ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

MODERN STATISTICAL AND LINGUISTIC APPROACHES TO PROCESSING TEXTS IN NATURAL LANGUAGES

ALEKSANDR EVGENJEVICH PETROV, DMITRII ALEKSANDROVICH SYTNIK

Complex Systems LLC
Russia, 170021, Tver, Skvortsova-Stepanova Street, 83

ABSTRACT

Natural language processing (NLP) is a research area that focuses on studying the methods of computer analysis and synthesis of natural languages. The sources of information can include not only texts, but also audio and video data. In this article, we will focus on text mining. The analysis is divided into the following subtasks: information extraction, tonality analysis, question-answer systems, etc. In turn, information extraction also includes subtasks: named entity recognition (NER), relation extraction, extraction of keywords and word combinations (collocations). The methods of NLP are divided into linguistic (based on rules and grammars) and probabilistic; there are also hybrid methods that combine both approaches. The aim of this paper is to provide an overview of modern approaches to text processing using the example of the tasks of named entities recognition and identifying the relationships between them.

Keywords: *NLP, Information Extraction, Named Entity Recognition, NER, Relation Extraction, Text Mining, Statistical Method, Linguistic Method, Machine Learning, Supervised Learning, Semi-Supervised Learning.*

1. INTRODUCTION

Text mining is a relevant, challenging and interesting applied task. Among the most relevant applied tasks, there are searching for relevant documents on the Internet on a user's request, automatic classification of texts, automatic abstracting, and searching for events in the text.

The purpose of research is to review modern NLP methods that allow solving the problem of recognition of named entities and relations between them.

The problems solved in this study are as follows: the analysis of contemporary literature, the study of NER methods and relations between them, formation of recommendations for their use with due regard to the constraints on the input data of each method, classification of methods on the mathematical apparatus used.

Named entity recognition is usually a preliminary stage of text mining. This task was first formulated at the MUC-6 Conference [1].

The aim of the task is to determine to which class an expression in the text belongs. For example, the expression "Vladimir Putin" belongs to the class PERSON (names). The existing approaches are divided into rule-based methods, such as FASTUS [2], machine learning methods (high results were shown by the work [3]) and hybrid methods (works [4-5]), using the advantages of both approaches.

The extraction of relations in the text is the task of identifying the relationship between named entities. Usually, binary relations are considered, such as *located_in (city, country)*, but there are also methods that identify the relations between several entities and find application in biomedicine. The task of relation extraction is not trivial. Some approaches search for relations only inside the sentence, while other approaches take into account the context of the document [6].

In this paper, we will consider the basic methods of natural language text processing, using the example of NER and relation extraction tasks.

Named entity recognition

One of the subtasks of information extraction is named entity recognition.

It was first formulated in 1996 at the Message Understanding Conference (MUC-6) as the task of identifying in the text expressions that refer to people, geographical locations, organizations, etc. More formally, it is the task of identifying text elements (words and sequences of words) and their classification. At this conference, the main classes were also proposed:

- EXANAME: people, organizations, geographic places
- NUMEX: monetary units
- TIMEX: date, time

The classes were supplemented and expanded. In [7], a set consisting of more than 200 classes is described. Let us consider some examples of entity extraction in a sentence: "*During his visit to [LOCATION Beijing] [PERSON Vladimir Putin] claimed the priority in the relationships with [REGION China]*."

Here, LOCATION, PERSON, REGION are the classes, the meaning of which is obvious. In the following example, the ambiguity in the definition of the entity class of the word Tesla arises: "*[B-ORG Tesla] [L-ORG Motors] is named after the electrical engineer and physicist [B-PERSON Nikola] [L-PERSON Tesla]*."

At the beginning of the sentence, it is part of the name of the company Tesla Motors, while at the end of the sentence, it is part of the name of the physicist Nikola Tesla. For correct class identification, the surrounding context of these entities should be taken into account.

The following features of NER should also be noted:

- language of the studied texts;
- genre and subject area.

The knowledge about the language simplifies the task. For example, in Russian proper names begin with a capital letter, while in Arabic this is not the case. Therefore, a method developed for one language may not work properly for another language.

Designing the systems that are resistant to the domain change is a challenging task [8]. A change in the genre and subject area results in a deterioration of the system performance, especially for systems based on manually created rules [9].

