

$$\begin{aligned} precision &= \frac{tp}{tp + fp}, \\ recall &= \frac{tp}{tp + fn}, \\ F1 &= 2 \cdot \frac{precision \cdot recall}{precision + recall}, \end{aligned} \tag{1}$$

where *tp* is the number of correctly defined words of the positive class (true positive);
fp is the number of words incorrectly assigned to the positive class (false positive);
fn is the number of words incorrectly assigned to the negative class (false negative).
In this case, the F1 measure for a multiclass classification is generalized as follows:

$$F1_{micro} = 2 \cdot \frac{precision_{micro} \cdot recall_{micro}}{precision_{micro} + recall_{micro}},$$

$$precision_{micro} = \frac{\sum_i tp}{\sum_i tp + \sum_i fp}, \quad recall_{micro} = \frac{\sum_i tp}{\sum_i tp + \sum_i fn}, \tag{2}$$

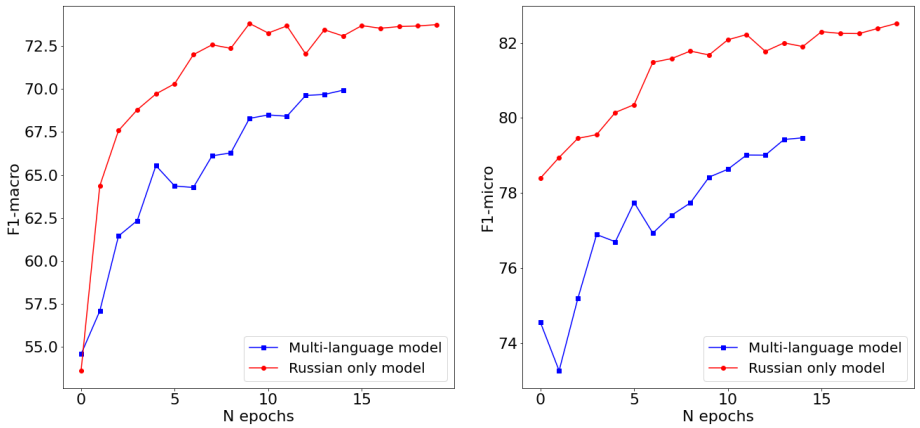
$$F1_{macro} = \frac{1}{|N|} \sum_{i \in N} F1_i,$$

where $F1_i$ is F1-measure for the *i*-th class calculated in a one-vs-rest approach to enable the calculation of the measure as for a binary classification, as given in Formula 1.

These metrics are good for measuring the quality in simple cases when, for example, an entity consists of one word. However, the model can give a partially correct answer, for example, add an extra word, miss something, or correctly define a word sequence for an entity, but make a mistake when defining a type. Nevertheless, though these metrics do not take into account such cases, the academic community has not yet proposed a better well-established alternative, so our experiments are also evaluated in terms of these metrics.

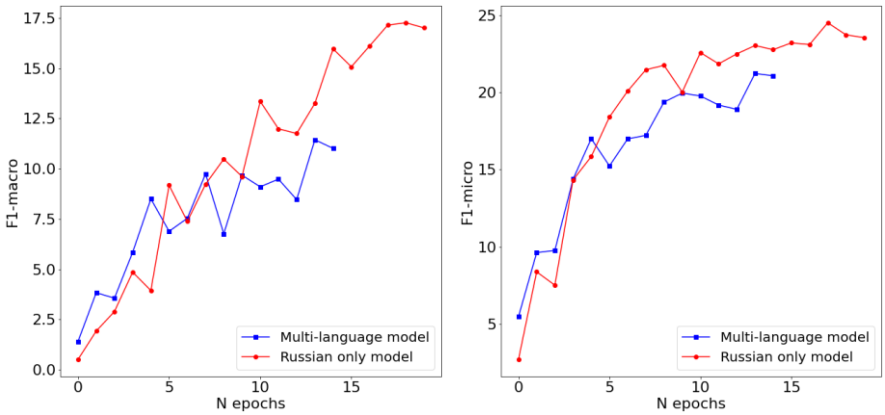
4.2. Comparison of results

As a result of the experiments, we have observed that the Russian model gives better results than the multilingual one. Despite the presence of foreign vocabulary in sentences, it does not make a significant contribution to the extraction of relations and entities (see Figure 3 and Figure 4).



