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

Moscow, June 17–20, 2020

RENERSANs: RELATION EXTRACTION AND NAMED ENTITY RECOGNITION AS SEQUENCE ANNOTATION

Davletov A. A. (davletov-aa@ranepa.ru)[1,3],
Gordeev D. I. (gordeev-di@ranepa.ru)[1],
Rey A. I. (rey-ai@ranepa.ru)[1],
Arefyev N. V. (nick.arefyev@gmail.com)[2,3,4]

[1]RANEPA, Moscow, Russia
[2]Samsung R&D Institute Russia, Moscow, Russia
[3]Lomonosov Moscow State University, Moscow, Russia
[4]National Research University Higher School of Economics
Moscow, Russian Federation

In this work we present our system for RuREBus shared task held together with Dialog 2020 conference. The task consisted of 3 subtasks: named entity recognition, relation extraction with provided named entity tags and end-to-end relation extraction. Our system took the first and the second place in the first and the second subtasks respectively. For the third subtask we submitted our solution only in the post-evaluation phase, however, it was among the top 2 best performing systems. The systems for all tasks are based on Transformer models. Relation extraction was solved as a sequence labelling problem. We also used joint task named entity and relation extraction learning.[1]

Key words: Relation Extraction, Named Entity Recognition, Transformer, BERT

DOI: 10.28995/2075-7182-2020-19-187-197

[1] https://github.com/AdisDavletov/DeftEval2020/tree/dev

РЕНЕРСАНС: ИЗВЛЕЧЕНИЕ ОТНОШЕНИЙ И АВТОМАТИЧЕСКОЕ РАСПОЗНАВАНИЕ ИМЕНОВАННЫХ СУЩНОСТЕЙ КАК РАЗМЕТКА ПОСЛЕДОВАТЕЛЬНОСТЕЙ

Давлетов А. А. (davletov-aa@ranepa.ru)[1,3],
Гордеев Д. И. (gordeev-di@ranepa.ru)[1],
Рей А. И. (rey-ai@ranepa.ru)[1],
Арефьев Н. В. (nick.arefyev@gmail.com) [2,3,4]

[1]РАНХиГС, Москва, Россия
[2]Московский Исследовательский Центр Самсунг
[3]Московский Государственный
Университет им. М. В. Ломоносова
[4]Национальный исследовательский университет
«Высшая школа экономики», Москва, Россия

В данной работе мы представляем нашу систему для соревнования RuREBus, проводящегося совместно с конференцией Dialog 2020. Задача состояла из 3 дорожек: распознавание именованных сущностей, классификация отношений между заранее аннотированными именованными сущностями и извлечение отношений из неаннотированного текста. Наша система заняла первое место на первой дорожке и второе место на второй. Для третьей задачи мы не успели своевременно представить решение, но оно бы оказалось в числе лучших систем. Системы для всех задач основаны на моделях Transformer. Извлечение отношений мы рассматривали как задачу разметки последовательностей. Также мы использовали совместное обучение для задач распознавания именованных сущностей и извлечения отношений.

Ключевые слова: извлечение отношений, распознавание именованных сущностей, Transformer, BERT

1. Introduction

This work is devoted to our solution for RuREBus [4] shared task held together with the conference Dialog 2020. RuREBus shared task was devoted to the problem of relation extraction and named entity recognition (NER) in a specialized business domain. It consisted of three subtasks: named entity recognition, relation extraction with provided named entity labels and end-to-end relation extraction. Our first subtask solution was a BERT-based [2] sequence labelling model. For the second one we applied joint named entity and relation extraction learning. We went with a similar approach for the third subtask. However, due to having no labelled named entities, they were inferred using the model trained for the first subtask.

Our NER model with the 0.561 F1-score at the test dataset took the first place in the shared task. Our second subtask model took the second place with the F1-score equal to 0.394.

Our work shows that the sequence labelling approach is viable for relation extraction. It also demonstrates that correct named entity labels are vital for relation extraction due to the difference in scores between the second and the third subtask models.

2. Related work

There are many ways to extract information from text. This task is often solved by extracting named entities and classifying relations between them. One of the most popular datasets for this task is TACRED [10] where semantic relations are understood as relations between two pairs of entities.

Nowadays, state-of-the-art results for this dataset are achieved with Transformer-based models [8]. The most advanced models (according to paperswithcode[2]) use extra training data or additional knowledge bases. For example, in the state-of-the-art system the authors use Wikipedia data [1]. However, such data is useless for domain-specific relations.

