Analysis and software implementation of the methods for extracting semantic proximity between the lexical units

S. Leoshchenko, A. Oliinyk, S. Subbotin

Abstract— This paper presents one of the ways solution to the problem of lexical ambiguity: the using methods of extraction and analysis of semantic relations between lexical units. The proposed methods are divided into groups, among which identified and discussed the most popular. As well, special attention is paid to the methods of k-nearest neighbor and mutual k-nearest neighbor, which allow to introduce the problem of extracting semantic proximity as one of the classification tasks.

Keywords—**S**oftware tools, semantic relations, semantic proximity, methods and metrics, software system, information extraction, Wikipedia, computational lexical semantics

I. INTRODUCTION

Today artificial intelligence is one of the most popular branches of science and technology. Moreover, every day, even ordinary people, who are not closely linked with science, are increasingly using this phrase. This is simply explained by the fact that more and more people use the device which in varying degrees use the tools and technologies of artificial intelligence.

But even the using of such advanced information technology, often does not allow to solve problems that the ordinary person can decides on a subconscious level. For example, choosing the most correct option from the list of synonyms given the content or the stylistic feature of the context. In such a situation, people in most cases will be able to find the exact variant in less than a second, in contrast to any software product that in turn reduces the efficiency of the use of, for example, automatic translators. One solution to this problem is the determination of semantic proximity. This study focuses on this method of solving a highly relevant problem.

II. ANALYSIS EXISTING METHODS OF EXTRACTING SEMANTIC RELATIONSHIPS

One of the classes of methods of solving the problem of lexical ambiguity are methods based on external knowledge sources. In turn, these methods can be divided into categories, and one category of such methods are methods based on the degree of semantic proximity calculated on the basis of semantic networks. In this category plays an important role completeness and timeliness of the network, and calculated measures of semantic proximity. By itself, the problem of computing semantic proximity resource intensive and requires considerable computing power.

To work with calculation methods of semantic proximity, it is necessary to highlight the existing semantic relations between words in natural languages. Such relationships are: synonyms, meronyms, antonyms, associations, etc.

A. Panchenko, S. Adeykin, A. Romanov and P. Romanov gives examples of successful application of semantic proximity in their works: such relations are successfully used in various NLP applications, such as word sense disambiguation, query expansion, document categorization or question answering [5].

However, existing resources in most cases are unavailable to ordinary users and to evaluate their effectiveness will be possible only according to given scientific articles. Moreover, the creation of the needed semantic resources in the manual would be too time-consuming, lengthy and costly process. This implies that the development and the subsequent creation of methods for automatic extraction and analysis of semantic relations is extremely important.

The most common method of extracting semantic relations based on lexico-syntactic patterns that are created manually [5]. This approach has many disadvantages, chief among which is the complexity of writing rules to

S. Leoshchenko Zaporizhzhya National Technical University, Zaporizhzhya, Ukraine (e-mail: sedrikleo@gmail.com).
A. Oliinyk Zaporizhzhya National Technical University, Zaporizhzhya, Ukraine (e-mail: olejnikaa@gmail.com).
S. Subbotin Zaporizhzhya National Technical University, Zaporizhzhya, Ukraine (e-mail: subbotin.csit@gmail.com).

derive the relations. From this disadvantage, we can conclude that the use of such rules will be correct with respect to only one language, others will have to create new rules. So there are methods based on distribution analysis [6,7]. Their advantage is that they do not require manual work, but the main drawback: low results when used to extract semantic relations [8]. Good results in the tasks demonstrate metrics of retrieve and evaluate semantic proximity, which based in using semantic network in Wikipedia that have been proposed recently [9,10,11]. The relevance of the use of Wikipedia as semantic network analysis is useful because it is one of the largest semantic networks. It covers most of the subject areas, daily updated and supplemented with the users is checked for errors and stores data almost all used languages. And since the existing research Wikipedia was used to test metrics extraction of semantic relations is not often, it can show good results which we had never seen before.

There are different methods based on artificial intelligence [15-20] that can be used for real word processing and analysis. Information technologies using different real word processing methods proposed in [21-25].But proposed in [15-25] methods and information technologies can't be used for semantic proximity extraction between the lexical units.

