Joint task learning for relation extraction and named entity recognition

Authors

Institution

In this work we present our system for RuREBus challenge 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 third subtasks respectively. For the second subtask we submitted our solution two hours after the deadline but before the test data release. It would also have been among the best 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.

Key words: relation extraction, named entity recognition, transformer, bert

Совместное обучение моделей для извлечения отношений и именованных сущностей

Авторы

Организация

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 третьей. Для второй 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.

Relation extraction was solved as a sequence labelling problem. We also used joint task named entity and relation extraction learning.

Ключевые слова: извлечение отношений, распознавание именованных сущностей, transformer, bert

1 Introduction

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 [9] where semantic relations are understood as relations between two pairs of entities.

Nowadays, state-of-the art results for this dataset are achieved by using Transformer-based models [7]. The most advanced models (according to paperswithcode [1]) use extra training data or additional knowledge bases. For example, the authors of the leading paper "Matching the Blanks: Distributional Similarity for Relation Learning" [1] use Wikipedia data which is infeasible for domain-specific relations.

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

Among the systems that do not use encyclopedias or other labeled data, the best results were achieved by Joshi et al. [4]. They pre-trained a BERT-inspired system. In their work 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.

However, it is difficult to compare results for relation extraction systems for languages besides English because such annotated datasets are scarce for most languages including Russian. 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. They also tend to work only for the most popular relation types such as geographical or professional ones which are common to Wikipedia.

There are few annotated datasets for the Russian language. Among similar tasks to relation extraction there was held FactRuEval 2016 within the conference Dialog 2016. During the competition contestants had to extract facts from news articles and to fill special slots in these facts (e.g. one of the fact types was 'Occupation' and its fields were 'POSITION', 'WHO', 'WHERE' and 'PHASE').

RuREBus competition was devoted to the problem of relation extraction and named entities recognition in a specialized business domain.

2 Shared task overview

The organizers of the competition 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 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).

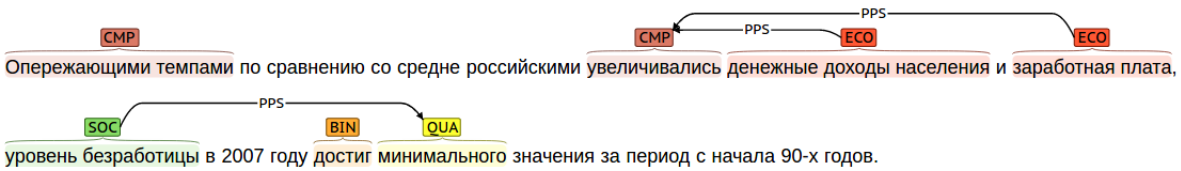

Figure 1: RuREBus annotation example.

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.

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).

3 Our solution

The data for the competition was presented in brat format [6] where texts were given as plain txt files and annotations were provided in another file with mixed labels for named

Type	Description
MET	Some quantitative metric
ECO	An economy entity or facility
BIN	A binary attribute
CMP	Comparative attribute
QUA	Qualitative attribute
ACT	Activity, actions, implemented policies
INST	Institutions and organizations
SOC	Social groups and characteristics

Table 1: Named entity 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
Forecasts	FNT	future neutral
Forecasts	FNS	future positive
Goals	GOL	some abstract goals
Tasks	TSK	tasks and performed actions to achieve goals

Table 2: Semantic relation types

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 [2]. It is a rule-based system that despite 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.

After dataset tokenization we got 10460 sentences containing 336023 tokens in the train set and 20483 sentences with 643668 tokens in the test set. There are 54377 and 89006 named entity tags in the train and test sets respectively (see Table 3).

Dataset	Number of			
	Sentences	Tokens	NER tags	Original NER tags
train	10460	336023	54377	54388
test	20483	643668	89006	89879

Table 3: Named entity types

3.1 Named entity recognition: track 1

The first task was to annotate named entities. First we transformed the data into the CONLL-2003 format where each line contained a word and its named entity tag. Sentences were separated with newlines. All texts were united in a single file where individual texts

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

were divided with two empty lines. We split the data into training and validation datasets. We used a BERT-based system [2] with PyTorch model code and pretrained weights provided by Hugging Face [8]. Due to competition being in Russian, we used the multilingual uncased base BERT model.

