Discrete & Continuous Models
& Applied Computational Science
ISSN 2658-7149 (online), 2658-4670 (print)

2023, 31 (1) 64–74
http://journals.rudn.ru/miph

Research article

UDC 004.912
DOI: 10.22363/2658-4670-2023-31-1-64-74
EDN: VNWSXI

Methods of extracting biomedical information from patents and scientific publications (on the example of chemical compounds)

Nikolay A. Kolpakov[1],
Alexey I. Molodchenkov[2,3], **Anton V. Lukin**[2,3]

[1] *Moscow Institute of Physics and Technology (MIPT),*
9, Institutskiy Pereulok, Dolgoprudny, Moscow Region, 141700, Russian Federation
[2] *Federal research center "Computer science and control" of RAS,*
44-2, Vavilova St., Moscow, 119333, Russian Federation
[3] *Peoples' Friendship University of Russia (RUDN University),*
6, Miklukho-Maklaya St., Moscow, 117198, Russian Federation

(received: March 9, 2023; revised: March 23, 2023; accepted: April 10, 2023)

Abstract. This article proposes an algorithm for solving the problem of extracting information from biomedical patents and scientific publications. The introduced algorithm is based on machine learning methods. Experiments were carried out on patents from the USPTO database. Experiments have shown that the best extraction quality was achieved by a model based on BioBERT.

Key words and phrases: machine learning, natural language processing, named entity recognition, biomedical texts processing

1. Introduction

Every year the number of biomedical patents and scientific publications increases significantly. Often these texts don't contain any descriptive metadata, and this, in turn, leads to a large amount of unstructured data. Consequently, there is an increasing need for tools that could accurately extract the required information from such texts.

To extract infromation from texts for further processing, both machine learning approaches and algorithms based on regular expressions can be used. In [1, 2], regular expressions play a key role, and, on the contrary, in [3, 4], achievements in the field of deep machine learning, in particular the model of conditional random fields, are used. And in [5], a transformer-based machine learning technique is used, which, with proper parameter settings, can extract biomedical data quite well.

Although tools have been created for analyzing and interacting with unstructured data, these solutions are often based on rules that are applicable

to the specific data being processed. In this paper, we propose a solution to the problem of extracting biomedical information from patents using regular expressions. Thus, the resulting structured information can be used to train complex neural network models that will allow us to correctly extract information from a larger number of texts.

2. Related work

There aren't many solutions that solve the problem. Often, existing algorithms are designed to solve a large range of problems, so they they do not give sufficiently high results when solving the task of extracting definitions from biomedical patents and scientific publications.

For example, Jinhyuk Lee and his colleagues presented BioBERT (Bidirectional Encoder Representations from Transformers for Biomedical Text Mining) [5], a transformer language model [6] developed for automatic processing of the language of the biomedical field, which is pre-trained on large biomedical texts. This model can extract biomedical named entities, biomedical relationships in the text, and can also provide answers to biomedical questions. BioBERT is initialized with the values of weight functions that were obtained for BERT [7] (this model was previously trained on texts from the English Wiki and BooksCorpus), after which BioBERT was further trained on biomedical texts (this includes annotations from PubMed and full-text PMC articles).

The article [3] presents a different approach to solving NLP problems in the field of biomedicine. CLAMP (Clinical Language Annotation, Modeling, and Processing) uses both machine learning-based and rule-based methods to extract information. This toolkit allows you to extract named entities, split text into tokens, and much more. In their program, the authors use 3 types of tokenizers (to choose from):

1) OpenNLP tokenizer [8] based on machine learning;
2) tokenizer based on the separation of words by specified characters;
3) a rule-based tokenizer with various configuration parameters.

And for the task of extracting named entities, the authors suggest using:

1) conditional random fields algorithm (CRF) [9];
2) an algorithm based on a dictionary with a large amount of biomedical vocabulary collected from various resources, such as UMLS;
3) a regular expression-based algorithm for objects with common patterns.

