Development of Methods for Extracting Information from Pharmacy Line Using Conditional Random Fields

Alexey I. Molodchenkov[1,2,3][0000−0003−0039−943X], Artem A. Nikolaev[1], and Evgenia A. Mitrokhina[2]

[1] Federal Research Center "Informatics and Control" of the Russian Academy of Sciences, Moscow

[2] Moscow Institute of Physics and Technology, Dolgoprudny

[3] Peoples' Friendship University of Russia, Moscow
mitrohina.ea@phystech.edu, aim@tesyan.ru

Abstract. The paper considers the solution to the problem of extracting information from short lines of pharmacological orientation in Russian language. As an example, pharmacy lines are used, from which you need to extract the full name of the drug, manufacturer, form of issue, dosage, number of pieces in a package and some other parameters. To extract this information, a conditional random field (CRF) algorithm was used. There was also created a method for preliminary standardization of the strings to bring string tokens to a single form. More than seven thousand pharmacy lines were marked for the experiments and 2 CRF models were trained - with and without preliminary standardization of the lines. For the model with standardization, the following results were obtained: accuracy for different data sets is 0.95 (on the validation set) and 0.89 (on the test set). For the model without standardization, the accuracy is 0.95 (on the validation set) and 0.87 (on the test set).

Keywords: Named Entity Recognition · Conditional Random Fields

1 Introduction

Extracting information from texts is relevant as it is used to solve a number of problems. The main goal of the tasks of extracting information from texts is to convert unstructured text data to some structured form (for example, a table or a semantic graph) for further processing of the received data.

Text analysis mainly consists of the following steps:

– vectorization of text;
– application of various methods (for example, machine learning) for their further processing, depending on the problem being solved.

Text vectorization is converting words to normal form and then converting them to vector form. For this, methods of tokenization, morphological analysis and vectorization are used. To convert words to an imperfect form, the libraries Mystem [1] for Russian, pymorphy2 [2] and nltk [3] for Russian and some other languages can be used. Methods and pre-trained models can be used to vectorize words and texts, such as: a bag of words [4], word2vec [5], doc2vec [6] and others.

At the next stage, depending on the task, regular expressions, rules, additional dictionaries, machine learning methods, etc. are applied.

In this paper, we consider the problem of extracting information from short pharmacy lines containing information about goods sold, for their further comparison with a predetermined reference book of medicinal products. Such solutions can be applied in various fields of activity and companies. For example, a marketing agency can use this information to assess the pharmaceutical market. Large companies with many warehouses and stores can use this kind of solution to automate the accounting of their products.

A feature of the texts used in this work is their small length and high density of entities that need to be recognized. For example, a pharmacy line contains information about the name of the drug, manufacturer, batch number, taste, if available, form of release, dosage, etc. Also, the texts contain many words that were not previously known (for example, new names of drugs or manufacturers), a minimum of grammar and many abbreviations. These features severely limit the application of the most commonly used approaches and algorithms.

2 Problem Statement

The task of extracting information from texts is a Named Entity Recognition task (NER). A named entity is an n-gram in text for which a class is defined. The task of recognizing named entities is to select continuous fragments of text and classify them.

At the entrance, a pharmacy line in Russian is given approximately of the following type: "АСКОРБИНОВАЯ К-ТА ГЛЕНВИТОЛ КЛУБНИКА №10 ТАБ.ЖЕВ. КРУТКА". It is necessary to first recognize the name of the drug, manufacturer, lot number and other parameters in this line, then link them to the reference name of the drug, manufacturer, lot number, etc. for further search for this string in the directory.

Let us list the problems that complicate the solution of this problem, which are to be solved:

- Abbreviations of some words ("к-та" instead of "кислота").
- Producers recorded in different languages ("биодерма лаборатория" and "BIODERMA LABORATORIES").
- Words that have multiple meanings depending on the context (the word «мед» as a taste or an abbreviation for the word «медицинский»).

3 An Overview of Named Entity Recognition Methods

Initially, the NER problem was solved without machine learning at all - using rule-based systems (for example, regular expressions). This solution stops working normally as soon as any ambiguities of the natural language come into play, but even in our task it can be used to determine the batch number, since a limited number of ways of recording it can be distinguished in the data. This solution gives us an f1-score of about 0.96 on one dataset and 0.93 on the other.