(a) $F1^{NER}_{macro}$ (b) $F1^{NER}_{micro}$

Figure 3. Comparison of metrics for the NER task with multilingual and Russian models. We use F1-macro and F1-micro measures described in 4.1



(a) $F1^{RE}_{macro}$ (b) $F1^{RE}_{micro}$

Figure 4. Comparison of metrics for the RE task with multilingual and Russian models. We use F1-macro and F1-micro measures described in 4.1

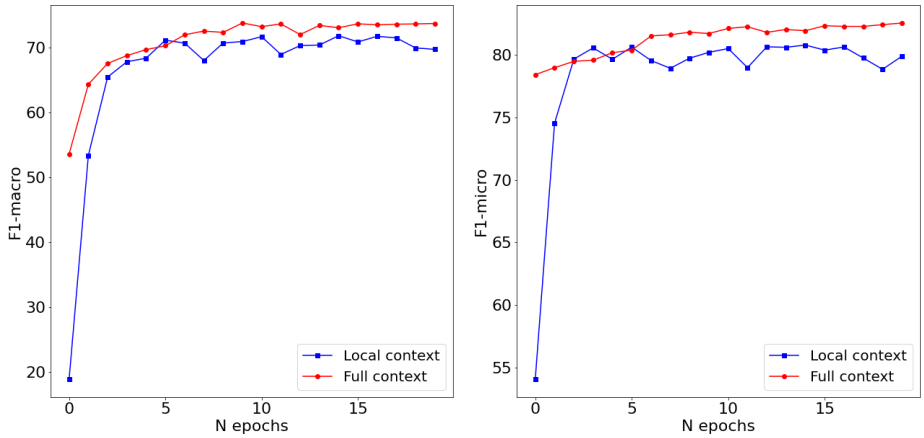
Table 3 shows metrics for the top-10 entity types.

Table 3. Top-10 results for NER on the Russian dataset

Entity type	precision	recall	F1-score
PERSON	81.70	94.86	93.25
AGE	92.66	90.23	91.43
CITY	87.10	87.30	87.20
PERCENT	82.35	87.50	84.85
ORDINAL	74.21	89.62	81.19
ORGANIZATION	78.37	80.22	79.29
RELIGION	76.19	69.57	72.73
LAW	73.33	52.03	60.87
DISEASE	68.37	47.18	55.83
FAMILY	44.44	23.53	30.77

As described in Section 2, two named entities along with a local context (i.e. words between the entities) are fed to the relation classifier as input. This approach is more suitable for the English language because it has a strict word order. The Russian language, on the contrary, has a relatively free word order, so we conducted experiments not only with a local context, but also with a full context, when the whole sentence containing both candidate entities is fed to the classifier.

All further results are given for the RuBERT model, since this model showed better results than the multilingual one. Figures 5 and 6 show the dependence of the $F1_{micro}$ and $F1_{macro}$ measures on the number of learning epochs on the test part of the dataset with a local context and a full context.



(a) $F1^{NER}_{macro}$ (b) $F1^{NER}_{micro}$

Figure 5. The impact of a local and full context on the results for the NER task. We use F1-macro and F1-micro measures described in 4.1

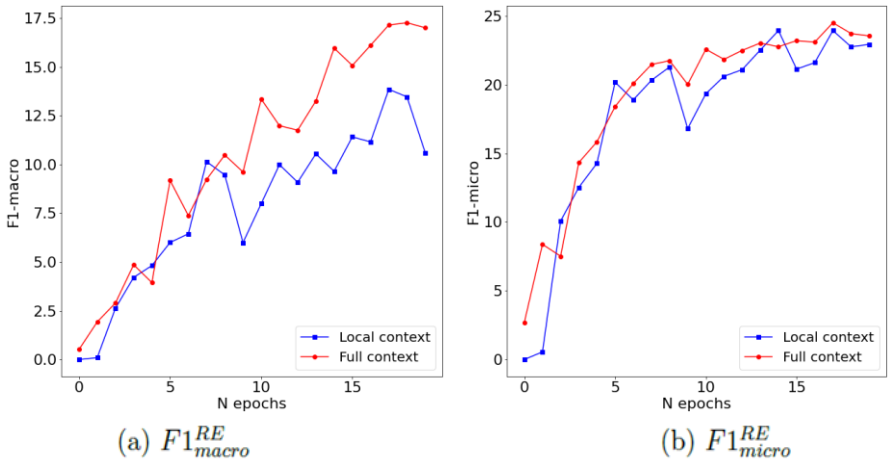


Figure 6. The impact of a local and full context on the results for the RE task. We use F1-macro and F1-micro measures described in 4.1

Clearly, using the full context improves the quality of extraction of both entities and relations.

