PROCEEDINGS OF SCIENCE

Evaluation of Machine Learning Methods for Relation Extraction Between Drug Adverse Effects and Medications in Russian Texts of Internet User Reviews

Alexander Sboev,[a,b] **Anton Selivanov,**[a,*] **Roman Rybka,**[a] **Ivan Moloshnikov**[a] **and Gleb Rylkov**[a]

[a] *National Reseatch Center "Kurchatov Institute", 1 Akademika Kurchatova Sq., Moscow, Russian Federation*

[b] *National Research Nuclear University "MePhI", 31 Kashira Hwy, Moscow, Russian Federation*

E-mail: sag111@mail.ru

The research considers an automatic extraction of relations between mentions of medications and adverse drug reactions in Russian-language drug reviews. This text analyzing method might be useful for pharmacovigilance and medicines reprofiling. Its application to Russian-language reviews hasn't been studied yet due to the lack of corpora with relation annotation in Russian. The study is aimed at solving this problem. It is based on the original dataset gathered by our group. It consists of annotated relations between entities from the Russian Drug Review Corpus, that contains the Internet users' reviews on medications in Russian language. Computational experiments were carried out on developed corpora using classical machine learning methods, as well as a more advanced neural network model based on Transformer layers – XLM-RoBERTa-sag. The list of applied classical machine learning methods consists of support vector machine, logistic regression, Naive Bayes classifier and gradient boosting. The concatenation of TF-IDF entity vectors of character n-grams was used as a text representation. Based on a set of experiments, the following hyperparameters of these methods were selected: the size of n-grams and the limitation on the frequency of occurrence of n-grams (too rare or too frequent n-grams were excluded from the feature vector). For XLM-RoBERTa-sag, the input data is represented as usual for such type of models (language models based on Transformer topology). The following input text representation types were considered during the experiments: a whole text, a text of target entity pairs; a text of target entity pairs with words between them; a text of target entity pairs and the whole input text, the latter input type is the one that maximizes accuracy. It is shown that XLM-RoBERTa-sag model achieves a result of 95%, according to the macro-averaged f1 metric, which is the state-of-the-art result of recognition of the relations between mentions of adverse drug reactions and medications in Russian-language online reviews. The Naive Bayes classifier with multivariate normal distribution achieves the best result among classical machine learning methods: 75%, which exceeds the result of random label generation by 21%.

The 5th International Workshop on Deep Learning in Computational Physics
28-29 June, 2021
Moscow, Russia

*Speaker

PoS(DLCP2021)006

1. Introduction

Automatic extraction of relations between named entities is a task of great importance in natural language processing. Medical relation extraction provides new information about existing medical entities: medications, diseases, adverse effects, etc. Information about the relations between medications and their adverse effects bears particular importance in terms of pharmacovigilance. Extraction of such information from texts expands the information base of known medications. It provides opportunity for enhanced quality control as well as successful reprofiling of medications. Therefore, relation extraction of the medical entities is a relevant task.

The scope of the current research is extraction of the relations from Internet users' reviews on medication in Russian language. Russian Drug Review Corpus [**?**], which was manually annotated and validated by specialists in pharmacology and machine learning, served as a dataset of the research.

The main task of the relation extraction method is to identify whether there is a link between two given entities in the text containing these entities. The application of the relation extraction method is well-studied for the English biomedical datasets, such as n2c2-2018 [6], ADE [5], DDI2013 [7], etc. There are various methods for achieving high accuracy on these datasets, like SpERT [4], bioRoBERTa [10], Table-Sequence [17], and others. However, there are some downsides of these methods, such as their high complexity and high requirements to computational resources. Thus, it's reasonable to get baselines using less complex methods first.

The main goal of the research was to establish baseline accuracy for the task of medical relation extraction on Russian-language reviews dataset. Comprehensive baseline results were achieved using the following methods: stratified random label generation as a basic "dummy" method to start with, basic machine learning methods (Logistic Regression, Support Vector Machine, Multinomial Naive Bayes Model, Gradient Boosting of Decision Trees) with limited input information (tf-idf n-grams of the text of the target entities), multilingual model additionally pre-trained on Russian texts about medicines (XLM-RoBERTa-sag [**?**]).

