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

Moscow, June 17–20, 2020

R-BERT FOR RELATIONSHIP EXTRACTION ON RUSSIAN BUSINESS DOCUMENTS

Korzun V. A. (sdernal2@gmail.com)
MIPT ABBYY Lab, Moscow, Russia

This paper provides results of participation in the Russian Relation Extraction for Business shared task (RuREBus) within DialogueEvaluation 2020. Our team took the first place among 5 other teams in Relation Extraction with Named Entities task. The experiments showed that the best model is based on R-BERT model. R-BERT achieved significant result in comparison with models based on Convolutional or Recurrent Neural Networks on the SemEval-2010 task 8 relational dataset. In order to adapt this model to RuREBus task we also added some modifications like negative sampling. In addition, we have tested other models for Relation Extraction and Named Entity Recognition tasks.

Key words: BERT, relationship extraction, named entity recognition, russian business, deep learning

DOI: 10.28995/2075-7182-2020-19-467-473

R-BERT ДЛЯ ИЗВЛЕЧЕНИЯ ОТНОШЕНИЙ НА ДОКУМЕНТАХ РУССКОГО БИЗНЕСА

Корзун В. А. (sdernal2@gmail.com)
МФТИ ABBYY Lab, Москва, Россия

1. Introduction

At the moment, many natural language processing (NLP) tasks are solved with large pre-trained language models like BERT [1]. Relationship extraction (RE) is one of them. One of state-of-the-art results [2] is based on pre-trained BERT with some modifications. There are also models based on Convolution or Recurrent Neural Networks, like [3] and [4].

Relationship extraction task can be considered as a sentence classification task. Given a sentence and a pair of nominals, the objective is to identify the relation between nominals. In this case task could be called relationship classification.

Named entity recognition (NER) could be a sub-task for relationship extraction, when pairs of nominals are not given and first must be found. NER task can be considered as a sequence labeling task. Given a sequence of tokens in a sentence, the objective is to classify each token. Classes should correspond to entity types. Quite often for each type of entity classes are added, corresponding to the beginning, middle or end of the entity.

In a broader sense in relationship extraction task there are also possible cases, when sentence contains more than two entities. One way to solve this problem is to duplicate sentence for each possible pair of entities. Then, the resulting samples are classified separately. Thus the relationship extraction task could be reduced to the relationship classification task.

Our contribution is to adapt state-of-the-art method for relationship classification to case when more than two entities could be found in the sentence by using special sampling approach.

2. Shared task overview

RuREBus is a public task for named entity recognition and relationship extraction for Russian on business documents. The dataset provided is new and was developed for this particular task. Authors describe their corpus as a real set of documents from business and task could be more complicated than for known datasets for relationship classification. For more information refer to [5].

2.1. Problem description

The whole competition divided into 3 tasks:

- **Named Entity Recognition.** The objective is to find entities spans and classify them. The object metric is exact micro F-measure.
- **Relationship Extraction with Named Entities.** Given an entities for each document find a relationship between them and provide relation type. The objective metric is micro F-measure.
- **End-to-end Relationship Extraction.** Similar to the previous, but without entities markup. First, the objective is to find entities, than identify relations between them.

The competition took place in 2 phases. First for 1st and 3rd task and second for the 2nd task.

2.2. Data description

Shared task organizers provided manually annotated documents and the large corpus of unlabelled documents for training models. The train set consists of 188 documents. Each document includes a raw text file and an annotation file in BRAT format. Annotation file contains a list of entities and relations in the following format:

‘`T{idx}\t{type} {span start} {span end}\t{value}`‘ for entities
‘`R{idx}\t{type} Arg1:T{e1 idx} Arg2:T{e2 idx}`‘ for relations

Spans are given as offset in symbols from the beginning of a document.

The test set provided for the 1st phase contains 544 raw text file for NER and end-to-end RE tasks. For the 2nd phase there are also annotations files with entities for RE with NE’s task.