OSCAR4 (Open-Source Chemistry Analysis Routines) [2] is an open system for automatic extraction of chemical terms from scientific articles. The basis of this work is the identification of chemicals based on regular expressions and identification based on a dictionary of predefined words. But to identify complex chemical compounds (which consist of several tokens), the Markov model of maximum entropy is used.

Also, there is a work [1] where the authors use morphology to extract biomedical words. The chemical object recognition system consists of two subsystems. The first extracts chemical objects and marks them in a normalized input document using a dictionary of predefined words and a morphological

approach. The morphology-based approach identifies the various elements in a chemical compound and combines them to create a final compound.

The second subsystem extracts additional chemical elements and distributes all recognized objects into classes of compounds and has such capabilities as decoding abbreviations and correcting spelling errors. In order to determine whether a certain entity is "chemical", the authors collected statistical information for each individual object. This information is used as the last stage of the extraction of named entities and is intended for the classification of the extracted object (either the object is chemical or not). These methods extract information from biomedical texts in general — they aren't aimed at extracting Markush structures [10] (see figure 1).

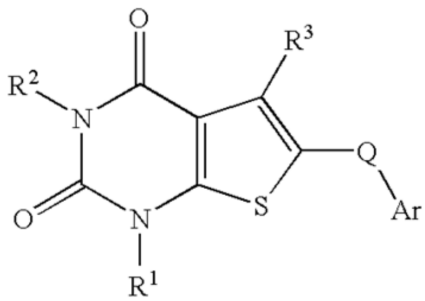

Figure 1. An example of a Markush structure, taken from US Patent 20040171623

3. The problem statement

Data concerning various biomedical patents are publicly available in various patent offices. Patents usually have a clear structure, which includes patent name, abstract, description, Claims and bibliographic information (date, patent number, authors).

The section we are interested in is Claims (see the figure 2), contains a description of the chemical compounds that are claimed by the authors of the patent. This is exactly the purpose of the legal protection provided by the patent. The Claims section may contain within itself several subsections that contain information on different chemical chains.

The connections presented in the Claims section can be described using the Markush structure [10] (see the figure 1). To find patents whose Markush structure is either the same or similar, you need to compare these structures. Since the Markush structure is a network model, then comparing such models directly is a very resource-intensive process. Therefore, so-called fingerprints are often used, which reflect the information presented in the Markush structures. But before that, you need to extract the information that is included in such structures, which is what this work is aimed at.

The task consists in extracting chemical compounds from the Claims section (see the table 1), names of variables (in place of which various values can be substituted), chemical elements, formulas and InChI codes [11] (see the figure 3) in order to transform this textual information into some structure of formal representation.

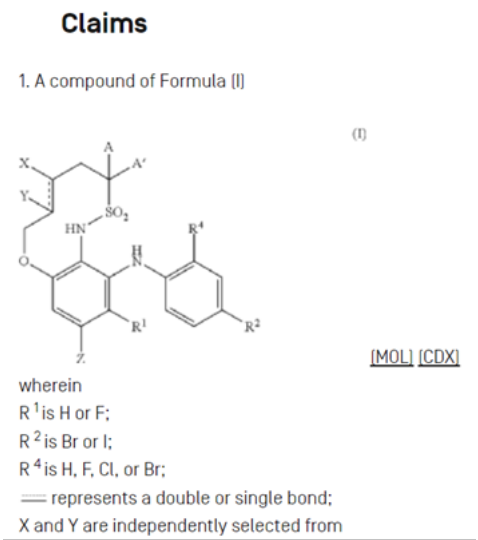

Figure 2. An example of the data in the Claims section, taken from US Patent 20120208859

Table 1

Examples of chemical compounds

Compound Name	Formula
nitrogen monoxide	NO
glucose	$C_6H_12O_6$
copper (II) sulfate	$CuSO_4$
carbon dioxide	CO_2
dichlorine heptoxide	Cl_2O_7