The evaluation of the performance of NER systems is carried out on texts marked up manually by experts. At the conference CoNLL'03 (Conference on Computational Natural Language Learning), the following criterion was proposed: a named entity is considered to be recognized correctly if the type and boundaries of this entity determined by the method coincide with the type and boundaries determined by the experts. As indicators of the system performance, the indicators of precision (P), recall (R) and F-score (F_1 score) can be considered:

$$\mathbf{P} = \frac{\text{number of correctly extracted entities}}{\text{total number of extracted entities}}, \quad (1)$$

$$\mathbf{R} = \frac{\text{number of correctly extracted entities}}{\text{number of entities in the corpus}}, \quad (2)$$

$$\mathbf{F} = \frac{2PR}{P+R}, \quad (3)$$

Example

Let us consider the sentence: "*The CEO of [ORG Microsoft company] [PER Steve Ballmer] today announced the release of [PROD Windows 7]*." In this sentence, there are three entities. This entities are extracted by the expert. Let our algorithm extract the entities as follows: "*The CEO of [ORG Microsoft company] [PER Steve] Ballmer today announced the release of Windows 7.*"

Let us calculate the values of P, R, and F.

P = 1([ORG Microsoft company]) / 2("[ORG Microsoft company],[PERSON Steve]) = 0.5,

R = 1/3,

F = 2 · 0.5 · 0.33 / (0.5 + 0.33) ~ 0.4

There are also other methods of quality evaluation, sensitive to errors in determining the classes of entities, for example, the methods described in [10].

2. METHODS

The main methods used in solving NER tasks are linguistic (based on rules and grammars) and statistical methods. One of the first works in this field was the work of Lisa Rau [11], in which she used a set of manually created rules and heuristics for identifying company names in a text.

Linguistic methods also include methods based on using regular expressions or sets of regular expressions. In [2], the system FASTUS is proposed, based on a set of finite state machines. It uses several machines, each of which processes a particular stage. This system showed a good result with P = 96% and R = 92% [12]. It is worth noting that systems based on finite state machines are inherently limited because natural languages cannot be described using regular expressions, as shown by Chomsky [13]. However, the advantage of such systems is the relative ease of implementation and good results for some tasks, such as extracting names of companies in the news stream.

The most popular methods for solving the NER task are machine learning methods. These methods include: support vector machine, hidden Markov chains, maximum entropy method, conditional random fields. Systems of this type are first trained on a certain text, marked up by named entities. It should be noted that the use of a training text is the bottleneck, because the relevance of this text is lost in the course of time: for example, facts which were extracted in 1990, are unlikely to be useful in 2016. Some modern studies focus on automatic acquisition of training data. For example, [14] demonstrates receiving a training corpus based on the Wikipedia data.

Support vector machine

A support vector machine (Figure 1) is generally not a statistical method, because it returns the margin between the vector and the separating hyperplane [15], where H_2 is a hyperplane with the largest possible margin. In [16], a method of mapping margins into probabilities was proposed.

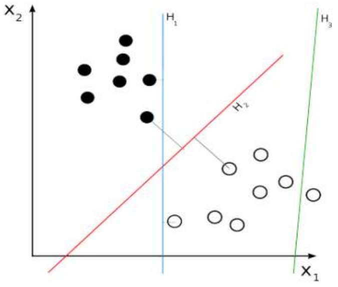

Figure 1. Support Vector Machine

The vectors represent the sets of word features. In general, the choice of features is an important factor affecting the method performance. Features can be conventionally divided into several groups:

- Word-level features include the word itself, n-grams, prefixes, suffixes of words, parts of speech, etc.
- Document-level features encode the information on the relationship of the word with the entire text, where we identify the entity (in the header, text of the article, etc.).
- Features of external sources: lists of named entities, stop words, words, etc.

The classic support vector machine is the binary classification method. In a simplified form, classifier training boils down to finding the hyperplane with the largest possible margin, which separates the training set of vectors. Afterwards, this hyperplane is used to classify the test vectors. Those vectors that lie on one side of the hyperplane, are assigned to one class, others – to the second class.

There are methods [17] that make it possible to generalize the classification by the number of classes greater than 2; this type of classification is called multiclass [18].