Table 4 shows the metrics for the top-10 relation types.

Table 4. Top-10 results for RE on the Russian dataset

Отношение	precision	recall	f1-score
WORKPLACE	26.76	32.85	29.50
LOCATED IN	16.67	38.38	23.24
MEDICAL CONDITION	25.00	18.18	21.05
PARTICIPANT IN	14.38	19.55	16.57
AGE IS	28.57	8.00	12.50
SUBORDINATE OF	33.33	6.67	11.11
ABBREVIATION	16.67	8.33	11.11
IDEOLOGY OF	33.33	5.88	10.00
AWARDED WITH	5.56	10.00	7.14
ALTERNATIVE NAME	4.88	3.92	4.35

Table 5 shows the scores for both tasks on the Russian dataset.

Table 5. Final quality indicators

Модель	$F1^{NER}_{micro}$	$F1^{NER}_{macro}$	$F1^{RE}_{micro}$	$F1^{RE}_{macro}$
Multilingual BERT (local context)	-	-	-	-
Multilingual BERT (full context)	79.47	69.93	21.06	11.01
RuBERT (local context)	79.86	69.75	22.92	10.60
RuBERT (full context)	82.52	73.74	23.54	17.00

It should be noted that for the NER task, the obtained results are comparable to the results on similar English corpora (CoNLL04, SciERC, ADE), which is shown in Table 6 taken from the paper [16]. However, the quality of relation extraction is still relatively low, although it increases with the use of the full context. Unfortunately, it is not possible to make a comparison with the indicators of similar datasets in Russian since the existing datasets have a smaller amount of data and number of types for entities and relations.

Table 6. Results for NER and RE with SpERT on English datasets

Dataset	$F1^{NER}_{micro}$	$F1^{NER}_{macro}$	$F1^{RE}_{micro}$	$F1^{RE}_{macro}$
CoNLL04	88.94	86.25	71.47	72.87
SciERC	67.62	70.33	46.44	50.84
ADE	89.25	89.28	79.24	78.84

These results for relation extraction are likely to be due to several factors. Firstly, the number of the examples for a large number of classes in the dataset is insufficient (in the future, this will be remedied). Secondly, the structure of sentences in Russian is more complex, and the Russian language is richer morphologically, so, on a larger amount of balanced data, the quality should be better.

5. Conclusion

In this paper, we have described the experiments for the task of information extraction from news texts in Russian in a setting with a wide variety of the types of entities and relations. We have adapted the SpERT model, which uses the BERT network as a core for a joint extraction of entities and relations.

The results obtained for NER are quite good and comparable with the results on English data, but the hypothesis that a joint extraction of entities and relations improves the quality of extraction has not been confirmed. The results have shown that using a full context (instead of a local context) improves the quality of the extraction of both entities and relations.

References

[1] Nguyen D.Q., Verspoor K. End-to-end neural relation extraction using deep biaffine attention // Proc. European Conference on Information Retrieval. – Springer, 2019. – P. 729 – 738.

[2] Yadav V., Bethard S. A survey on recent advances in named entity recognition from deep learning models // Proc. 27th International Conference on Computational Linguistics (COLING 2018). – 2018. – P. 2145 – 2158.

[3] Zeng D., Liu K., Lai S., Zhou G., Zhao J. Relation classification via convolutional deep neural network // Proc. The 25th international conference on computational linguistics: technical papers (COLING 2014). – 2014. – P. 2335 – 2344.

[4] Zhang D., Wang D. Relation classification via recurrent neural network. – https://arxiv.org/pdf/1508.01006.pdf.

[5] Bekoulis G., Deleu J., Demeester T., Develder C. Joint entity recognition and relation extraction as a multi-head selection problem // Expert Systems with Applications. – 2018. – Vol. 114. – P. 34 – 45.