A. The content measures

As a rule, the content measures of semantic proximity based on the use of Wikipedia articles as vectors in a space of words or terms. Elements of the vector are usually the frequency of occurrence of relevant terms in the text of the articles or their weight, computed according to the scheme TF-IDF:

$$w_{t,d} = tf_{t,d} \cdot idf_{t,d}, \quad (1)$$

where $tf_{t,d}$ is the frequency of occurrence (term frequency) of term t in the text d and

$$idf_{t,d} = \log \frac{|D|}{|\{d \in D \mid t \in d\}|} \quad (2)$$

– inverse term frequency t in megen Wikipedia articles. The semantic proximity of the two articles may be further evaluated as the cosine of the angle between their vector representation.

Method Latent Semantic Analysis (LSA) [13] uses singular value decomposition (SVD-decomposition of the matrix of TF-IDF weights of the terms in the articles to construct the space of the factors (subjects). Further texts can be represented by vectors in the new space to calculate the semantic proximity between them. The development of the LSA method is probabilistic latent semantic analysis (probabilistic LSA pLSA) [14]. Unlike LSA that assumes a normal distribution of documents and terms, pLSA is based on a multinomial distribution and uses a mixed decomposition instead of the SVD decomposition.

One of the drawbacks of LSA is the difficulty of interpretation of the obtained topics. Gabrilovich and Markovich [10] proposed to design the input texts directly on many of the concepts of Wikipedia. Method is called Explicit Semantic Analysis (ESA).

Each Wikipedia concept in the ESA method is represented as a vector of words of the relevant article, weighted according to the scheme TF-IDF. The authors then build an inverted index that displays in the list of articles, where they meet. Concept with a very small weights for a given word are discarded. The input text, denote it by $T = \{w_i\}$ is represented as a TF-IDF vector of weights of words $\langle v_i \rangle$, where v_i is the weight of w_i and then displayed in the space of concepts. Let $\langle k_j \rangle$ is the index entry for the key w_i, where k_j is the TF-IDF weight of the word w_i in the article $c_i, c_j \in c_1, c_2, ..., c_N$, N is the total number of Wikipedia concepts. Then the weight of the concept c_j in the vector that matches the text is calculated as $\sum_{w_i \in T} v_i k_i$. The elements of the received vector reflects semantic similarity of text to appropriate concepts. To calculate the semantic proximity of the two texts, the authors compute the cosine of the angle between the vector representations in the space of concepts.

B. B. Method of the random walk

The random walk model has been successfully used for ranking web pages in Internet search engines, for example, well-known PageRank algorithm [17] based on this model. Olivier in his work [18] carried out a comparative analysis of measures of semantic proximity of Wikipedia articles, based on a model of random walks. He considered several measures based on the measure of green, local PageRank, index cross-referencing and cosine terms of the quality of the ranking of these activities. Grin's measure is determined by the following formula:

$$G_{ij} = \sum_{t=0}^{\infty} \delta_{ij} p^t - v_j . \qquad (3)$$

In this formula, P is the stochastic matrix based on the adjacency matrix, where $P_{ij} = 1 \big/ \sum A_{ik}$, v is own stationary vector of the matrix P . This measure corresponds to the time spent wandering by the user in the node j if it is started from node i . Similar to the Grin's measure, as local PageRank is expressed by the following formula:

$$S_{ij} = \left(\sum_{t=0}^{\infty} c(1-c)^t p^t \right)_{ij} . \qquad (4)$$

In this formula, the random walk process starts from node i , with equal probability clicks on the links of the graph and with probability c , returns to node i . No factor c , S_i turns into a stationary vector of the Markov chain V .

According to the observations of Olivier, the qualitative results give only the modification of the action of the green, and the local PageRank through a very small diameter turns Wikipedia into a global PageRank, which does not carry information about the semantic proximity. One of the most successful methods was the method of $S_{ij} = G_{ij} \log V_j$; on data obtained from the manual ranking of Wikipedia articles, it gave the best results.

A similar approach is used If, in the proposed method, location, PageSim [19].

On computational efficiency of the method of random walks essentially benefit compared with methods a pair of walks, but at a small diameter Wikipedia the use of these measures means the bypass of the entire reference graph of Wikipedia, as for the problem of counting the proximity of two articles and to rank. That is, the computational complexity of this family of measures for both problems is $O(n)$, where n is the number of links Wikipedia. This efficiency is insufficient for the application of these metrics in the practical development of such technology for multilingual text mining, developed by scientists at RAS – Texterra.