3. RESULTS

In [19], 15 features are considered (features of words, features of context, etc.). A binary feature vector is built. If the word has the i-th feature, then x [i] = 1; otherwise, x [i] = 0, where x is a feature vector. The value of the F-score is 90.03%.

In [20], the word-level features are also used. This work applies the probabilistic approach based on Platt's idea [16]. The approach, described in the article, is based on building a lattice (lattice-based approach). Each sentence is processed separately. For each word in the sentence, a column in the lattice is created. Each column contains one vertex for each possible tag. Each vertex "x" in one column is connected by an edge with each vertex "y" in the next column, if "y" is allowed to follow "x". For example, an edge from the I-LOC class is forbidden in B-PER. The task is to assign certain probabilities to edges and to find the path with the highest possible probability in this lattice. Testing and training were carried out on the CoNLL-2003 data for texts in English and German. The values of the F-score are presented in Tables 1 and 2.

Table 1. The Values Of The F-Score From The Article [20] For The English Text On The Conll-2003 Data.

English	P	R	F
LOC	88.22 %	89.33 %	88.77 %
MISC	74.89 %	73.50 %	74.19 %
ORG	79.31 %	78.69 %	79.00 %
PER	89.71 %	91.65 %	90.67 %
Total	84.45 %	84.90 %	84.67 %

Table 2. The Values Of The F-Score From The Article [20] For The German Text On The Conll-2003 Data.

German	P	R	F
LOC	75.08 %	72.17 %	73.60 %
MISC	63.62 %	42.54 %	50.98 %
ORG	69.20 %	58.99 %	63.69 %
PER	86.53 %	74.73 %	80.20 %
Total	75.97 %	64.82 %	69.96 %

From Tables 1 and 2, it can be seen that the results are rather high. The baseline system proposed for CoNLL-2003 gives the values of the F-score at the level of 59.61 for the English text and 30.30 for the German text.

Hidden Markov chains

A Markov chain is a sequence of random events, in which the probability of each event depends only on the current state of the process and does not depend on its earlier states. An event is understood as the transition from one state into another. A hidden Markov model (HMM) is characterized by the presence of hidden (not observed) states. Hidden states are classes of recognized entities.

4. RESULTS

In [21], a HMM is used to classify the entities by the class of names (NAME). As features, the word-level features were used, such as: all uppercase characters, all lower case characters, the first word in the sentence, word that contains numbers and letters, etc. The test set for the English language is the MUC-6 set. The values of the F-score are presented in Table 3.

Table 3. The Values Of The F-Score For The English Text On The MUC-6 Data.

Character case	F
Mixed	93%
Upper	91%

In [22], 4 sets of features are used: word-level features (f1), semantic features (f2), the dictionary of named entities (f3): dates (DATE), organizations (ORG), people (PERSON), geographical locations (LOC), and other, external features (f4) that represent the list of already recognized named entities for searching the aliases of the words. Test data: MUC-6 and MUC-7 datasets for English. The values of the F-score are presented in Table 4.

Table 4. The Values Of The F-Score From The Article [22] For The English Text On The MUC-6 And MUC-7 Data Depending On The Combinations Of Groups Of Features.

Composition	F	P	R
$f = f^1$	77.6%	81.0%	74.1%
$f = f^1, f^2$	87.4%	88.6%	86.1%
$f = f^1, f^2, f^3$	89.3%	90.5%	88.2%
$f = f^1, f^2, f^4$	92.9%	92.6%	93.1%
$f = f^1, f^2, f^3, f^4$	94.1%	93.7%	94.5%

The above-mentioned article also identifies the dependence of the results on the test sample size: 200 KB of training data give the F-score of 90%, while the decrease to 100 KB results in a significant deterioration of results.

Maximum entropy method

Speaking informally, entropy is a measure of information uncertainty. The entropy of a discrete random variable X with possible values $\{x_1, ..., x_n\}$ and the probability function P(X) is calculated by the formula [23]:

$$H(X) = -\sum_{i=1}^{n} P(x_i) \cdot log_b P(x_i), \quad (4)$$

where b is usually taken to be equal to 2.