In the set-theoretic annotation, the problem can be formulated as follows: there are patents and scientific publications X where each element $x \in X$ is represented as $x = x_1, \dots, x_n$ ($x_1, \dots, x_n$ – is a sequence of words (tokens)), and a set of classes $Y = (y_1, \dots, y_5)$ is given, where:

— y_1 is the Claim number;
— y_2 is the variable to which we are looking for a description;
— y_3 is the description of the variable;
— y_4 is a link to another Claim;
— y_5 is in cases if the token doesn't match $y_1, \dots, y_4$.

It is necessary to construct a function F, that maps each element $x \in X$ to the corresponding element $y \in Y$.

The task of extracting information from texts is the search and classification of named entities (Named Entity Recognition) represented in unstructured text, according to predefined categories. A named entity is an n-gram in the text for which a category (class, label) is defined.

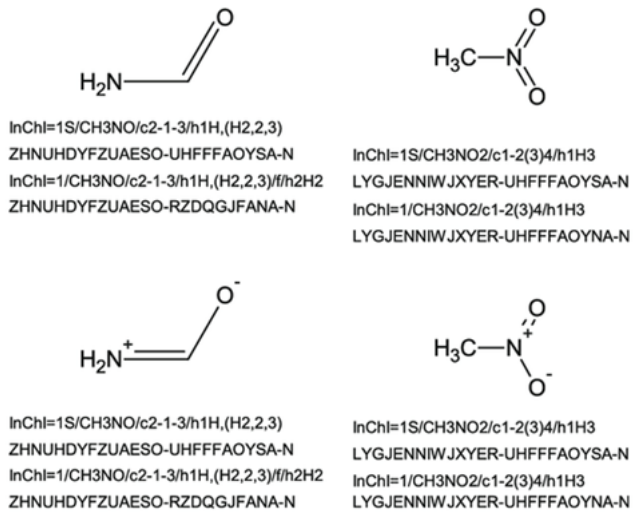

Figure 3. Examples of InChI codes [11]

4. Methodology

The algorithm for extracting information from texts can be divided into the following steps:

1) compilation of a dataset;
2) input data pre-processing;
3) data vectorization and feature extraction;
4) model training for extracting the necessary information from texts.

Compilation of the dataset also includes automated token markup. Data pre-processing includes normalization and tokenization of input data. The scheme of the algorithm is shown in figure 4.

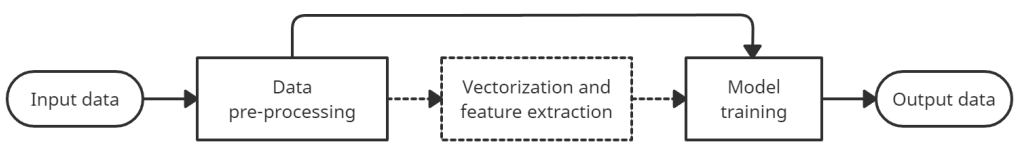

Figure 4. An algorithm for extracting information from texts

4.1. Compilation of a dataset

The data we worked with was taken from the USPTO patent database [12]. All data is initially presented in XML files that contain structured information about patents: description, annotation, bibliographic data, and Claims. To develop the algorithm, only the Claims section is taken from the files.

4.2. Input data pre-processing

The first stage of data processing is the extraction of the Claims section from the available dataset. Since such data has a similar design, the extraction is performed using regular expressions.

After extracting Claims, it is necessary to prepare the data for further work. To do this, the following string normalization is performed:

1. Extra spaces at the beginning and end of lines are removed.
2. Empty lines are also removed.
3. Each line is split in such a way that they contain only one description of variables. This is done by searching in each line of the following construction: *...variables ...definition-verb ...definitions ...definition-end-symbol*. At the same time, the situation is considered when a description of nested variables can be provided on the same line. For example, "Z is OR3, wherein R3 is C1-C6 alkyl". In this case, the string doesn't split.
4. If there is no *definition-end-symbol* in the string, then the strings are combined until the desired character is found.
5. If the line is the initial one for the Claim, but the Claim numbers are separated by a dash, then the subsequent content is copied for each Claim number from the given interval.