In [7], the authors investigated several different ways to recognize names, dates, locations, phone numbers and times from short messages in Swedish, including regular expressions. This method shows the best result for dates (0.72 F-measures), the worst - for locations (0.57 F-measures). The paper also shows that dictionaries and parts of speech significantly improve this result (the average F-measure increased from 0.65 to 0.84).

Progress in solving the NER problem has become the methods of classical supervised machine learning. In addition, entity dictionaries were actively used, which did not solve the ambiguity problem, but improved the quality. Among the algorithms that were actively used then were Support Vector Machine (SVM, [8]) and Conditional Random Fields (CRF, [9]), but also decision trees ([10]), hidden Markov models ([11]) and others. The disadvantage of these models is that feature selection is a completely empirical process, primarily based on linguistic intuition, and then a trial and error method; and the choice of features depends on the problem, which implies additional research for each new NLP problem. A more detailed overview of methods for solving the problem of recognizing named entities can be found in the source [12].

If we are talking about modern algorithms, then the problem of recognizing named entities is solved usually by neural network algorithms using Bi-LSTM + CRF (long short-term memory + conditional random fields [13]). Pre-trained embeddings are applied to the Bi-LSTM input, after several layers of Bi-LSTM and the output is a conditional random field (an undirected graph model, without which, as a rule, it is impossible to achieve state-of-the-art results). You can also add capitalization features, parts of speech, morphological features, etc. to the input to embeddings (Bi-LSTM + CRF + Char + Capitalization + POS).

In the article [14], the authors tested several variants of neural network architectures containing char and word Bi-LSTM, CRF, word embeddings, highway networks, etc. on three Russian-language datasets (Gareev's dataset, FactRuEval 2016, Persons-1000), and it was the Bi-LSTM + CRF + external word embeddings model that showed state-of-the-art results (F-measure 87.17, 99.26, 82.10, respectively).

Separately, I would like to mention that short texts differ significantly from long ones, and standard methods for recognizing named entities will work poorly for them. This is exactly what is shown in the article [15] - the quality has dropped from the usual 0.8 - 0.9 to 0.3 - 0.5 for tweets.

[16] demonstrates the results of using various existing systems for the task of recognizing named entities in tweets. Some Twitter-specific methods achieve F1 scores over 0.8, but are still far from the current results achieved with longer news texts. The authors say that the main reason for the deterioration in results is the poor use of capital letters (poor capitalization) - this feature is very important for the task of recognizing named entities. Also, abbreviations and slangs worsen

the quality of words that are not included in the dictionary, but their influence is no longer so significant.

4 Training CRF Model to Extract Entities from Pharmacy Strings

The training was carried out on 6000 marked lines, which were combined into a table. Each row of the table contains the pharmacy line itself, as well as all the parameters that need to be extracted from it. The piece of the data is in the table 1. The output is a trained CRF model capable of predicting an ordered sequence of classes corresponding to these tokens for any ordered sequence of tokens.

Table 1. Initial data format

Аптечная строка	Производитель	Наименование препарата	Форма выпуска	Дозировка	Объём	Кол-во штук в упаковке
САГЕНИТ ТАБ 100МГ Х 30	НИЖФАРМ - РОССИЯ	САГЕНИТ	ТАБ	100МГ	NaN	Х 30
ЭХИНАЦЕЯ 1,5Г №20	ХОРСТ КОМПАНИЯ (АЛТАЙ)	ЭХИНАЦЕЯ	NaN	NaN	1,5Г	№20
КОМПЛИВИТ КАЛЬЦИЙ Д-3 ФОРТЕ ТАБЛ ЖЕВ №100 МЯТНЫЕ	ФАРМСТАНДАРТ-УФИМСКИЙ ВИТАМИННЫЙ 3-Д ОАО	КОМПЛИВИТ КАЛЬЦИЙ Д-3 ФОРТЕ	ТАБЛ ЖЕВ МЯТНЫЕ	NaN	NaN	№100
ГРУДНОЙ СБОР №1 50Г	ЛЕК С+	ГРУДНОЙ СБОР №1	NaN	NaN	50Г	NaN
ТЕТРАЦИКЛИН ТАБ. П/ПЛЕН. ОБ. 100МГ №20(БЛИСТЕР)	БИОСИНТЕЗ	ТЕТРАЦИКЛИН	ТАБ. П/ПЛЕН. ОБ.	100МГ	NaN	№20

The learning algorithm consists of the following steps:

- String standardization
- Converting strings to the format required for using CRF
- Extraction of features from words
- Train the CRF Model to Predict the Class for a Word Based on Extracted Features

Let's consider the presented steps of the algorithm in more detail.