Let us consider a Bernoulli distribution [24], describing the process of coin tossing. The entropy of this distribution is defined by the formula:

$$\mathrm{H(X)} = -\mathrm{q} \cdot \log_2(\mathrm{q}) - \mathrm{p} \cdot \log_2(\mathrm{p}). (5)$$

From the formula (5), it can be seen that if the probability P(heads) of getting heads is equal to 1 and P(tails) = 0, the entropy is equal to 0, i.e., there is no uncertainty in coin tossing – we always get heads.

The maximum value H(X) is achieved when $p = q = 0.5$, i.e. getting heads and tails is equiprobable.

According to the maximum entropy principle, the most characteristic distribution is the one which maximizes the entropy.

This approach was used, for example, in the article [25]. The maximum entropy classifier was used to classify each word in the following way: the beginning of the named entity (B tag), a word inside the named entity (C tag), a word at the end of the named entity (L tag) or a unique word (U tag). During testing, the classifier can generate inadmissible sequences of classes (for example, the LOC-L class follows the PER-B class). Therefore, the probability of the transition between classes $P(c_i|c_{i-1})$, is determined, which is equal to 1 if the sequence is allowable; otherwise, it is equal to zero. Thus, the probability of assigning the classes c_1, ..., c_n to the words in the sentence s in the document D is determined by the formula

$$\mathrm{P}(c_1, ..., c_n|s, D) = \prod_{i=1}^{n} \mathrm{P}(c_i|s, D) * \mathrm{P}(c_i|c_{i-1}), \quad (6)$$

where $P(c_i|s, D)$ is determined by the maximum entropy classifier.

As word features, local and global features are used. Local features take into account neighboring words in the considered sentence. Global features account for the occurrence of the word in the entire document. In addition, automatically extracted lists, received at the preprocessing stage, can be used. These lists include some unigrams, bigrams, suffixes of words, etc. For more details, one can refer to the article itself [25].

5. RESULTS

As a dataset, the English and German corpus was used. The value of the F-score for the English text is from 85% to 93% (for different classes of entities). Using global features gives an increase in the F-score by approximately 2%. In general, the approach described in the paper can also be used for other languages.

Method of conditional random fields

The method of conditional random fields (CRF) is a graph model, which is used to represent the joint distributions of a set of several random variables. The CRF method deals with the conditional distribution (y|x) of the sequence of labels y, where x is the vector of observed elements (in this case tokens).

This method is used, for example, in [3] and in [26] for extracting temporal expressions in Russian.

6. RESULTS

In [3], the values of the F-score are from 86.72% to 92.28%. In [26], the value of the F-score is: 93.05%

Hybrid methods

There are also approaches that use hybrid methods [4-5]. In [4], the system of TEG (trainable extraction grammar) is described – a system of trainable grammar for entity extraction. In fact, it is a context-free grammar with the added function of the probability P(r) of the application of the rule r at the output phase. A set of rules, which initially have an equal probability of application, is created, and then, in the process of training, the probability distribution of rules is identified. In the course of parsing, the probabilities of rule application are used, and the parse tree with the maximum probability is selected. Results: the paper [4] contains the comparison of the HMM (Hidden Markov Model) classifier, DIAL Rules (a system based only on rules) and the TEG. The DIAL system shows a high precision at the level of 93% for the class PERSON, but the lowest recall – 81.32%. The TAG system shows the accuracy of 90.78% (above HMM, but below DIAL) and the highest recall; thus, the value of the F-score reaches 92.24%.

In [5], the MERGE system (Maximum Entropy Rule Guided Extractor) is presented, which uses the Maximum Entropy Markov Model (MEMM). For each token, a feature vector is built in a special way according to a certain set of rules. The results, as well as those of TEG, show the superiority of the hybrid method over the methods of the manual creation of rules and methods of machine learning. It is enough to create a small amount of rules. The average value of the F-score is 94.2% for the MUC-7 data.

It should be noted that it is not quite correct to compare the results of the operation of the above-described methods, as they used different training and test sets. However, it can be said that they have a rather high precision and recall and give the value of the F-score from 75% to 94.5%.

NER and Wikipedia

Recently, the possibility of using structured information sources, such as Wikipedia, for extracting named entities has been actively explored.

Wikipedia articles often describe a specific object and begin with abstracts, in which the entire article is briefly described. It is possible to select the first sentence of the first paragraph and find out that in many cases it is the definition.