C. Wikipedia Link-based Measure

Approach called WLM (Wikipedia Link-based Measure) was presented in [15]. The authors suggested two local metrics to calculate the semantic proximity between the concepts of Wikipedia.

The first metric considers the set of all articles referenced in the initial concept. Suppose we want to calculate the semantic proximity between the concepts A and B . Denote by W the set of all concepts of Wikipedia, and through T is the set of concepts referenced A or B . Each initial concept $s \in \{A, B\}$ and the $t \in T$ is mapped to the value

$$count(s \to t) \cdot \log \frac{|W|}{|\{w \in W \mid w \to T\}|} , \qquad (5)$$

where $count(s \to t)$ is the number of links from s to t , $\{w \in W \mid w \to T\}$ is the set of all articles that link to t . It is easy to see the similarity of this scheme with TF-IDF, but with a difference: instead of the terms of reference are resolved. Semantic proximity of concepts A and B is then computed as the cosine between the $|T|$ -dimensional vectors of weights.

The second metric is based on Google normalized distance (Normalized Google Distance) [16], based in turn, on the count of occurrences of terms that represent the desired concepts in the search query results in Google. Web pages containing both terms signal the presence of semantic links between them. WLM version operates instead of the query results links articles:

$$sim(A, B) = \frac{\log(\max(|in(A)|, |in(B)|)) - \log(|in(A) \cap in(B)|)}{\log(|W|) - \log(\min(|in(A)|, |in(B)|))} \qquad (6)$$

where A, B are concepts, issues, $|in(A)|, |in(B)|$ – a lot of articles that reference A and B respectively, W is the set of all concepts of Wikipedia.

III. A COMPARISON OF EXISTING METHODS

The study of existing computing methods of semantic proximity of concepts based on Wikipedia was discovered unsatisfactory outputs, use the values of the lengths of the shortest paths between concepts in the Wikipedia graph, compared with the state-of-the-art methods (which include themed events and activities on the basis of the random walk). However, it has been suggested that not all types of links contribute equally to the semantic closeness between the concepts, and the more accurate their ranking, which other researchers had not previously carried out, can improve the quality are determined with the help of their estimated location.

Also the study revealed the lack of standard datasets for testing measures of proximity between the concepts of Wikipedia, because the vast majority of methods working with concepts not directly but through the terms or texts. For the latter, in turn, there are standard datasets, and their adaptation to the problem of estimating the semantic proximity of the concepts can have on power in further studies.

IV. STATEMENT OF THE PROBLEM

The main goal of a software product that will be developed can be represented as follows: for each word C_i from the input set of words $C = \{C_1, C_2, C_3, ..., C_n\}$ to find pairs of semantically related words, i.e. $R = \{\langle C_i, C_j \rangle, ..., \langle C_i, C_k \rangle \mid C_i \neq C_j \neq C_k, i, j, k \leq n\}$.

It should also be noted that the methods to be used, do not return the type of the found context, i.e., $R \subseteq C \times C$. Techniques characterized by efficiency, suitability for use with languages available in Wikipedia and sufficient accuracy. The novelty of this work compared to existing research and development is as follows: proposed and investigated and explored new methods for extracting semantic relations from Wikipedia articles based on the algorithms of nearest and mutual nearest neighbors and the two metrics semantic proximity of the words (the cosine of the angle between vectors definitions and the total Lemma in the definitions).

V. ANALYSIS OF METHODS AND METRICS THAT WILL BE USED

A. K-nearest neighbor method

K-nearest neighbor method is a simple non-parametric classification method where a classification of objects within the space of properties use distance (Euclidean, generally), is calculated to all other sites. Select the objects to which the smallest distance, and they are allocated in a separate class.

Method k-nearest neighbors is a metric algorithm for automatic classification of objects. The basic principle of nearest neighbor is that the object is assigned the class which is the most common among the neighbors of this element. The neighbors are taken on the basis of a set of objects whose classes are already known, and, on the basis of the key for the method k values, is calculated, which class is the most numerous among them. Each object has a finite number of attributes (dimensions). It is assumed that there is a certain set of objects with the existing classification [26].