The resulting rows are then grouped by Claim. All the actions described above to normalize strings are also performed using regular expressions.

The use of normalization will allow us to train the model on a small sample more efficiently, and also will increase its accuracy. Grouping and splitting the rows will simplify the subsequent markup of the data.

The next step is to assign each token a label from the possible:

— CLAIM is Claim number;
— VAR is the variable that we are looking a description for; the description of this variable is substituted only to the last place where it was mentioned before the meeting of this variable;
— VAR-ALL is the variable that we are looking a description for; the description of this variable is substituted in all places where it was mentioned;
— DEF is description of the variable;
— REF is the link to another Claim;
— O is in case none of the above labels are assigned to the token.

The assignment of the corresponding label to tokens is carried out using the marking tools, provided by Federal research center "Computer science and control" of RAS. The result of these tools is data, containing the token, its label, the line number where it was found and the unique Claim number.

4.3. Vectorization and feature extraction

Since not all classification models accept string data values as input, it is necessary to vectorize such features. These include tokens and corresponding labels.

If a number can be associated with each unique label, then the situation is completely different with tokens. For each token, a vector of dimension 100 is constructed using the Word2Vec model [13, 14] to obtain vector representations of natural language words.

Word2Vec was trained on the collected dataset. The training took place for 10 epochs, with a sliding window size of 8.

To further train the machine learning model, it is necessary to combine tokens into lists based on Claims membership, and then submit these lists to Word2Vec as input. The result of such a model will be the mapping of each token to its vector representation.

Some machine learning algorithms, for example, based on Conditional Random Fields (CRF), are working better with features containing information about neighboring tokens relative to the one under consideration.

Therefore, another way of representation the data submitted to the input of such models is to match each token with a set of features. These features are:

— the token itself;
— the last 2–3 characters of the token;
— flag whether the token starts with a capital letter;
— flag whether the token is a number;
— flag whether the token contains only uppercase letters;
— information about neighboring tokens (a neighboring token and 3 flags, as in the previous points).

4.4. Model training

Both classical machine learning methods (Support Vector Machine [15], Conditional Random Fields [9]) and deep learning models (Stanford NER [16], BERT [7], BioBERT [5]), which are already pre-trained, were used as classification methods that would assign labels to previously unknown data based on marked and vectorized data.

Fine-tuning a pre-trained model, such as BioBERT, BERT and Stanford NER, was carried out on the data obtained in section 4.2 are tokens, labels, and Claims numbers. For BioBERT and BERT, in order to prevent overfitting, the number of epochs was chosen equal to 5 (see the figure 5).

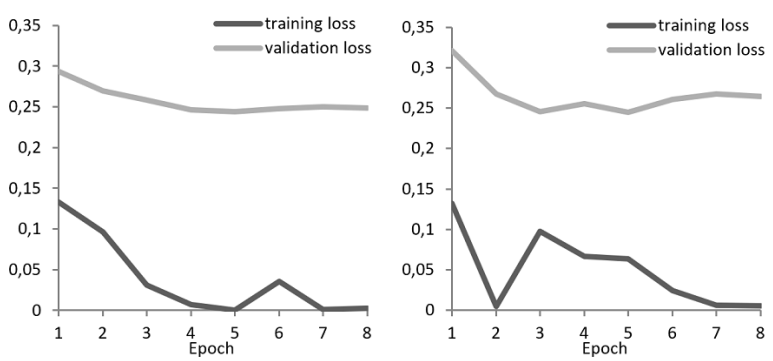

Figure 5. Epoch vs Loss graphs. Left graph is for BioBERT. Right graph is for BERT

The support vector machine (SVM) method was trained from scratch on vectorized data, and the conditional random field method (CRF) was trained on extracted features obtained in section 4.3.