Additional information is also used, such as: information sheets, links to other articles, translations of this article, specifying the article category.

In [27], the English version of Wikipedia was used as a training corpus. A method of classifying articles by the entity categories was proposed. The value of the F-score: 0.92. The authors also expanded the training corpus by using the corpora MUC-7, CoNLL-2003, BNN, which eventually led to a deterioration of the method performance.

In [28], it is noted that the defining noun (which stands after the verb *to be*) can be a good indicator of the entity class. The authors use such a noun as a feature for the training of the classifier based on the model of conditional random fields. On the CoNLL-2003 data, the method showed the value of the F-score equal to 86.6%.

Some studies deal with the generation of named entities on the basis of their categories [29]. There is also a separate research area, which includes works dedicated to the classification of entities in Wikipedia itself, for example, [30-34].

Thus, by using Wikipedia, it is possible to automatically generate dictionaries of named entities. These dictionaries can be used at different stages of NER as additional sources of information for classifiers.

The result of named entity extraction can be used directly, for example, to form an initial notion of the document content. As an example, one can consider e-mail processing – it is possible to identify in a letter the named entities PRODUCTS, then find which companies offer such products, and show the user the advertisements of these companies related to this particular product.

However, the NER task is a subtask of a more general task of relation extraction. Its essence is to find relationships between entities.

Methods of relation extraction

The relationship between entities is determined in the form of a tuple: t = (e1, e2, ..., en), where ei are entities in the predetermined relation r inside the document [35]. Most relation extraction systems extract binary relations. Examples of such relations may include: located_in (Red Square, Moscow), buys (Microsoft, Skype) – means that Microsoft bought Skype.

There are works devoted to the study of higher-order relations, which find application, in particular, in biomedicine [36], but we will discuss only the basic approaches for binary relations.

The methods are divided into supervised methods and semi-supervised methods. Supervised methods represent a classification task. The classifier is trained on a set of positive and negative examples of relations [18]. Both “simple” and more complicated word-level features (for example, parse trees) can be used. In this case, the notion of similarity between the objects K(x,y) is introduced. x and y can be such objects as strings, word sequences, trees, etc. A formal definition can be found in [32, Section 2.2: Kernel Methods].

In [37], the maximum entropy method is applied. As features, various lexical, syntactic and semantic features are taken. The data sets of ACE (Automatic Content Extraction) are used. The results presented in the article are comparable with the best results of other participants of the ACE.

In [38], a support vector machine (SVM) and polynomial and linear kernels are used to classify different types of relationships. The text processing includes several stages: tokenization, sentence parsing, and dependency analysis. At each stage, the kernel function is used for presenting the information. The method for determining kernel functions is described in [35]. The results show that each additional kernel function improves the values of precision, recall, and the F-score. When all five kernel functions are used, the values are the following: P = 69.23%, R = 70.5%, F-score = 70.35%.

Semi-supervised methods are described in [39-42]. In [39], the DIPRE system (Dual Iterative Pattern Relation Expansion) is described. Its specific feature is that it recognizes the relationships of the type (author, article) from the Web. The system starts with a small set of ready pairs called seeds and applies the following algorithm:

1. Take strings that contain a seed (the string s contains the seed (x,y), if either x or y is contained in s).

2. Derive the patterns from the found strings. The patterns are essentially regular expressions derived from strings.

3. Apply patterns to strings to get a new set of pairs (author, article).

4. Add the new set of pairs to seeds and repeat the procedure, until a certain criterion is met, for example, no new relations are found.

The Snowball system [40] is in essence similar to the DIPRE system, but it does not rely on exact text matching, introducing the concept of weights that makes it possible to adjust the system in cases of the difference in punctuation, etc.

One of the common disadvantages of all these systems is that they search for relations in a sentence, and not between several sentences or even throughout a document. In [43], a method of distant supervision was proposed, which uses for training Freebase [44] – a large semantic database with thousands of relationships.