4.1 String Standardization

By standardizing a string in this task, we mean bringing the string tokens to a single form. The method that standardizes strings does the following conversions:

- Removes extra characters (quotes, brackets, commas)
- Brings tokens in cyrillic to a single form, uses a dictionary of substitutions for this. At this step, the most frequent errors in the spelling of tokens are "corrected", the ending is brought to a pre-selected form and abbreviations are replaced with full words
- In fractions, replaces a comma with a dot
- Removes extra spaces and add spaces where needed.

Example string before standardization
'ВАКСИГРИП СУСП.В/М И П/К 0,5МЛ/ДОЗА ШПР. №1'
and after it 'ВАКСИГРИП СУСПЕНЗИИ ВНУТРИМЫШЕЧНОГО ВВЕДЕНИЯ И ПОДКОЖНОГО 0.5 МЛ ДОЗА ШПР №1'
The application of standardization in this task has several goals:

- This approach allows you to improve the accuracy of the model and learn better on a small sample (or a smaller sample to achieve similar quality, if we consider an approach with and without standardization).
- Since we isolate and classify tokens to further search for the closest drug or product in a directory consisting of all possible options, the second goal of standardization is to use ordinary equality instead of using metrics to compare the proximity of tokens. This allows you to use filtering by those fields that are unambiguously standardized in our country.

4.2 Converting a String to the Form Required to Use CRF

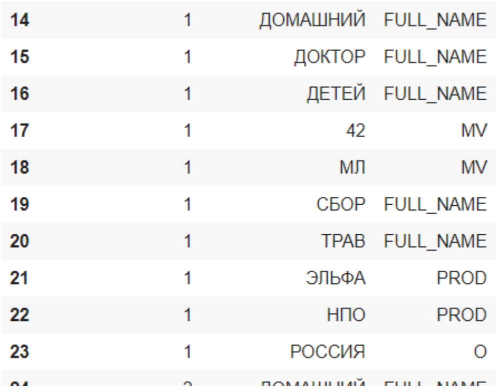
Initially, the data is a table of almost 6,000 labeled rows. Each row of the table contains the pharmacy row itself, as well as all the parameters that need to be extracted from it (see Table 1).

To train the CRF model, it is necessary to present the data in the form of a table, each row of which contains one token, the number of the pharmacy line from which this token was taken, as well as the class corresponding to this token (see Fig. 1).

Description of possible classes:

- FORM_QN - number of pieces in a package
- FULL_NAME - full name of the drug
- MV - volume
- NM_D - dosage
- NM_F - form of issue
- PROD - manufacturer
- O - does not belong to any of the above classes

Not all the parameters listed here are required to appear in every line.



14	1	ДОМАШНИЙ	FULL_NAME
15	1	ДОКТОР	FULL_NAME
16	1	ДЕТЕЙ	FULL_NAME
17	1	42	MV
18	1	МЛ	MV
19	1	СБОР	FULL_NAME
20	1	ТРАВ	FULL_NAME
21	1	ЭЛЬФА	PROD
22	1	НПО	PROD
23	1	РОССИЯ	O
24	2	ДОМАШНИЙ	FULL_NAME

Fig. 1. Data in the format required to use the CRF

4.3 Features that Were Used to Train the Model

As features of the word were used: the word itself in lower case, the last 2 characters of this word, the length of the word and a flag about whether this token is a number or not. And also the same features for two neighboring tokens.

4.4 Teaching the CRF Model to Predict the Class for a Word

To train the model and conduct experiments, the entire data set was divided into training and test samples (the size of the test sample is 20% of the entire data set).

The CRF (Conditional Random Fields) method was chosen as a classification method, because it allows you to independently form a set of features by which you can vectorize words and texts and is popular for the NER problem, as it is intended for marking sequences. Using word embedding and other standard vectorization methods is not suitable for this task. New drugs appear, all words are specific, and the existing methods and pre-trained models were trained in a common vocabulary.