5. Experiments

Within this work, a series of experiments was carried out to solve the classification problem. The experiments were conducted on 100 documents with more than 1,700 Claims. The training sample consisted of 70 documents, and the validation sample consisted of 30 documents.

Standard quality metrics were used to compare the results: precision, recall and F1-score [17]. Let's look at them in more detail.

To begin with, let's consider what TP, FP and FN are:

— TP is the number of tokens that the classifier has assigned the correct labels to;
— FP is the number of tokens that have the label O, but the classifier assigned them a different label;
— FN is the number of tokens that have a certain label (not O), but the classifier assigned them to another group.

Accuracy is the proportion of tokens that belong to a particular class, relative to all tokens that the classifier has assigned such a class label to. This metric is calculated:

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{1}$$

Recall is the proportion of tokens that the classifier has assigned a specific class label to, relative to all tokens that have this label. This metric is calculated by equation:

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{2}$$

F1-score is the average harmonic value of accuracy and recall. This metric is calculated by equation:

$$F1 - score = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2 * TP}{2 * TP + FP + FN}. \tag{3}$$

The classification results based on test data are shown in the table 2.

Table 2

Metrics values for various classification methods

Model Name	Precision	Recall	F1-score
SVM	0.5276	0.6340	0.5675
CRF	0.6701	0.6358	0.6378
Stanford NER	0.7530	0.8488	0.7981
BERT	0.8437	0.8978	0.8699
BioBERT	0.8467	0.9012	0.8731

From the performed experiments, it can be seen that classical machine learning methods show results much worse than pre-trained deep learning models, which, in turn, classify tokens at a fairly good level.

6. Conclusion

This article describes a method for solving the problem of extracting information from biomedical texts for its further processing. This method makes it possible to extract a description of chemical compounds that are claimed by the authors of patents. The machine learning models, such as SVM, CRF, Stanford NER, BERT and BioBERT, on which experiments were afterwards carried out, were trained. In the future, it is planned to convert the received data into the InChI code format and write fingerprints that correspond to the Markush structures claimed by the patent authors. It is also planned to conduct another series of experiments to improve the quality of information extraction from texts.

References

[1] S. A. Akhondi *et al.*, "Automatic identification of relevant chemical compounds from patents," *Database: the journal of biological databases and curation*, vol. 1, pp. 1–14, 2019. DOI: 10.1093/database/baz001.

[2] D. Jessop, S. Adams, E. Willighagen, L. Hawizy, and P. Murray-Rust, "OSCAR4: A flexible architecture for chemical textmining," *Journal of cheminformatics*, vol. 3, no. 1, pp. 1–12, 2011. DOI: 10.1186/1758-2946-3-41.

[3] E. Soysal *et al.*, "CLAMP — a toolkit for efficiently building customized clinical natural language processing pipelines," *Journal of the American Medical Informatics Association*, vol. 25, no. 3, pp. 331–336, 2017. DOI: 10.1093/jamia/ocx132.

[4] M. Swain and J. Cole, "ChemDataExtractor: a toolkit for automated extraction of chemical information from the scientific literature," *Journal of Chemical Information and Modeling*, vol. 56, no. 10, pp. 1894–1904, 2016. DOI: 10.17863/CAM.10935.

[5] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. So, and J. Kang, "BioBERT: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics (Oxford, England)*, vol. 36, no. 4, pp. 1234–1240, 2019. DOI: 10.1093/bioinformatics/btz682.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, pp. 5998–6008, 2017.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, pp. 4171–4186, 2018. DOI: 10.18653/v1/N19-1423.

[8] *The OpenNLP Project*, http://opennlp.apache.org, Accessed: 2023-03-07.