The essence of the method is to go through the sentence and extract named entities such as people, organizations, and geographic locations. If we found a pair of entities, and this pair is contained in Freebase, we extract the features from this sentence and add them to the features of the found relation. The method is based on the assumption that if two entities are involved in the relation, then any sentence which contains these entities can express this relation. Individual sentences can return incorrect results; therefore, the multiclass logistic regression classifier is used. The feature vector is supplemented from each sentence, where the entities occurred together. For example, if two entities occurred in 10 sentences, and for each of them 3 features were extracted, the total amount of features will be equal to 30. Then, the categorizer is started, which determines the relation, to which the pair of entities belongs, on the basis of features. This work identified the lexical features (sequence of words between two entities, parts of speech of these words, the window of k words to the left and to the right of the entities, etc.) and syntactic features: the dependencies between words, and the named entity type for an entity.

The main feature of the method is that it uses a combination of information from different mentions of one and the same relation. When evaluating the results, the authors conclude that combining syntactic and lexical features shows better results than using these features separately.

[45] proposes a method for extracting relations not locally in the sentence, but within a set of documents (relation extraction across documents). The work relies on the ideas of the work [43] and uses Freebase and Wikipedia.

The method performance is determined by precision, recall and the F-score (F_1 score) which are defined in the same way as for the NER task. The only difference is that entities are replaced by relations.

It should be noted that for supervised methods we have a marked up corpus and can precisely calculate P, R, and F. However, for semi-supervised methods, it is difficult to use these metrics directly. In [39], a method for such calculations is proposed. Let us also note, that in [43] the results were checked by human using Amazon's Mechanical service Turkservice [46].

7. DISCUSSION

The main goal of this paper was the study of the modern methods of processing texts in natural language used for solving NER tasks and identifying relations. It should be noted that the machine learning methods and hybrid methods have been most widely used recently. The advantage of methods based on using rules for NER is that they usually ensure high precision, but at the same time they are characterized by low recall. This can be explained by the fact that the domain expert is able to create good rules for this domain. However, it is not possible to write the rules for each and every case; therefore, these methods give low recall over time.

The advantage of methods based on machine learning is that there is no need in determining the rules. However, it is necessary to mark up a large training corpus in order that the methods give adequate results. One more problem is related to the fact that it is necessary to competently select the features and avoid overtraining.

There is also a problem of the obsolescence of training data that over time has a direct impact on results. To overcome this, one can apply methods using online encyclopaedias, such as Wikipedia and Freebase.

Hybrid methods use both rules and machine learning, which ensures an increase of the F-score gains by several percent.

There are also advanced methods for identifying relations that use the context of the entire document. Basically, machine learning and semi-supervised methods are most widely used for solving this task.

8. CONCLUSION

In this study we have identified the main text processing methods by the examples of NER tasks and by identifying relations. Constraints and recommendations on the use of each method have also been formed. Depending on the mathematical apparatus used, three groups of methods have been pointed out: methods based on rules; methods using machine learning algorithms (statistical or probabilistic ones); hybrid methods combining advantages of the algorithms of the first and second groups.

Using the example of NER tasks and relation extraction tasks, the authors analyzed the modern methods of natural language processing.

Originally, a rule-based approach was applied. This is a rather simple method, since there is no need in creating training samples; it is possible to create a few relatively simple rules and get a result at once. It was shown that rule-based methods often ensure high precision, but at the same time they are characterized by low recall. However, as the requirements for such systems are increased, creating the rules becomes quite a difficult task, and it becomes impossible for the rules to cover all possible situations.

Currently, the machine learning methods are of special interest. These methods ensure both high precision and high recall. The advantage of these methods is that there is no need to create rules manually and monitor their correctness. One of the disadvantages is the need for the manual mark-up of training data by experts. As known, to ensure high performance of methods, large training sets are needed. One more problem is so-called overtraining. An overtrained model generates good forecasts on the training data, but can make serious mistakes on the test data. Usually, this is related to the selection of a very large number of features. One of the ways to overcome this problem implies selecting an optimal amount of features.

The problem of the manual mark-up of training corpora can be overcome by automatic creation of corpora. For this purpose, one can use online encyclopedia such as Wikipedia and Freebase. In a number of works, it has been shown that this method gives good results.

The authors also analyzed the hybrid methods using the rules and methods of training for the application of these rules. It was shown that these approaches ensure an increase in the quality by a few percent.

In contrast to the existing reviews, this study considered methods using electronic online-libraries for automatic marking of text corpora necessary for learning algorithms.

A further subject of research may be the assessment of the computational complexity of these methods. It is evident that if a method is good but it takes exponential time from the input data, it is of small practical importance; therefore, it is advisable to understand the relationship between the method performance and its computational complexity.