2. Materials and Methods

2.1 Methods

2.1.1 "Dummy" Models

"Dummy" models were considered to be the low-level baseline. Such models generate labels randomly or according to some simple principle. The following methods were checked as methods for "dummy" classification:

- most frequent class labeling – each prediction is the most frequent class in the dataset (in case of extraction of relations between adverse reaction and medication in Russian Drug Review dataset it counts each input example as an example with relation);

- uniform random labeling – labels are predicted randomly according to a uniform probability distribution, therefore without taking into account any characteristics of the input dataset;

- stratified random labeling – labels are predicted randomly, but unlike the previous option, the probability distribution is similar to the one in the training data.

The accuracy of the "dummy" methods based on the random label generation was averaged over 100 launches in order to operate with more stable results and prevent the occurrence probability of random outliers.

2.1.2 Basic Machine Learning Methods

Basic machine learning methods perform decently in many applications [13][2][18]. These methods are highly efficient in terms of computational complexity. Due to that fact it is possible to search for the optimal set in an extensive space of hyperparameters and to test hypotheses relatively quickly.

The first goal of using basic machine learning methods was to obtain a strong baseline for relation extraction of medical entities in Russian language that exceeds "Dummy" models' results.

As a text data representation for the baseline on basic machine learning methods we used: concatenation of a frequency features (tf-idf) of the character n-grams of the target entities. The size of the n-gram n and the frequency filter of tf-idf were considered as the hyperparameters to tune during the experiments.

The second goal of using basic machine learning methods was to check if the information about the entities' text is sufficient to achieve competitive accuracy for the task.

The following methods were used during the experiments with basic machine learning:

- Logistic regression [8] – a basic linear model for text classification using a logistic function to estimate the probability of an example to be of a certain class;
- Support vector machine [15] – a linear model based on building hyperplane that maximizes the margin between two classes;
- Multinomial Naive Bayes model [12] – a popular solution for baselines in such text analysis tasks as spam filtering or text classification. It performs text classification based on words'/n-grams' co-occurrence probability;
- Gradient Boosting [11] – a strong decision tree-based ensemble model, which iteratively "boosts" the result of each tree by building a next tree, that should classify examples that the previous tree did not classify correctly.

2.1.3 Deep Language Model

In this work we used XLM-RoBERTa-sag model described in our previous work [**?**]. The model XLM-RoBERTa [3] is additionally trained on a dataset of unlabeled internet texts about medicines (~1.65M texts). Original XLM-RoBERTa is a multilingual language model based on Transformers [16] that was trained on a larger multilingual corpus from the CommonCrawl project which contains 2.5TB of texts. The model contains more than 550M parameters.

This type of models based on the Transformer topology [16], that consists of multihead attention layers, which create vector representations of input data parts (words in case of NLP) that encode information about their context.

Such models are state-of-the-art in many of the modern natural language processing tasks. In English, additional training on domain-specific texts ensures maximum accuracy in solving medical-related natural language processing tasks [10].

Text pre-processing includes text splitting into words or word parts – "tokens". In the case of XLM-RoBERTa-sag SentencePiece tokenizer [9] is used.

It's necessary to use the same tokenizer with pre-trained language models to take advantage out of preliminary training, because only in this case words of input text will be represented with vectors formed on base of model vocabulary.

The advantages of these models are:

- such model contains a vast massive of information about language constructions as a whole;
- in case of preliminary training on domain-specific texts models also contain information useful in particular tasks;
- the opportunity to fine-tune the model on the specific task and/or dataset ensures even higher model accuracy.

The disadvantages of such models include:

- a high number of model weights makes learning procedures hard to tune for efficient task solving in the case of fine-tuning;
- a high number of hyperparameters and possible learning techniques makes it more difficult to find the optimal combination to achieve the highest possible accuracy, even in fine-tuning stage only;
- the pre-trained model requires significantly more memory space than basic machine learning methods;
- such language model has squared dependency of algorithm complexity on input size. To overcome this issue, a limitation on the length of the input sequence is used – in the case of XLM-RoBERTa-sag this limitation can't exceed 512 tokens, which could be too restrictive in the case of long texts.

Nevertheless, language models are currently considered to be standard in modern natural language processing, so another goal of the research was to achieve results based on such models.

We used two versions of the model:

- XLM-RoBERTa-base-sag – 12 Transformer blocks, 768 hidden neurons, 8 Attention Heads, 125 millions of parameters, 2 epochs of additional training on Russian texts about medications;
- XLM-RoBERTa-large-sag – 24 Transformer blocks, 1024 hidden neurons, 16 Attention Heads, 355 millions of parameters, 1 epoch of additional training on Russian texts about medications;

To solve classification task such models use special token added to input sequence: [CLS]. During the input data processing this token accumulates information about text as a whole. At the training stage model weights are adjusted to the state, in which they create vector representation of [CLS] token informative in terms of current task to solve, in other words, they achieve small values of loss function during the class prediction based on the vector of [CLS] token.