[9] *CRFsuite: a Fast Implementation of Conditional Random Fields (CRFs)*, http://www.chokkan.org/software/crfsuite/, Accessed: 2023-03-07.

[10] J. M. Bernard, "Handling of Markush Structures," *Journal of chemical information and computer sciences*, vol. 31, no. 1, pp. 64–68, 1991. DOI: 10.1021/ci00001a010.

[11] S. Heller, A. McNaught, I. Pletnev, S. Stein, and D. Tchekhovskoi, "The IUPAC International Chemical Identifier," *Journal of Cheminformatics*, vol. 7, pp. 1–34, 2015. DOI: 10.1186/s13321-015-0068-4.

[12] *USPTO*, https://www.uspto.gov/patents, Accessed: 2023-03-07.

[13] T. Mikolov, G. Corrado, K. Chen, and J. Dean, "Efficient estimation of word representations in vector space," *Proceedings of Workshop at ICLR*, pp. 1–12, 2013.

[14] T. Mikolov, W.-T. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations," *Proceedings of NAACL-HLT*, pp. 746–751, 2013.

[15] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 03, pp. 273–297, 1995. DOI: 10.1007/BF00994018.

[16] J. R. Finkel, T. Grenager, and C. Manning, "Incorporating non-local information into information extraction systems by Gibbs sampling," *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pp. 363–370, 2005. DOI: 10.3115/1219840.1219885.

[17] T. M. Mitchell, *Machine learning*. McGraw-Hill New York, 1997, 432 pp.

For citation:

N. A. Kolpakov, A. I. Molodchenkov, A. V. Lukin, Methods of extracting biomedical information from patents and scientific publications (on the example of chemical compounds), Discrete and Continuous Models and Applied Computational Science 31 (1) (2023) 64–74. DOI: 10.22363/2658-4670-2023-31-1-64-74.

Information about the authors:

Kolpakov, Nikolay A. — Master's degree student of Phystech School of Applied Mathematics and Informatics of Moscow Institute of Physics and Technology (e-mail: kolpakov.na@phystech.edu, ORCID: https://orcid.org/0000-0002-1640-1357)

Molodchenkov, Alexey I. — Candidate of Technical Sciences, Federal Research Center "Computer Science and Control" of RAS employee, employee of the Peoples' Friendship University of Russia (e-mail: aim@tesyan.ru, ORCID: https://orcid.org/0000-0003-0039-943X)

Lukin, Anton V. — Federal Research Center "Computer Science and Control" of RAS employee, employee of the Peoples' Friendship University of Russia (e-mail: antonvlukin@gmail.com, ORCID: https://orcid.org/0000-0003-4391-1958)

УДК 004.912
DOI: 10.22363/2658-4670-2023-31-1-64-74
EDN: VNWSXI

Методы извлечения биомедицинских текстов из патентов и научных публикаций (на примере химических соединений)

Н. А. Колпаков[1], **А. И. Молодченков**[2,3], **А. В. Лукин**[2,3]

[1] *Московский физико-технический институт, Институтский переулок, д. 9, Долгопрудный, Московская область, 141701, Россия*
[2] *Федеральный исследовательский центр «Информатика и управление» РАН, ул. Вавилова, д. 44, корп. 2, Москва, 119333, Россия*
[3] *Российский университет дружбы народов ул. Миклухо-Маклая, д. 6, Москва, 117198, Россия*

Аннотация. В данной статье предложен алгоритм для решения задачи извлечения информации из биомедицинских патентов и научных публикаций. Представленный алгоритм основан на методах машинного обучения. Авторами были проведены эксперименты на патентах из базы USPTO. Эксперименты показали, что лучшее качество извлечения продемонстрировала модель, построенная на основе BioBERT.

Ключевые слова: машинное обучение, обработка естественного языка, извлечение именованных сущностей, обработка биомедицинских текстов