In the most general form, the algorithm of nearest neighbors is:

$$a(x) = \arg\max_{y \in Y} \sum_{i=1}^{m} [x_{i:x} = y] \omega(i; x), \tag{7}$$

where $\omega(i; x)$ is a given weight function, which evaluates the degree of importance of the i-th neighbor for the classification of the object u. So, if $\omega(i; x) = 1$ for $i < k$, the algorithm corresponds to method k-nearest neighbors. But the problem with several possible answers, the maximum amount of votes can be achieved in several classes simultaneously. The ambiguity can be eliminated if the weight function to take a non-linear sequence, such as a geometric progression: in this example, $\omega(i; x) = [i \leq k] q^i$, which corresponds to the method of exponentially weighted k nearest neighbors, and assume $0.5 \leq q \leq 1$[27].

Among the main advantages of this method are:

- the simplicity of the implementation;
- the classification, carried out the algorithm, it is easy to interpret by presenting to the user several objects.

Among the main disadvantages of this method are:

- excessive complexity of decision rule because of the need of storing the training sample;

- the nearest neighbor search involves the comparison of an object is classified, together with all objects of the sample, which requires linear in the length of the sampling operations.

B. Mutual k-nearest neighbor method

The method of mutual nearest neighbors is a method to consider mutually k-nearest neighboring data points, not just the nearest neighbor [28].

Let $N_k(x)$ is the set of k nearest neighbors of x in D_n, $N_k^{'}(x_i)$ the set of k nearest neighbors x_i in $(D_n \setminus \{x_i\}) \cup \{x\}$. A set of mutual k -nearest neighbor (MkNNs) x is defined as:

$$\mathrm{M}_k(x) = \{x_i \in N_k(x) : x \in N_k^{'}(x_i)\}, \tag{8}$$

Then the mutual k -nearest neighbor is defined as:

$$m_n^{MkNNR}(x) = \begin{cases} \frac{1}{M_k(x)} \sum_{i:x_i \in \mathrm{M}_k(x)}^{k} y_i, \mathit{if}\ M_k(x) \neq 0 \\ 0, \mathit{if}\ M_k(x) = 0 \end{cases}, \tag{9}$$

whereas $M_k(x) = |\mathrm{M}_k(x)|$.

In General, if we compare the method of mutual nearest neighbors, with a limit based on the mutual KNN, it should be noted that Mutual KNN better able to identify clusters of various shapes and sizes. Moreover, this method is less prone to noise in the data and can detect deviations [29].

C. Metric number of the common Lemma in the word definitions

The metric uses a measure of semantic proximity based on shared words in the definitions of the two terms.

$$similarity(t_i, t_j) = \frac{2|(d_i \cap d_j)/\, stopwords|}{|d_i| + |d_j|} \tag{10}$$

The numerator of the fraction equals the total number of words in the initial form, without the list of words defined as stop words.
The denominator is the sum of all words in each of the two chosen definitions.
However, this metric does not take into account the length of the definitions. This is the main drawback of the metric, since the length of some definitions can reach up to one hundred rows and have a large number of commonly used terms that are not included in the list of stop words, for example: system, population, range, amount etc [30]

D. Metric the cosine of the angle between the definitions

In order to compensate for the effect of the length of definitions in the connectivity between the terms, used a metric called "the cosine of the angle between vectors definitions". The definition is represented as an N-dimensional vector, and then evaluates the time using the obtained vectors.
The dimension of the vector is determined depending on the nature of the investigated problem. Thus, the vector definition will include only terms from the dictionary. If the dictionary is not found in the definitions, we denote the element becomes zero, in other cases, the element becomes equal to the number of occurrences in the definition of the term.

$$similarity(t_i, t_j) = \frac{f_i \cdot f_j}{\|f_i\| \cdot \|f_j\|} = \frac{\sum_{k=1,N} f_{ik} f_{jk}}{\sqrt{\sum_{k=1,N} f_{ik}^2} \sqrt{\sum_{k=1,N} f_{jk}^2}} \tag{11}$$

In the formula of cosine similarity between the numerator represents the dot product of the two above definitions, and the denominator is equal to the product of the Euclidean norms of these vectors. The denominator in the formula normalizes by the length of the vectors, so the result can be interpreted as the scalar product of normalized vectors corresponding to the two definition.