A random field is a multidimensional random variable V, where each component is a one-dimensional random variable. For convenience, we will assume that $\forall i \quad V_i$ are discrete and the set of their values is finite. We denote the implementation of a multidimensional random variable V as $v \in \Omega$, where Ω is the set of all possible configurations. A random field can be represented as a graph, in which the vertices are the components of the multidimensional random variable V, the edges are the dependencies between them. A random field is called Markov if 2 Markovian conditions are satisfied:

1. $\forall v \in \Omega \quad P(V = v) > 0$
2. $P(V_i = v_i | V_j = v_j, j \in A \backslash \{i\}) = P(V_i = v_i | V_j = v_j, j \in \delta i)$

where δi - set of neighbors of the vertex V_i.

A conditional random field is a Markov random field, in which the set of random variables is divided into 2 disjoint subsets - X and Y - the set of observable and hidden variables. The prediction task is to optimally reconstruct the values of y, provided that we know the observables x. That is, the optimization task is to maximize the conditional probability p (y | x): $y^* = argmax_y p(y|x)$. Calculation of the model p * (y | x) is solved as an optimization problem with given constraints (the difference between the observation and its estimate must be minimal and the condition $\sum_x p(y|x) = 1$ for all x). According to the Hammersley-Clifford theorem (which connects Markov random fields and the Gibbs distribution), we need to maximize

$$p(y|x) = \frac{\prod\limits_{c \in C(G)} \psi_c(x,y)}{\sum\limits_{y' \in y} \prod\limits_{c \in C(G)} \psi_c(x,y')},$$

where the factor functions ψ_c are usually the exponent of a linear combination of functions from features with weights that need to be determined during training $\psi_c = exp(\sum\limits_{k=1}^{K} f_k(x_c, y_c)\theta_k)$. This method belongs to the probabilistic methods of classical machine learning. Its implementation has good speed, which is very important when processing large amounts of information.

More details about the CRF method can be found in [9].

5 Experimental Research

For the experiments, 2 samples were used. The first sample contains 6,000 pharmacy lines and is randomly divided into training and validation at a ratio of 80%/20%. The second sample is an additional 1000 lines taken from another dataset, which contains a significant proportion of the unknown drug for the model, since they were absent in the training sample. This sample was used for the test.

The two resulting models (with and without string standardization) were tested on validation and test datasets. In the tables 2, 3, 4 and 5, you can see the results of the experiments.

Vectorization of tokens by n-grams and further comparison of vectors using cosine distance were used as a baseline. The resulting average accuracy for further comparison was 0.65.

The first thing you may notice is better quality of both models compared to the baseline.

The model shows the worst results on the test data (especially for MV and NM_D). This can be explained by the fact that the data in the test set contain a large number of completely new drugs for the model and have some differences from the data on which the training and validation was carried out. For example, dosages and volumes without specifying units of measurement are more common in the test set.

You can also notice that on the validation set string standardization does not improve the prediction quality, but on the test set, there are noticeable improvements for volume, dosage and form of release - the classes on which the standardization method has the most significant influence. The difference with validation can be explained by the fact that the data in the test set have more typos and abbreviations that need to be corrected through standardization, so the consequences of standardization are more noticeable.

In all experiments the model predicts full name of the drug NM_FULL best of all, the worst predictable classes are dosage NM_D and volume MV. Difficulties with dosage and volume may occur because they are too similar and easy to confuse.

Table 2. No preprocessing of lines on validation set

	Precision	Recall	F1	support
FORM_QN	0.97	0.98	0.98	545
FULL_NAME	0.95	0.97	0.96	4112
MV	0.97	0.98	0.97	832
NM_D	0.91	0.89	0.90	469
NM_F	0.95	0.93	0.94	1047
PROD	0.94	0.97	0.95	2283
O	0.95	0.88	0.91	2280
accuracy			0.95	11568
macro avg	0.95	0.94	0.94	11568
weighted avg	0.95	0.95	0.95	11568