As it was mentioned before, there are many degrees of freedom in such models that require consideration in order to achieve higher accuracy, in scope of the current research the following options were analyzed:

- usage of a fine-tuning procedure;
- entities highlighting – it is a way to highlight parts of texts with high importance in a way that the model "notices" them. Several types of entity highlighting were considered:

  No highlighting.

  Example: Yesterday I've got a shot of Sputnik-V, now I feel a little fever, but I heard it's still better than EpiVAC;

  Highlighting of entities of target pair only.

  Example: Yesterday I've got a shot of [T_MED]Sputnik-V[\T_MED], now I feel a little [T_ADR]fever[\T_ADR], but I heard it's still better than EpiVAC.

  Highlighting of all entities.

  Example: Yesterday I've got a shot of [T_MED]Sputnik-V[\T_MED], now I feel a little [T_ADR]fever[\T_ADR], but I heard it's still better than [1_MED]EpiVAC[\1_MED].

- adding special "highlighting" tokens to language model vocabulary – to guarantee that these special tokens would be in language model vocabulary, and thus would be interpreted as particular entities, it's necessary to add these tokens to model vocabulary. After that they would have random vector representation, but during the fine-tuning their representation will be changed to better fit the task;
- early stopping technique using [1];
- maximum input sequence length (in tokens);
- learning rate;
- batch size;
- maximum learning epoch number;
- learning rate decay technique using [14].

The following text representation variants were considered during the experiments:

- the whole text – tokenized input text was used as an input;
- a text of target entities only – same as in basic machine learning, only text of the target entities was used as input data;
- a text of target entities, separation token, and a text between target entities – as the previous one, but concatenated with the text between target entities to give context;
- a text of target entities, separation token, and the whole text – text of target entities concatenated with the whole text – this way of representation highlights target entities for language model.

2.2 Dataset

We used Russian Drug Review Corpus [**?**] — a dataset of user reviews on medications that were manually annotated by experts with selection of huge number of entities, such as: disease, symptoms, drug names, forms of drugs, etc. Additionally relationships annotation is presented for the most entity types. The dataset was annotated by experts in the pharmaceutical industry according to a guideline drawn up in conjunction with machine learning specialists.

In this work we focused on extracting the relations between adverse drug reactions and medication names as the most important for pharmacovigilance. The texts of the RDRS corpus that contain ADR and Medication entities were divided into training and test parts, the composition of which is presented in Table 1.

Table 1: Statistics on the RDRS dataset part with ADR-Drugname relations

Number of	Train	Test
Texts	502	126
Sentences	4016	1008
Words	82425	20961
“ADR” type entities	1461	356
“Drugname” type entities	1416	368
Relations	3444	845
Avg. numbers of relations per text	6.9	6.7

3. Experiments

3.1 Experiment Setup

We used the following experimental setup:

- Fixed stratified split on train (80%) and test (20%) sets; Relations from single review could be in one set only to avoid overfitting and to get reliable results;

- Hyperparameters of basic machine learning methods were chosen on a test set, the main reason for that was the desire to obtain maximum theoretical accuracy as a baseline to overcome;
- Hyperparameters of language model fine-tuning were chosen on the validation part of a train set to estimate the model accuracy in circumstances close to real;
- Language model used early stopping and learning rate decay (Experiments show positive effect on model accuracy of such techniques);

"Dummy" models and basic machine learning methods experiments were carried out on a local machine with the following configuration: CPU Intel® Core™ i5-7400 @ 3.00GHz × 4, 16 Gb RAM. Experiments on language model were carried out using compute cluster node with the following configuration: CPU Intel® Xeon™ E5-2650v2 (2.6 GHz) x 8, 128 Gb RAM, NVIDIA Tesla V100 (16 Gb).

Macro-averaged f1-score metric was used to estimate machine learning models for relation extraction. This method of averaging f1-score for each class takes into account an imbalance between classes in a dataset, and to obtain an estimation with respect to class representativeness.