9. ACKNOWLEDGEMENTS

The work was financially supported by the Ministry of Education and Science of the Russian Federation (Grant Agreement 14.579.21.0088, the unique identifier for applied scientific research RFMEFI57914X0088).

REFERENCES:

[1] Grishman, Ralph; Sundheim, B. 1996. Message Understanding Conference – 6: A Brief History. In Proc. International Conference on Computational Linguistics.

[2] J. Hobbs, D. Appelt, J. Bear, D. Israel, and M. Tyson, (November 1992). FASTUS: A System For Extracting Information From Natural-Language Text. 519. AI Center, SRI International, 333 Ravenswood Ave., Menlo Park, CA 94025.

[3] Jenny Rose Finkel, TrondGrenager, and Christopher Manning, 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, ACL '05, pages 363–370, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[4] Rosenfeld, B., Feldman, R., Fresko, M., Schler, J., Aumann,Y., 2004. TEG – A Hybrid Approach to Information Extraction.Proc. of the 13th ACM.

[5] Fresko, M., Rosenfeld, B., and Feldman, R., 2005. A hybrid approach to NER by MEMM and manual rules.Proceeding s of the 14th ACM international conference on Information and knowledge management, ACM, 361–362.

[6] Limin Yao, Sebastian Riedel, and Andrew McCallum, 2010. Collective cross-document relation extraction without labelled data. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing EMNLP-2010), pages 1013–1023.

[7] Sekine S., Nobata C., 2004. Definition, Dictionaries and Tagger for Extended Named Entity Hierarchy. // LREC. 2004.

[8] Nadeau D., Sekine S., 2007. A survey of named entity recognition and classification // Lingvisticae Investigationes. 2007. Vol. 30, no. 1. P. 3–26.

[9] Poibeau T., Kosseim L., 2001. Proper name extraction from non-journalistic texts // Language and computers. 2001. Vol. 37, no. 1. P. 144–157.

[10] Christopher Manning, 2006. Doing Named Entity Recognition? Don't optimize for F1. August 2006.

[11] L. F. Rau., 1991. Extracting company names from text. In Proc. of the Seventh Conference on Artificial Intelligence Applications CAIA-92 (Volume I: Technical Papers), pages 29–32, Miami Beach, FL, 1991.

[12] P. Jackson, and I. Moulinier., 2002. Natural language processing for online applications. Text retrieval, extraction and categorization. Natural Language Processing Benjamins, Amsterdam, Philadelphia, (2002), p. 81

[13] Chomsky, N. (1959). On certain formal properties of grammars. Information and Control, 2, 137–167.

[14] Alexander E. Richman and Patrick Schone, 2008. Mining Wiki Resources for Multilingual Named Entity Recognition," ACL'08, 2008.

[15] https://en.wikipedia.org/wiki/Hyperplane

[16] JohnC.Platt. 1999. Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods.In B. Scholkopf A. Smola, P.BartlettandD.Schuurmans,editors, Advancesin Large Margin Classifiers. MIT Press.

[17] Vapnik, V. (1995) The Nature of Statistical Learning Theory. Springer-Verlag, London.; Bishop, Christopher M. (2006). Pattern Recognition and Machine Learning.Springer.

[18] https://en.wikipedia.org/wiki/Multiclass_classification

[19] H. Isozaki, and H. Kazawa, 2002. Efficient Support Vector Classifiers for Named Entity Recognition. ProceedingsoftheInternational Conference on Computational Linguistics, (2002)

[20] James Mayfield, Paul McNamee, and Christine Piatko, 2003. Named entity recognition using hundreds of thousands of features. In Walter Daelemans and Miles Osborne, editors, Proceedings of CoNLL-2003, pages 184–187. Edmonton, Canada, 2003.