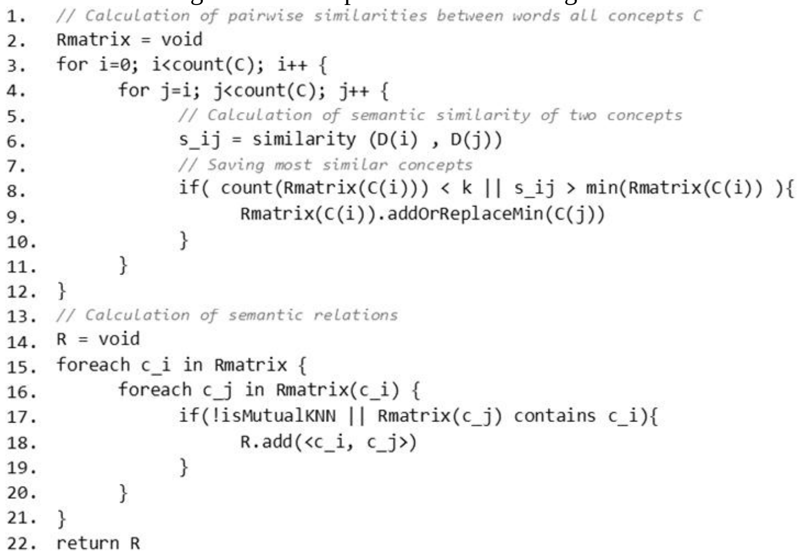
To implement the metrics necessary to pre-treatments:

- the normalization of definition – bringing the definition of each word in the initial form;
- normalization dictionary – bringing each word dictionary in primary forms;
- the search terms used in the definition (the search takes into account all words in terms of the dictionary);
- the formation of vectors definition of that dimension, which would equal the number of terms in the dictionary [30].

VI. The algorithm of the software package

Methods of extraction of semantic relations, which are used based on component analysis [35, 36], the principle of which is that semantically similar words have similar definitions. The proposed algorithms use one of two metrics of similarity denitions, the number of common words [37] or the cosine of the angle between vectors definitions [38]. As input algorithms for extracting semantic relationships take a lot of words C, between which it is necessary to calculate the ratios and their definitions D.

The first version of the algorithm computes semantic relations using the method of nearest neighbors KNN, the second – using the method of mutual nearest neighbors MKNN (Mutual KNN). The only metaparameter algorithms the number of nearest neighbors k. The pseudocode of the algorithms is shown in Fig. 1.



```
1.  // Calculation of pairwise similarities between words all concepts C
2.  Rmatrix = void
3.  for i=0; i<count(C); i++ {
4.      for j=i; j<count(C); j++ {
5.          // Calculation of semantic similarity of two concepts
6.          s_ij = similarity (D(i) , D(j))
7.          // Saving most similar concepts
8.          if( count(Rmatrix(C(i))) < k || s_ij > min(Rmatrix(C(i)) ){
9.              Rmatrix(C(i)).addOrReplaceMin(C(j))
10.         }
11.     }
12. }
13. // Calculation of semantic relations
14. R = void
15. foreach c_i in Rmatrix {
16.     foreach c_j in Rmatrix(c_i) {
17.         if(!isMutualKNN || Rmatrix(c_j) contains c_i){
18.             R.add(<c_i, c_j>)
19.         }
20.     }
21. }
22. return R
```

Fig.1 Pseudocode of the algorithms

Algorithms is the following. First, calculate a measure of semantic proximity of all possible pairs of definitions (line 6). On the basis of the calculated values filled in the array closest words Rmatrix for each definition (lines 1-12). The number of array elements supported is identical to k (number of nearest neighbors) – this allows to strongly reduce memory consumption without loss of information about the connectivity of the words. After filling the array the most similar words for each definition all that remains to be done to retrieve the result set of relations R in KNN method – just fill in the source set, and for the MKNN method is to check for each definition: it is an array of the most similar words pair, and if included – add to the result set (lines 13-21).

The complexity of the algorithms is proportional to the amount fed to the input of the words $|C|$. Time complexity is $O(|C^2|)$, space complexity is also proportional to the number of nearest neighbors k is $O(k|C|)$.