Among the systems that do not use encyclopedias or other labeled data, the best results were achieved by Joshi et al. [5]. They pre-trained a BERT-like system, but instead of predicting individual masked tokens they trained the model to infer contiguous random spans. The model was also trained to predict each token in the masked span using output representations of only span boundary tokens. This significantly improved results of their model in comparison with the vanilla BERT. As in both described works we also incorporated information about named entity spans.

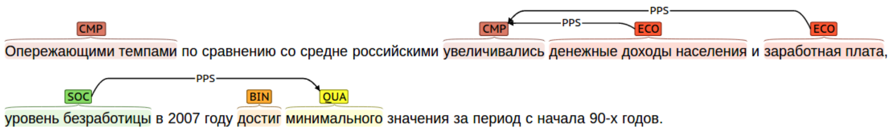
However, it is difficult to compare results for relation extraction systems for languages besides English (including Russian) because such annotated datasets are scarce for them. Some researchers have tried to solve this problem using unsupervised language-agnostic approaches and relying on knowledge databases such as Wikidata and various online encyclopedias such as Wikipedia [3]. Models trained this way tend to be not specialized because the original database does not contain relations from the required domain. The results are good only for the most popular relation types such as geographical or professional ones, which frequently appear in Wikipedia.

3. Shared task overview

The organizers of the shared task have provided approximately 300 annotated texts in total. All texts were provided by the Ministry of Economic Development of the Russian Federation. The corpus consists of various regional and strategic plan reports. There are in total 8 named entity classes and 11 semantic relation classes (see **Tables 1** and **2**). The organizers have also provided a large unannotated dataset for

[2] https://paperswithcode.com/sota/relation-extraction-on-tacred

language model fine-tuning. However, we did not use it. A named entity can consist of several words. All entities and relations do not span across sentences. There may be many-to-many, many-to-one and other types of relations (see Fig. 1).



Опережающими темпами по сравнению со средне российскими увеличивались денежные доходы населения и заработная плата, уровень безработицы в 2007 году достиг минимального значения за период с начала 90-х годов.

Figure 1: RuREBus annotation example

Named entity groups could contain rather broad types of entities, for example "SOC" entities contained social groups as well as various social attributes—phrases like 'blue collar workers' and 'housing accessibility' corresponded to this group.

Table 1: Named entity types

Type	Description	Examples
MET	Some quantitative metric	доля сельского населения (rural population ratio); положение в округе (ranking in the neighbourhood)
ECO	An economy entity or facility	обрабатывающим сектором промышленности (processing industry); экономического кризиса (economic crisis)
BIN	A binary attribute	входит в состав (is part of)
CMP	Comparative attribute	рост (growth); увеличился (increased); в наибольшей степени (to the greatest extent)
QUA	Qualitative attribute	лидирующее (leading)
ACT	Activity, actions, implemented policies	восстановление экономики региона region economy reconstruction
INST	Institutions and organizations	Алтайского края (Altai region); Сибири (Siberia)
SOC	Social groups and characteristics	населения края (region population); здравоохранение (health care)

Table 2: Semantic relation types

Group	Type	Description
Current state of affairs	NNG	now negative
Current state of affairs	NNT	now neutral
Current state of affairs	NPS	now positive
Results	PNG	past negative
Results	PNT	past neutral
Results	PNS	past positive
Forecasts	FNG	future negative

Group	Type	Description
Forecasts	FNT	future neutral
Forecasts	FNS	future positive
Goals	GOL	some abstract goals
Tasks	TSK	tasks and actions performed to achieve goals

The organizers first held tracks 1 and 3 and after that track 2 was also run. We describe our solutions in the same order (first tracks 1 and 3, then track 2).

4. Named entity recognition and relation extraction as sequence labelling

The data for the shared task was presented in brat format [7] where texts were given as plain text files and annotations were provided in another file with mixed labels for named entities and relations between them. Thus, we first had to separate the labels and transform the data into special formats used by our models.

We used Razdel library to split plain texts into sentences and tokens.[3] It is a rule-based system that along with splitting sentences can also provide sentence and token offsets in the source text. Offset ranges provided by Razdel were used during preprocessing and postprocessing to map tags and relations to text spans which are required by the brat format (see **Table 3**). We had some conversion problems and the number of NER tags in the brat format did not correspond to the number of tags after processing, for the training dataset the difference was minor, but for the test dataset almost 1% of named entity tags were lost during the preprocessing stage (see **Table 3**).