2.3. Data problems

We found some difficulties working with dataset:

- **Inconsistent punctuation.** Some documents contain punctuation marks separated from other tokens by space. It could help in tokenization, but other documents have punctuation marks attached to tokens.
- **New line separation.** New line separators can be found between sentences as well as inside sentences. This brings a lot of problems with sentence segmentation.
- **Long sentences.** There are also a lot of long sentences in the dataset. Some of them represent enumerations divided by semicolon. It also brings problems with training models such BERT which GPU memory consumption depends on sequence length.

To train BERT on GPU with 8GB RAM, sequence length should be limited. Therefore, we decided to split sentences as follows:

1. split input document by sentences using nltk [6] sentence tokenizer
2. split the remaining long sentences by the following substrings consistently:

   ‘`;`‘, ‘`\n\n`‘ and ‘`\n`‘

3. item split the remaining sentences by maximum length (120)

However, to train models based on recurrent networks we decided to use only nltk sentence tokenizer.

3. Solution for NER task

For the NER task we have tried different models based on LSTM with Self-attention and BERT.

3.1. Self-attention LSTM

For the NER task we mostly used Bidirectional LSTM followed by Self-attention. First, we have tried to use only Russian fasttext word embeddings [10] as token features and got an adequate score. Then, we have added casing features, part-of-speech tags from pymorphy [7] and character embeddings. Results and model dimensions are given below.

Table 1: Results

Features	F-score on test
only fasttext	0.4357
+ casing + char	0.4559
+ POS-tags	0.4638

Table 2: Model dimensions

LSTM hidden	400
word embeddings	300
char embeddings	20
chars hidden	50
POS-tag embeddings	30
casing embeddings	10
dropout	0.4

3.2. BERT for NER

Further, we have tried to use output from pre-trained BERT as token features instead of using fasttext embeddings and other features. We have tried the simple Multilayer perception and the same encoder from previous experiments (BiLSTM + Self-Attention) as encoders. We have also tried to freeze and unfreeze BERT parameters. And as pre-trained BERT models we used RuBERT from DeepPavlov [8] and multilingual-cased from Transformers library [9]. The results are listed in **Table 3**.

Table 3: BERT results on NER

Model	F-score
RuBERT(frozen) + MLP	0.1916
RuBERT(frozen) + SALSTM	0.2941
*RuBERT(unfrozen) + SALSTM	0.43
*Multilingual cased BERT(unfrozen) + MLP	0.5144
*RuBERT(unfrozen) + MLP	0.5469
lst-place competitor	0.561

* the results were obtained after phase end

We found that using BERT output as embeddings for subsequent encoder degrades the overall quality. To get the maximum benefit from BERT, we must unfreeze its parameters and use the simplest classifier. Furthermore, the latest release of RuBERT for PyTorch outperforms standard multilingual BERT. However, using just BERT is not enough to take state-of-the-art result for this task. To improve our results we could use ensemble of BERT models or other state-of-the-art approaches for NER.

4. Solution for RE task

For relationship extraction task we used two models based on BERT. For both models we used pre-trained multilingual-cased BERT. To solve this task, first, we reduced it to relationship classification task like SemEval-2010 Task 8 by sampling sentence for each pair of entities.

4.1. Negative sampling

In SemEval-2010 Task 8 only two entities are given for each sentence. In RuREBus task we have various number of entities for each sentence. Therefore, we create a sample containing a sentence and a pair of entities, for each two entities in the sentence. A sample is called negative if its pair of entities is not in the markup, the rest of samples are positives. Now the task is reduced to SemEval task, where samples are taken instead of sentences.

The number of samples created such way was enormous and the training epoch would last a very long time. Therefore we suggested to reduce the number of negative samples. The way of reduction is taking all positive pairs and some number of negatives. For negative samples we took random pairs from sentence in number similar to positive samples. There were also taken some negative samples from sentences without positive ones. And for each train epoch negative sample are taken randomly. This allowed us to reduce number of training batches by about 3 times. Nevertheless, all possible pairs of nominals are taken for evaluation.