BERT is a Transformer based model [7]. On top of BERT outputs we added a linear layer with softmax activation function and dropout regularization. The cross entropy loss function was used to train the model. For each word token in the sentence we took BERT embedding 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 describing borders between sentences and various spans).

<DEV RESULTS + HYPERPARAMETER RESULTS>

Our system with 0.561 micro F1-score on the public leaderboard outperformed solutions presented by other contestants.

3.2 End-to-end relation extraction: track 3

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

Akin to BERT-multitask learning, in this competition we wanted to experiment with simultaneous finetuning for separate tracks. RuREBus competition 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 ending 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 limiting the number of model calls and our model run time does not depend on the sequence length (unlike recurrent neural networks).

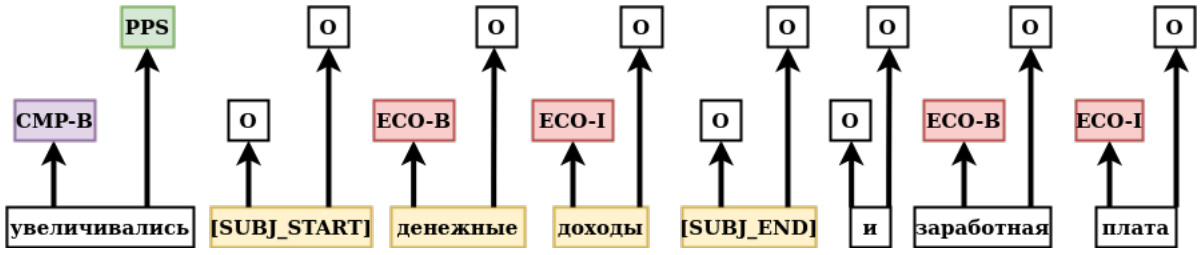

Figure 2: Joint relation extraction and named entity recognition training.

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.

However, joint task learning only worsened our results and the best result at the validation set was

<VALIDATION TEST RESULTS>

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.

3.3 Relation classification with provided named entity tags: track 2

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 competition.

We also attempted at using the multi-task learning procedure described in previous section. However, as in 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 also set to zero.

For subtask 2 we also tried a base XLM-RoBERTa [5] 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 alse 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.

<VALIDATION DETAILS>

4 Results

All in all, our named entity recognition model with micro F1-score equal to 0.561 took the first place in the competition. 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 [3]). It can be attributed to the small nmmber of training examples and complexity of the domain.

Our end-to-end relation extraction model despite being one of the best solutions at the competition 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.

Multi-task learning did not improve our results for this task as well.

5 Conclusion

In this work we present our system for RuREBus challenge held together with Dialog 2020 conference. The task consisted of 3 tracks: named entity recognition, relation extraction with

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

provided named entity tags and end-to-end relation extraction. All tracks we 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. However, it did not improve our results on the validation dataset. 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.

6 Acknowledgments

We would like to thank the organizers of the competition. We believe that their work will be very helpful for the development of natural language processing for the Russian language.

References

[1] Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. Matching the Blanks: Distributional Similarity for Relation Learning. In *arxiv.org*, pages 2895–2905, 2019.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. oct 2018.

[3] Nicolas Heist and Heiko Paulheim. Language-agnostic relation extraction from wikipedia abstracts. In *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, volume 10587 LNCS, pages 383–399, 2017.

[4] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, Omer Levy, and † Allen. SpanBERT: Improving Pre-training by Representing and Predicting Spans. Technical report.

[5] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arxiv.org*, 2019.

[6] Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. brat: a Web-based Tool for NLP-Assisted Text Annotation. In *Proc. Demonstr. Sess. EACL 2012*, Avignon, France, 2012. Association for Computational Linguistics.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Adv. Neural Inf. Process. Syst.*, volume 2017-Decem, pages 5999–6009, 2017.

[8] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *ArXiv*, abs/1910.0, 2019.

[9] Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D Manning. Position-aware attention and supervised data improve slot filling. In *EMNLP 2017 - Conf. Empir. Methods Nat. Lang. Process. Proc.*, pages 35–45, 2017.