VII. Experiments and results

Investigated the algorithms KNN and MKNN with the two above-described metrics of closeness and different values of the number of nearest neighbors k, the results clearly demonstrated using the diagram in Fig. 2. The results indicate almost linear increase in the number of relationships detected, depending on the parameter k for both algorithms. The number of found relations is only slightly dependent on the used metric. The KNN algorithm gets more relations than MKNN, with an equal number of nearest neighbors k. This is because MKNN removes pairs that are not mutual neighbors, in contrast to the KNN.

Also evaluated the accuracy of algorithms KNN and MKNN for $k = 2$ of the set with 775 definitions. For this was the marked files manually to remote relationships and calculated the precision of the retrieval as the number of true relations to the amount of the withdrawn relationship. The results are shown in the table 1.

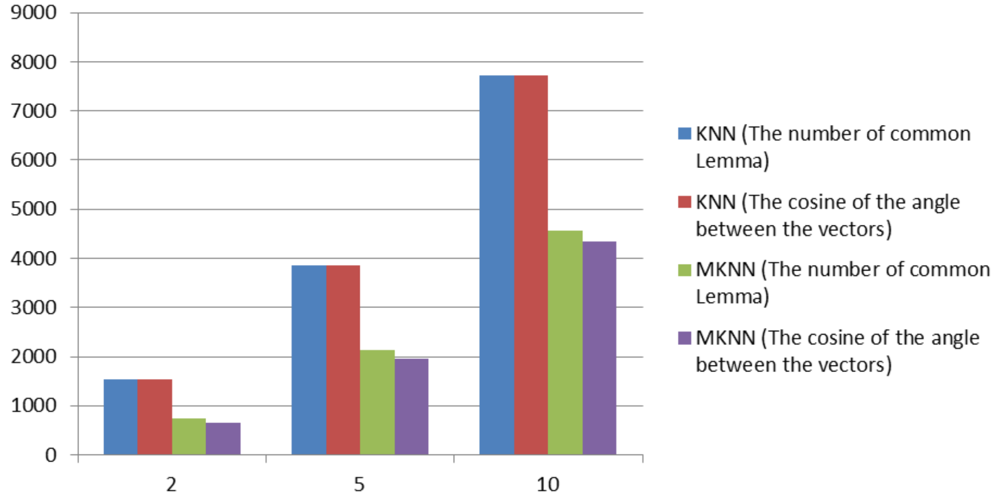


Fig.2 Results

Table 1 The accuracy of extraction methods KNN and MKNN if $k = 2$

Method	The Metric	Is Taken	Correct	Accuracy
KNN	The number of common Lemma	1546	1167	0.761
	The cosine of the angle between the vectors	1546	1167	0.761
MKNN	The number of common Lemma	724	603	0.833
	The cosine of the angle between the vectors	652	499	0.763

Due to the large number of recovered relationships (Fig. 2), evaluation of extraction manually for all values of k is heavy. For large values of k the accuracy of the extraction of the relations needs to decrease. When using the method I recommend using $k \in [1;10]$. In the future I plan to use WordNet and a standard testing set of semantic relations, such as BLESS [39], for a more accurate evaluation of the quality of retrieval.

VIII. Conclusion

The article considers new methods for extracting semantic relations. Despite the fact that this are simple methods for data mining, they demonstrate very good results. To assess the quality of methods used universal metrics which complement each other. The simplicity of the proposed methods and metrics ensures stable operation and high performance of the developed software.

Developed a software package "SemAnalysis", implements the basic functions: analysis and extraction of semantic relations, namely semantic proximity between words. This program uses methods: nearest neighbors and mutual nearest neighbors and metrics: the total number of LEM in the definitions and the cosine of the angle between vectors definitions. Also the software package uses a large data base of terms and their definitions.

Due to the fact that the program is created using C++ programming language, it can be running on the computer running Windows or UNIX-like operating system.

Developed the software package has a practical focus and is ready to use.

During the development of the program and analysis of the developed program complex has some ideas for improvement such as:

- the use of modern document-oriented database system – MongoDB;
- introduction parallel programming algorithm of the program.
- continued use of popular semantic networks – WordNet and Wikipedia.

New opportunities will be able to expand the functionality of the software and enhance its performance.