4.2. NER embeddings over BERT

The first idea is to use BERT with NER embeddings. We concatenated BERT output with NER embeddings for each token. Then it is passed through attention to create sample feature vector. The vector obtained is used by MLP classifier to predict classes distribution.

4.3. R-BERT

The next model is inspired by [2]. Their model got one of the best results in SemEval-2010 Task 8 [11].

The approach is as follows. First, special tokens are inserted before and after entities and modified sentence is passed through the pre-trained BERT. Then, the BERT outputs corresponding to the [CLS] token and each entity are taken. Entities outputs are averaged and all 3 received vectors pushed through FeedForward layers. Then they are concatenated and passed into a Linear Layer followed by Softmax. All hyperparameters are taken from the original article except the max sentence length which was 120. The table with results of used models and the closed competitor are listed below:

Table 4: Results on RE

Model	F-score
BERT + NER embeddings	0.2066
R-BERT	0.44
Closest competitor	0.394

Hence, R-BERT outperformed our first model and the closest competitor and got state-of-the-art result on this task. It seems to us that the main feature of R-BERT is special token insertion. This could help BERT to attend on entities and generate output more suitable for the task. R-BERT authors also claim that output corresponding the [CLS] token represents the semantics of the sentence and the output corresponding the entities represents their semantics. As a result, this model achieved the best score in the competition. However, some samples could be lost due to tokenization. And the large unlabeled corpus was not used. It can be used to pre-train BERT for the domain and probably get better results.

5. Conclusion

In this paper we present a winning solution of Russian Relation Extraction for Business task within Dialogue Evaluation 2020. We took one of the best models for Relation Classification for English and combined it with special sampling procedure. We have also compered BERT and RNN based approaches for Named Entity Recognition task. This result with such score is not sufficient for fully automatic relationship extraction, given also the low score on named entity recognition task. Nevertheless, this approach can be used as an auxiliary for manual markup.

References

1. *Devlin, J., Chang, M. W., Lee, K., & Toutanova, K.* (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
2. *Wu, S., & He, Y.* (2019, November). Enriching pre-trained language model with entity information for relation classification. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management (pp. 2361–2364).
3. *Wang, L., Cao, Z., De Melo, G., & Liu, Z.* (2016, August). Relation classification via multi-level attention cnns. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 1298–1307).
4. *Zhang, S., Zheng, D., Hu, X., & Yang, M.* (2015, October). Bidirectional long short-term memory networks for relation classification. In Proceedings of the 29th Pacific Asia conference on language, information and computation (pp. 73–78).

5. *Ivanin V., Artemova E., Batura T., Ivanov V., Sarkisyan V., Tutubalina E., Smurov I.* (2020). RuREBus-2020 Shared Task: Russian Relation Extraction for Business. Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialog" [Komp'iuternaia Lingvistika i Intellektual'nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii "Dialog"], Moscow, Russia
6. *Bird, Steven, Edward Loper and Ewan Klein* (2009), Natural Language Processing with Python. O'Reilly Media Inc.
7. *Korobov M.* (2015). Morphological Analyzer and Generator for Russian and Ukrainian Languages. Analysis of Images, Social Networks and Texts (pp 320–332).
8. *Kuratov, Y., Arkhipov, M.* (2019). Adaptation of Deep Bidirectional Multilingual Transformers for Russian Language. arXiv preprint arXiv:1905.07213.
9. *Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Brew, J.* (2019). Huggingface's transformers: State-of-the-art natural language processing. ArXiv, abs/1910.03771.
10. *Grave, E., Bojanowski, P., Gupta, P., Joulin, A., & Mikolov, T.* (2018). Learning word vectors for 157 languages. arXiv preprint arXiv:1802.06893.
11. *Hendrickx, I., Kim, S. N., Kozareva, Z., Nakov, P., Séaghdha, D. O., Padó, S., ... & Szpakowicz, S.* (2010, July). Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In Proceedings of the 5th International Workshop on Semantic Evaluation (pp. 33–38). Association for Computational Linguistics.