3.2 Hyperparameter Space for Tuning

The following hyperparameter space was used during the experiments on basic machine learning methods in search for optimal setup to maximize accuracy:

- n-gram size range: (1,1), (3,3), (1,3), (3,5), (3,8), (3,10);
- tf-idf frequency filter on rare n-grams, lower than: 0.5%, 1%, 5%;
- tf-idf frequency filter on frequent n-grams, higher than: 99.5%, 99%, 95%

The hyperparameters of the language model were searched manually in consequential experiments with the analysis of train and validation loss during the training phase, so language model hyperparameters were set manually on base of the validation accuracy, without taking into account the accuracy on a test set.

4. Results

The results of experiments are aimed on obtaining baselines for relation extraction of medications and adverse effects in Russian Drug Review Corpus. The Table 2 presents baselines on the best out of "dummy" algorithms, and basic machine learning algorithms.

The following definitions were used in the table:

- "n-min" – minimum size of the character n-grams to calculate tf-idf for entities representation;
- "n-max" – maximum size of the character n-grams to calculate tf-idf for entities representation;
- "min freq" – term frequency filter that excludes n-grams that are too rare;

Table 2: Accuracy comparison of "dummy" and basic machine learning methods

	n-min	n-max	min freq	max freq	f1-macro
Stratified random class (dummy model) (average on 100 executions)	-	-	-	-	0.54
Logistic Regression	1	3	0.050	0.950	0.72
Support Vector Machine	1	3	0.050	0.950	0.74
Multinomial Naive Bayes	1	3	0.005	0.995	**0.75**
Gradient Boosting	1	3	0.050	0.950	0.74

- "max freq" – term frequency filter that excludes n-grams that occur too often (for example, popular conjunctions);
- "f1-macro" – macro-averaged f1-measure metric for model estimation.

The features presented in the table were determined during the hyperparameter space search for optimal setting to maximize macro-averaged f1-score.

The table shows that information about the entities text only is sufficient for achieving relatively high accuracy for the relation extraction task in comparison with "dummy" baseline. The best model out of the basic machine learning methods is based on Multinomial Naive Bayes, that achieves 0.75 macro-averaged f1-score, which is 21% higher than "dummy" method, used as a simple baseline. Achieved accuracy could be used as a baseline for further experiments.

In this section we present the results of experiments on the application of the neural network approach. Table 3 shows results on searching for optimal configuration of the values of the following features: fine-tuning, entities highlighting method, adding special screen tokens to the vocabulary. All the experiments in the table used a whole text as an input. "LM-base" stands for XLM-RoBERTa-base-sag, "LM-large" for XLM-RoBERTa-large-sag.

Table 3: Accuracy comparison of different feature sets on language model with full text input

Group	Feature	LM-base f1-macro	LM-large f1-macro
Without fine-tuning, no special tokens	All entities highlighted	0.21	0.44
	Target entities highlighted	0.18	0.44
Without fine-tuning, with special tokens	All entities highlighted	0.18	0.44
	Target entities highlighted	0.18	0.44
With fine-tuning, no special tokens	All entities highlighted	0.47	0.70
	Target entities highlighted	0.77	0.75
With fine-tuning, with special tokens	All entities highlighted	0.72	0.63
	Target entities highlighted	**0.78**	**0.82**

The results in the table are the best obtained during the set of experiments with different hyperparameters' values. Resulting values for XLM-RoBERTa-base-sag are:

- maximum input length – 512;
- early stopping active;
- learning rate – 0.00005;
- batch size – 32;
- maximum epochs – 10;
- learning rate decay active;

Resulting hyperparameters' values for XLM-RoBERTa-large-sag are:

- maximum input length – 512;
- early stopping active;
- learning rate – 0.00001;
- batch size – 8 (there wasn't enough memory for bigger batch size with XLM-RoBERTa-large);
- maximum epochs – 10;
- learning rate decay active;

Obviously, fine-tuning yields higher accuracy. Highlighting target entities yields higher accuracy than highlighting all entities (one possible explanation is that highlighting the text saturated with entities created too much noise from the special tokens). Resulting conclusion on this set of experiments is that the usage fine-tuning and target entities highlighting maximizes accuracy on the language model, but this result is lower than one of basic machine learning models. The further experiments were aimed at finding a text representation form efficient enough to overcome the baseline set by basic machine learning models.

Table 4 contains results on the experiments with different text representation methods.

Table 4: A comparison of language model accuracy with different methods of text representation

Text representation	LM-base f1-macro	LM-large f1-macro
Whole text	0.78	0.82
Text of target entities only	0.75	0.76
Text of target entities and text between them	0.81	0.80
Text of target entities and the whole text	**0.91**	**0.95**

The hyperparameters in experiments from the Table 4 are the same as in experiments from the Table 3.