Table 3: Named entity types

	Number of			
Dataset	Sentences	Tokens	processed NER tags	Original NER tags (brat)
train	10,460	336,023	54,377	54,388
test	20,483	643,668	89,006	89,879

4.1. Subtask 1: Named Entity Recognition

The first task was to annotate named entities. First we transformed the data word-wise into the BIO-format (beginning, inside, outside). We randomly split the data into training and validation datasets in 0.7 to 0.3 ratio. The split was performed text-wise. After hyperparameter tuning we did not retrain the model using both training and validation data. We used a BERT-based system [2] with PyTorch model code and pretrained weights provided by Hugging Face [9]. Due to the shared task datasets being in Russian, we used the multilingual uncased base BERT model.

[3] https://github.com/natasha/razdel

BERT is a Transformer based model [8]. On top of BERT outputs we added a linear layer and dropout regularization. The cross entropy loss function was used to train the model. BERT outputs an embedding for each token, i.e. a word may include several BPE-tokens. As we had one label for a word we needed to decide how to aggregate predictions from word tokens. We went with the easiest approach and for each word in the sentence we took a BERT embedding only from its first BPE-token and fed it to the dropout layer followed by the linear layer. All non entity tokens were ignored (i.e. padding tokens and tokens with O tag).

Table 4: Subtask 1: Results on test and development sets on single system

Model	Learning rate	Weight Decay	Dropout	test
bert	1e-5	0.1	0.2	0.548
bert	1e-5	1.0	0.2	0.547
bert	5e-5	0.01	0.1	0.552
bert	5e-5	0.01	0.2	0.555
bert	5e-5	0.1	0.1	0.555
bert	**5e-5**	**0.1**	**0.2**	**0.561**
bert	5e-5	1.0	0.1	0.554
bert	5e-5	1.0	0.2	0.56

Our system with 0.561 micro F1-score on the public leaderboard outperformed solutions presented by other contestants. In Table 4 we provide results for the test dataset that was provided by the organizers after the shared task.

4.2. Subtask 3: End-to-end Relation Extraction

The second and the third subtasks were relation classification. In the second subtask the organizers provided named entity tags while in the third they did not. For both tracks we used the equivalent approach.

Akin to BERT-multitask learning, in this shared task we wanted to experiment with simultaneous finetuning for separate tracks. RuREBus shared task provided an excellent framework for this idea because we had separate tracks with different target values but the same input data. Thus, we tried a multitask architecture to jointly predict tags and relations. To do so, we consider relation extraction as a sequence labeling problem (similar to how named entity recognition is usually solved). In each example we have one marked main entity and we predict all named entity tags and all relations between the main token and all other tokens in the sentence (see Fig. 2). We put an empty relation label ('0') if a token does not have relation to the marked entity and the relation tag otherwise. Special tokens marking the beginning and the end of the main entity are added to input to tell the system which entity it should predict relations with. Thus, for each sentence we had to make *n* predictions where *n* is the number of named entities in the sentence. We did not relabel previously inferred named entity tags with new predictions.

Sequence labelling might be a preferable solution if we are interested in processing the whole sentence in a smaller number of batches and our algorithm runtime does not depend on the sequence length (unlike recurrent neural networks).

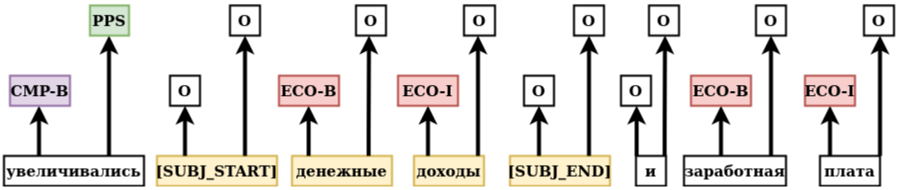


Figure 2: Joint relation extraction and named entity recognition training

Table 5: Subtask 3: F1 scores on test set. Models marked with * were trained after the competition

model	Learning rate	Weight decay	Dropout	NER loss weight	Relation loss weight	test
bert	1e-5	0.1	0.1	0	1.0	0.121
bert	**1e-5**	**0.1**	**0.1**	**0.1**	**1.0**	**0.127**
*xlm-r	1e-5	0.1	0.1	0	1.0	0.194
*xlm-r	1e-5	0.1	0.1	0.01	1.0	0.187
*xlm-r	1e-5	0.1	0.1	0.05	1.0	0.193
*xlm-r	1e-5	0.1	0.1	0.1	1.0	0.182
*xlm-r	1e-5	0.1	0.1	0.2	1.0	0.157
*xlm-r	1e-5	0.1	0.1	1.0	0	0.002
*xlm-r	1e-5	0.1	0.1	1.0	1.0	0.128
*xlm-r	1e-5	0.1	0.2	0.1	1.0	0.169
*xlm-r	1e-5	0.2	0.1	0.1	1.0	0.173
*xlm-r	2e-5	0.1	0.1	0	1.0	0.188
*xlm-r	5e-5	0.1	0.2	0	1.0	0.171
*xlm-r	5e-5	0.2	0.2	0	1.0	0.113

For end-to-end relation extraction we went with a two-stage approach. At first we used the model from the first track to label named entities. After that using the provided named entity predictions we trained our model to infer semantic relations.