[21] D. Bikel, S. Miller, R. Schwartz, and R. Weischedel, 1997. Nymble: a High-Performance Learning Name-finder. Proc. of the 5th conference on Applied Natural Language Processing (ANLP 97), (1997)

[22] G. Zhou, and J. Su, 2002. Named Entity Recognition using an HMM-based Chunk Tagger. Proceedings of the 40th Annual Annual Meeting of the Association for Computational Linguistics (ACL'02), Philadelphia,PA, (2002)

[23] https://en.wikipedia.org/wiki/Entropy_ (information_theory)#Definition

[24]https://en.wikipedia.org/wiki/Bernoulli_ distribution

[25] Hai L. Chieu and Hwee T. Ng., 2002. Named entity recognition: a maximum entropy approach using global information. In Proceedings of the 19th international conference on Computational linguistics, pages 1–7, Morristown, NJ, USA, 2002. AssociationforComputationalLinguistics.

[26] Romanenko, A.A., 2011. Applying Conditional Random Fields to the Tasks of Natural Language Processing (Graduation Thesis). Moscow: Moscow Institute of Physics and Technology.

[27] James R. Joel Nothman, 2008. Transforming Wikipedia into Named Entity Training Data.pages 124–132, 2008.

[28] Jun'ichiKazama and Kentaro Torisawa, 2007. Exploiting Wikipedia as External Knowledge for Named Entity Recognition. In Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pages 698–707, 2007.

[29] Lev Ratinov and Dan Roth, 2009. Design challenges and misconceptions in named entity recognition. In CoNLL '09: Proceedings of the Thirteenth Conference on Computational Natural Language Learning, pages 147–155, Morristown, NJ, USA,2009. Association for Computational Linguistics.

[30] Abhijit Bhole, Blaz Fortuna, Marko Grobelnik, and DunjaMladenic, 2007. Extracting named entities and relating them over time based on wikipedia. Informatica (Slovenia), 31(4):463–468, 2007.

[31] WisamDakka and Silviu-PetruCucerzan, 2009. Augmenting Wikipedia with Named Entity Tags, 2008.

[32] A. Toral and R. Munoz, 2006. A proposal to automatically build and maintain gazetteers for Named Entity Recognition by using Wikipedia.EACL 2006, 2006

[33] Sam Tardif, James R. Curran, and Tara Murphy, 2009. Improved Text Categorisation for Wikipedia Named Entities. In Proceedings of the Australasian Language Technology Association Workshop 2009, pages 104–108, Sydney, Australia, December 2009.

[34] Christian Bohn and KjetilNorvag, 2010. Extracting named entities and synonyms from wikipedia. In AINA, pages 1300–1307. IEEE Computer Society, 2010.

[35] Nguyen Bach and Sameer Badaskar, 2007.A review of relation extraction. Literature review for Language and Statistics II.

[36] McDonald, R., Pereira, F., Kulick, S., Winters, S., Jin, Y., & White, P., 2005. Simple algorithms for complex relation extraction with applications to biomedical ie. ACL '05: Proceedings of the43rd Annual Meeting on Association for Computational Linguistics (pp. 491–498).Ann Arbor, Michigan.

[37] Kambhatla, N., 2004. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations.Proceedings of the ACL 2004.

[38] Zhao, S., Grishman, R., 2005. Extracting relations with integrated information using kernel methods. Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (pp. 419–426).

[39] Brin, S., 1998. Extracting patterns and relations from the world wide web. WebDB Workshop at 6th International Conference on Extending Database Technology, EDBT ’98.

[40] Agichtein, E., &Gravano, L., 2000. Snowball: Extracting relations from large plain-text collections. Proceedings of the Fifth ACM International Conference on Digital Libraries.

[41] Etzioni, O., Cafarella, M., Downey, D., Popescu, A. M., Shaked, T., Soderland, S., Weld, D. S., Yates, A., 2005. Unsupervised Named-Entity Extraction from the Web: An Experimental Study. Artificial Intelligence (pp. 191–134).

[42] Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., Etzioni, O., 2007. Open informatio extraction from the web. IJCAI ’07: Proceedings of the 20th International Joint Conference on Artificial Intelligence. Hyderabad, India.

[43] Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky, 2009. Distant supervision for relation extraction without labeled data. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP (ACL ’09), pages 1003–1011. Associationfor Computational Linguistics.

[44] https://ru.wikipedia.org/wiki/Freebase

[45] Limin Yao, Sebastian Riedel, and Andrew McCallum, 2010. Collective cross-document relation extraction without labelled data. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing EMNLP-2010), pages 1013–1023.

[46] https://en.wikipedia.org/wiki/Amazon_ Mechanical_Turk.