The table shows that both the information about the target entities separated from the text and the entire text are important for achieving high accuracy and for overcoming basic machine

learning methods. Input representation as an entity-only text concatenated with whole text makes it possible to achieve macro-averaged f1-score equal to 95%, which is 41% higher than random class prediction, 20% higher than basic machine learning models with hyperparameters tuned on a test set.

5. Conclusions

The best-achieved accuracy (macro-averaged f1) with language model is 95%, which is 41% higher than the random class prediction, 20% higher than basic machine learning model (Multinomial Naive Bayes Classifier) with hyperparameters tuned on a test set.

The best accuracy achieved with the text representation includes target entity text, separation token, and the whole text. Hypothetically, it yields a specific vector representation of each entity during the training process.

The baseline obtained for pharm entities extraction of relations between medications and adverse effects is 54% (based on stratified random classification "dummy" algorithm with the random class prediction);

It is shown that basic machine learning algorithms that use target entities character n-grams TF-IDF with frequency filtering achieve accuracy equal to 75% (Multinomial Naive Bayes), which is 21% higher than by random class prediction.

These results could be an important step in the analysis of medical texts in Russian, particularly in the extraction of relations between medications and their adverse effects.

Our further goal is to improve achieved accuracy by using more complex methods and apply deep language models in case of jointly defining named entities and their relationships.

Acknowledgments

This work has been supported by the Russian Science Foundation grant 20-11-20246 and carried out using computing resources of the federal collective usage center Complex for Simulation and Data Processing for Mega-science Facilities at NRC "Kurchatov Institute", http://ckp.nrcki.ru/.

References

[1] Rich Caruana, Steve Lawrence, and Lee Giles. Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. *Advances in neural information processing systems*, pages 402–408, 2001.

[2] Pang-jo Chun, Shota Izumi, and Tatsuro Yamane. Automatic detection method of cracks from concrete surface imagery using two-step light gradient boosting machine. *Computer-Aided Civil and Infrastructure Engineering*, 36(1):61–72, 2021.

[3] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*, 2019.

[4] Markus Eberts and Adrian Ulges. Span-based joint entity and relation extraction with transformer pre-training. In *ECAI 2020*, pages 2006–2013. IOS Press, 2020.

[5] Harsha Gurulingappa, Abdul Mateen Rajput, Angus Roberts, Juliane Fluck, Martin Hofmann-Apitius, and Luca Toldo. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. *Journal of Biomedical Informatics*, 45(5):885 – 892, 2012. Text Mining and Natural Language Processing in Pharmacogenomics.

[6] Sam Henry, Kevin Buchan, Michele Filannino, Amber Stubbs, and Ozlem Uzuner. 2018 n2c2 shared task on adverse drug events and medication extraction in electronic health records. *Journal of the American Medical Informatics Association*, 27(1):3–12, October 2019.

[7] María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. The ddi corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of biomedical informatics*, 46(5):914–920, 2013.

[8] David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant. *Applied logistic regression*, volume 398. John Wiley & Sons, 2013.

[9] Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*, 2018.

[10] Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. Pretrained language models for biomedical and clinical tasks: Understanding and extending the state-of-the-art. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 146–157, Online, November 2020. Association for Computational Linguistics.

[11] Llew Mason, Jonathan Baxter, Peter Bartlett, and Marcus Frean. Boosting algorithms as gradient descent in function space. In *Proc. NIPS*, volume 12, pages 512–518, 1999.

[12] Irina Rish et al. An empirical study of the naive bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, volume 3, pages 41–46, 2001.

[13] Kshira Sagar Sahoo, Bata Krishna Tripathy, Kshirasagar Naik, Somula Ramasubbareddy, Balamurugan Balusamy, Manju Khari, and Daniel Burgos. An evolutionary svm model for ddos attack detection in software defined networks. *IEEE Access*, 8:132502–132513, 2020.

[14] Leslie N Smith. Cyclical learning rates for training neural networks. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 464–472. IEEE, 2017.

[15] Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300, 1999.

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008. Curran Associates, Inc., 2017.

[17] Jue Wang and Wei Lu. Two are better than one: Joint entity and relation extraction with table-sequence encoders. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1706–1721, 2020.

[18] Feng Xu, Zhenchun Pan, and Rui Xia. E-commerce product review sentiment classification based on a naïve bayes continuous learning framework. *Information Processing & Management*, 57(5):102221, 2020.