In this task we used the same multilingual uncased BERT model as in subtask 1. However, to get simultaneous relation and named entity predictions on top of the model we added another dropout layer followed by tag and relation linear layers. We use weighted sum of cross entropy losses for tag and relation labeling as our final loss for optimization. Padding tokens do not contribute to our loss calculation.

The system showed 0.132 micro F1-score using public test data and it would have taken the first place among the provided systems, if we had managed to submit our solution before the deadline. Joint task learning has slightly improved our results. After the shared task end we also tried RoBERTa base model instead of BERT. It improved our results but the model did not benefit from joint task learning (see Table 5). Our local evaluation results using the test dataset are slightly worse than the results at the public leaderboard.

4.3. Subtask 2: Relation Extraction for given Named Entities

The model for this track is equivalent to the system used for end-to-end relation extraction. This track was very similar to end-to-end relation extraction. However, instead of using named entity labels predicted by our model, we could use the manual annotation provided by the organizers of the shared task.

For subtask 2 we also tried a base XLM-RoBERTa [6] model also provided by Hugging Face. RoBERTa is BERT inspired model which optimized many hyper-parameter choices in the underlying model. RoBERTa authors have replaced static masking with random masking during language training. They also removed additional sentence prediction loss, increased the batch size, trained on longer sequences and enhanced the original Wikipedia dataset with various Common Crawl datasets. All these adjustments helped RoBERTa to outperform BERT in many benchmarks such as GLUE or SQuAD 2.0.

Table 6: Subtask 2: Results on test and development sets for relation extraction and additional named entity subtask

Model	Learning rate	Weight decay	Dropout	NER loss weight	Relation loss weight	Relation extraction F1		NER F1	
						test	dev	test	dev
bert	1e-5	0.1	0.1	0	1.0	0.25	0.784	0.002	0.001
bert	1e-5	0.1	0.1	0.1	1.0	0.263	0.757	0.189	0.172
xlm-r	1e-5	0.1	0.1	0	1.0	0.391	0.678	0.040	0.049
xlm-r	1e-5	0.1	0.1	0.01	1.0	0.381	0.677	0.330	0.294
xlm-r	1e-5	0.1	0.1	0.05	1.0	0.39	0.685	0.482	0.440
xlm-r	1e-5	0.1	0.1	0.1	1.0	0.379	0.667	0.503	0.468
xlm-r	1e-5	0.1	0.1	0.2	1.0	0.34	0.662	0.501	0.492
xlm-r	1e-5	0.1	0.1	1.0	0	0.006	0.022	0.465	0.463
xlm-r	1e-5	0.1	0.1	1.0	1.0	0.271	0.679	0.489	0.456
xlm-r	1e-5	0.1	0.2	0.1	1.0	0.355	0.668	0.497	0.465
xlm-r	1e-5	0.2	0.1	0.1	1.0	0.357	0.668	0.004	0.001
xlm-r	**2e-5**	**0.1**	**0.1**	**0**	**1.0**	**0.394**	**0.675**	**0.004**	**0.001**
xlm-r	5e-5	0.1	0.2	0	1.0	0.369	0.65	0.059	0.067
xlm-r	5e-5	0.2	0.2	0	1.0	0.272	0.599	0.010	0.0187

In this task our best model with the F1 score equal to 0.394 took the second place.

RoBERTa-based models outperformed BERT-based ones. As we did not include named entity type information in the input, but only spans, we also attempted at using the multi-task learning procedure described in the previous section. However, unlike the previous case the quality deteriorated when the model was trained to predict named entity tags. Thus, the loss coefficient for named entity recognition was set to zero in the final model. Learning rate, weight decay and other hyperparameters you may see in Table 6.