[6] Luan Y., Wadden D., He L., Shah A., Ostendorf M., Hajishirzi H. A general framework for information extraction using dynamic span graphs // Proc. The 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL). – 2019. – Vol.1. – P. 3036 – 3046.

[7] Bruches E., Pauls A., Batura T., Isachenko V. Entity recognition and relation extraction from scientific and technical texts in Russian // 2020 Science and Artificial Intelligence conference (S.A.I.ence). – 2020. – P. 41 – 45.

[8] Artemova E., Batura T., Golenkovskaya A., Ivanin V., Ivanov V., Sarkisyan V., Smurov I., Tutubalina E. So What's the plan? Mining strategic planning documents // International Conference on Digital Transformation and Global Society (DTGS 2020). – Springer, Cham. 2020. – P. 208 – 222.

[9] Verga P., Strubell E., McCallum A. Simultaneously self-attending to all mentions for full-abstract biological relation extraction // Proc. The 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL). – 2018. – Vol.1. – P. 872 – 884.

[10] Wang H., Tan M., Yu M., Chang S., Wang D., Xu K., Guo X., Potdar S. Extracting multiple-relations in one-pass with pre-trained transformers // Proc. The 57th Annual Meeting of the Association for Computational Linguistics (ACL). – 2019. – P. 1371 – 1377.

[11] Miwa M., Sasaki Y. Modeling joint entity and relation extraction with table representation // Proc. The 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). – 2014. – P. 1858 – 1869.

[12] Gupta P., Schütze H., Andrassy B. Table filling multi-task recurrent neural network for joint entity and relation extraction // Proc. The 26th International Conference on Computational Linguistics: Technical Papers (COLING 2016). – 2016. – P. 2537–2547.

[13] Miwa M., Bansal M. End-to-end relation extraction using LSTMs on sequences and tree structures // Proc. The 54th Annual Meeting of the Association for Computational Linguistics (ACL). – 2016. – P. 1105 – 1116.

[14] Zhou P., Zheng S., Xu J., Qi Z., Bao H., Xu B. Joint extraction of multiple relations and entities by using a hybrid neural network // Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated BigData. – Springer, 2017. – P. 135 – 146.

[15] Liu Y., Ott M., Goyal N., Du J., Joshi M., Chen D., Levy O., Lewis M., Zettlemoyer L., Stoyanov V. RoBERTa: A robustly optimized BERT pretraining approach. – https://arxiv.org/pdf/1907.11692.pdf.

[16] Eberts M., Ulges A. Span-based joint entity and relation extraction with transformer pre-training. – http://ecai2020.eu/papers/1283_paper.pdf.

[17] Devlin J., Chang M., Lee K., Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding // Proc. NAACL-HLT. – 2019. – P. 4171 – 4186.

[18] Radford A., Narasimhan K., Salimans T., Sutskever I. Improving language understanding by generative pre-training. – 2018. – https://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf.

[19] Dai Z., Yang Z., Yang Y., Carbonell J., Le Q., Salakhutdinov R. Transformer-xl: Attentive language models beyond afixed-length context // Proc. The 57th Annual Meeting of the Association for Computational Linguistics. – 2019. – P. 2978 – 2988.

[20] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser L., Polosukhin I. Attention is all you need // Proc. The 31st International Conference on Neural Information Processing Systems (NIPS'17). – 2017. – P. 6000 – 6010.

[21] Yang W., Xie Y., Lin A., Li X., Tan L., Xiong K., Li M., Lin J. End-to-end open-domain question answering with bertserini // Proc. NAACL-HLT. – 2019. – P. 72 – 77.

[22] Chatterjee A., Narahari K., Joshi M., Agrawal P. SemEval-2019 task 3: EmoContext contextual emotion detection in text // Proc. The 13th International Workshop on Semantic Evaluation. – 2019. – P. 39 – 48.

[23] Wadden D., Wennberg U., Luan Y., Hajishirzi H. Entity, relation, and event extraction with contextualized span representations // Proc. The 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-JCNLP). – 2019. – P. 5784 – 5789.

[24] Loukachevitch N., Artemova E., Batura T., Braslavski P., Denisov I., Ivanov V., Manandhar S., Pugachev A., Tutubalina E. NEREL: A Russian Dataset with Nested Named Entities, Relations and Events // Proc. The International Conference on Recent Advances in Natural Language Processing (RANLP 2021). – 2021. – P. 876 – 885.