Table 3. No preprocessing of lines on test set

	Precision	Recall	F1	support
FORM_QN	0.97	0.97	0.97	981
FULL_NAME	0.86	0.89	0.87	2042
MV	0.58	0.90	0.71	140
NM_D	0.70	0.48	0.57	152
NM_F	0.86	0.83	0.84	1228
PROD	0.87	0.93	0.90	1958
O	0.86	0.77	0.82	1852
accuracy			0.87	8353
macro avg	0.82	0.82	0.81	8353
weighted avg	0.87	0.87	0.87	8353

Table 4. With line preprocessing on validation set

	Precision	Recall	F1	support
FORM_QN	0.96	0.98	0.97	534
FULL_NAME	0.95	0.97	0.96	4445
MV	0.94	0.98	0.96	1419
NM_D	0.95	0.90	0.92	939
NM_F	0.97	0.95	0.96	1377
PROD	0.95	0.98	0.96	2603
O	0.95	0.85	0.90	1555
accuracy			0.95	12872
macro avg	0.95	0.94	0.95	12872
weighted avg	0.95	0.95	0.95	12872

Table 5. With line preprocessing on test set

	Precision	Recall	F1	support
FORM_QN	0.97	0.95	0.96	855
FULL_NAME	0.88	0.91	0.89	2186
MV	0.66	0.83	0.74	285
NM_D	0.75	0.59	0.66	272
NM_F	0.95	0.90	0.92	1811
PROD	0.88	0.95	0.91	2218
O	0.88	0.78	0.82	1299
accuracy			0.89	8926
macro avg	0.85	0.84	0.84	8926
weighted avg	0.89	0.89	0.89	8926

6 Conclusion

Using the CRF method, it was possible to obtain a model showing good results in the recognition of named entities in short texts of pharmacological topics.

Accuracy for the validation data is 0.95, for the test data it is 0.89. Deterioration of results can be explained by the emergence of new drugs that are absent in the training sample, and by some differences in the data structure - for example, the frequent absence of units of measure for volume and dosages. In the future, it is planned to improve the quality by using combinations of different approaches to build a model for the classification of words and by expanding the set of features for vectorization of tokens.

References

1. Mystem. https://yandex.ru/dev/mystem/.
2. Pymorphy2. https://pymorphy2.readthedocs.io/en/stable/.
3. Natural language toolkit. https://www.nltk.org/.
4. Harris Zellig. Distributional structure. *Word*, 10:146–162, 1954.
5. Greg Corrado Tomas Mikolov, Kai Chen and Jeffrey Dean. Efficient estimation of word representations in vector space. *ICLR Workshop Papers*, 2013.
6. Baldwin T. Lau J. H. An empirical evaluation of doc2vec with practical insights into document embedding generation. *arXiv preprint arXiv:1607.0536*, 2016.
7. Tobias Ek, Camilla Kirkegaard, Håkan Jonsson, and Pierre Nugues. Named entity recognition for short text messages. *Procedia - Social and Behavioral Sciences*, 27:178–187, 2011. Computational Linguistics and Related Fields.
8. William S Noble. What is a support vector machine? *Nature Biotechnology*, 2006.
9. Bengong Yu and Zhaodi Fan. A comprehensive review of conditional random fields: variants, hybrids and applications. *Artificial Intelligence Review*, 2020.
10. S. B. Kotsiantis. Decision trees: a recent overview. *Artificial Intelligence Review*, 2013.
11. L. Rabiner and B. Juang. An introduction to hidden markov models. *IEEE ASSP Magazine*, 3(1):4–16, 1986.
12. David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification, Jan 2007.
13. Changki LEE. Lstm-crf models for named entity recognition. *IEICE Transactions on Information and Systems*, E100.D(4):882–887, 2017.
14. The Anh Le, Mikhail Arkhipov, and Mikhail Burtsev. Application of a hybrid bi-lstm-crf model to the task of russian named entity recognition. pages 91–103, 09 2018.
15. Alan Ritter, Sam Clark, Oren Etzioni, et al. Named entity recognition in tweets: an experimental study. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pages 1524–1534, 2011.
16. Leon Derczynski, Diana Maynard, Giuseppe Rizzo, Marieke van Erp, Genevieve Gorrell, Raphaël Troncy, Johann Petrak, and Kalina Bontcheva. Analysis of named entity recognition and linking for tweets. *Information Processing and Management*, 51(2):32–49, 2015.