5. Results

All in all, our named entity recognition model with micro F1-score equal to 0.561 took the first place in the shared task. However, the results are lower than for other named entity recognition datasets (e.g. for the Ontonotes dataset Transformer-based models usually get > 0.85 in F1-score[4]). It can be attributed to the small number of training examples and complexity of the domain. While training relation extraction models we also did not use information about named entity types. The authors of SpanBERT [5] claim that it may improve model scores. It can be further investigated in future work.

Our end-to-end relation extraction model despite being one of the best solutions at the shared task was much worse than the model trained with manual annotations provided by the organizers. In future we will try to use approaches similar to pseudo labelling where we include only those named entity predictions that have high logit scores instead of all predictions. The difference in results also demonstrates that correct named entity labels are vital for relation extraction. The results may also benefit from a larger dataset.

Multi-task learning improved our results only in end-to-end relation extraction for BERT-based models. RoBERTa was better than BERT in this shared task in both relation extraction subtasks.

Some typical errors may be found in Table 7. All our models tend to generate a lot of false positives, e.g. for a single text we predicted 165 relations instead of 88 and 421 named entities instead of 354. This might be tweaked in future. Sometimes these mislabellings might be due to errors in the dataset. For relation extraction such mistakes tend to snowball. Another typical NER mistake is span mismatch, usually it is due to the model not including the whole phrase group into the prediction.

Table 7: Error examples

Task	Prediction	Test	Comment
NER	администрации Костромской области	—	Might be alabelling mistake
NER	производства бумаги	производства бумаги и бумажных изделий, прочих готовых изделий, прочих транспортных средств и оборудования	Only partial span overlap
NER	—	действующих производств	Not found
End-to-end RelEx	динамика региональной экономики -> NPS -> Положительная	динамика региональной экономики -> NPS -> Положительная	Correct
End-to-end RelEx	динамика региональной экономики -> NPS -> Экспорт	—	Relation does not exist

[4] see http://docs.deeppavlov.ai/en/master/features/models/ner.html

6. Conclusion

In this work we present our system for RuREBus shared task held together with Dialog 2020 conference. The task consisted of 3 tracks: named entity recognition, relation extraction with provided named entity tags and end-to-end relation extraction. All tracks were considered as sequence labelling problems. We show that sequence labelling might be a decent approach for the relation extraction problem. We also attempted to use joint-task learning for relation extraction and named entity recognition. However, it only slightly improved our results for end-to-end relation extraction and was outperformed by single task learning in most cases. Yet it should be noted that in other unrelated domains it enhanced results of our models so joint tasks should be carefully chosen and it requires further investigation. The system took the first place in the named entity recognition track and the second place in the third track. For the second task we failed to submit the solution till the deadline but it was among the best systems. The systems for all tasks are based on Transformer models.

7. Acknowledgments

We would like to thank the organizers of the shared task. We believe that their work will be very helpful for the development of natural language processing for the Russian language. The contribution of Nikolay Arefyev to the paper was partially done within the framework of the HSE University Basic Research Program funded by the Russian Academic Excellence Project ‘5-100’.

References

1. *Baldini Soares, L. et al.:* Matching the Blanks: Distributional Similarity for Relation Learning. In: Arxiv.org. pp. 2895–2905 (2019).
2. *Devlin, J. et al.:* BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. (2018).
3. *Heist, N., Paulheim, H.:* Language-agnostic relation extraction from wikipedia abstracts. In: Lect. Notes comput. Sci. (Including subser. Lect. Notes artif. Intell. Lect. Notes bioinformatics). pp. 383–399 (2017).
4. *Ivanin V., Artemova E., Batura T., Ivanov V., Sarkisyan V., Tutubalina E., Smurov I.:* 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 (2020).
5. *Joshi, M. et al.:* SpanBERT: Improving Pre-training by Representing and Predicting Spans.
6. *Liu, Y. et al.:* RoBERTa: A Robustly Optimized BERT Pretraining Approach. arxiv.org. (2019).
7. *Stenetorp, P. et al.:* brat: a Web-based Tool for NLP-Assisted Text Annotation. In: Proc. Demonstr. Sess. EACL 2012. Association for Computational Linguistics, Avignon, France (2012).

8. *Vaswani, A. et al.:* Attention is all you need. In: Adv. Neural inf. Process. Syst. pp. 5999–6009 (2017).
9. *Wolf, T. et al.:* HuggingFace's Transformers: State-of-the-art Natural Language Processing. ArXiv. abs/1910.0, (2019).
10. *Zhang, Y. et al.:* Position-aware attention and supervised data improve slot filling. In: EMNLP 2017—conf. Empir. Methods nat. Lang. Process. Proc. pp. 35–45 (2017).