ACKNOWLEDGMENT

This article is the result of the project implementation: TEMPUS CERES Centers of Excellence for young RESearchers, reg.no. 544137-TEMPUS-1-2013-1-SK-TEMPUS-JPHES

REFERENCES

[1] Patwardhan S., Pedersen T. Using WordNet-based context vectors to estimate the semantic relatedness of concepts. EACL, 2006. P. 1-12.

[2] Hsu M.H., Tsai M.F., Chen H.H. Query expansion with conceptnet and wordnet: An intrinsic comparison. Information Retrieval Technology. LNCS, 2006. P. 1–13.

[3] Tikk D., Yang J.D., Bang S.L. Hierarchical text categorization using fuzzy relational thesaurus. Prague: KYBERNETIKA, 2003. P. 583–600.

[4] Sun R., Jiang J., Fan Y. Using syntactic and semantic relation analysis in question answering. In Proceedings of the. TREC, 2005. 243 p.

[5] Panchenko A., Adeykin S., Romanov A., Romanov P., Extraction of Semantic Relations between Concepts with KNN Algorithms on Wikipedia. URL: http://it-claim.ru/Projects/RFH/Publications/2%20kv/2012%2022%202%20Panchenko.pdf

[6] Hearst M.A., Automatic acquisition of hyponyms from large text corpora, Proceedings of the 14th conference on Computational linguistics. COLING, 1992. P. 12-18.

[7] Lin D. Automatic retrieval and clustering of similar words. Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics. Montreal: COLING-ACL, 1998. P. 768-774.

[8] Heylen K., Peirsman Y., Geeraerts D., Speelman D. Modelling word similarity: an evaluation of automatic synonymy extraction algorithms. Proceedings of the Sixth International Language Resources and Evaluation. Leuven: LREC, 2008. P. 3243-3249.

[9] Curran J.R., Moens M. Improvements in automatic thesaurus extraction. Proceedings of the ACL-02 workshop on Unsupervised lexical acquisition. ACL, 2002. P. 59-66.

[10] Strube M., Ponzetto S.P. WikiRelate! Computing semantic related-ness using Wikipedia. Proceedings of the National Conference on Artificial Intelligence. Heidelberg: AAAI Press, 2006. P. 1419-1429.

[11] Gabrilovich E., Markovitch S. Computing semantic relatedness using wikipedia-based explicit semantic analysis. International Joint Conference on Artificial Intelligence. Israel: Israel Institute of Technology, 2007. P. 12-20.

[12] Zesch T., Müller C., Gurevych I. Extracting lexical semantic knowledge from wikipedia and wiktionary. Marrakech: LREC, 2008. P. 1646–1652.

[13] Resnik P . Using information content to evaluate semantic similarity in a taxonomy. San Francisco: Morgan Kauffman Publishers Inc., 1995. P. 448–453.

[14] Deerwester S. C. et al. Indexing by latent semantic analysis. Tokio: JASIS, 1990. P. 391-407.

[15] Hofmann T. Probabilistic latent semantic analysis. San Francisco: Morgan Kaufmann Publishers Inc., 1999. P. 289-296.

[16] Witten I., Milne D. An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. Chicago: AAAI Press, 2008. P. 25-30.

[17] Cilibrasi R. L., Vitanyi P. M. B. The google similarity distance. Amsterdam: IEEE Transaction, 2007. P. 370-383.

[18] Lee M. D., Pincombe B. M., Welsh M. B. An empirical evaluation of models of text document similarity. Cognitive Science, 2005. P. 12-18.

[19] Velikhov P.E. Measures of semantic proximity of Wikipedia entries and their application at text processing. SPb.: Technologies and Computer Systems, 2009. P. 23-37. (In Russian).

[20] Akiba T., Iwata Y., Yoshida Y. Fast exact shortest-path distance queries on large net-works by pruned landmark labeling. New York: ACM, 2013. P. 349-360.

[21] Székely G. J. et al. Brownian distance covariance //The annals of applied statistics. Annals of Applied Statistics, 2009. P. 1236-1265.

[22] Bellet A., Habrard A., Sebban M. A Survey on Metric Learning for Feature Vectors and Structured Data. arXiv preprint arXiv, 2013. 59 P.

[23] Xing E. P. et al. Distance metric learning with application to clustering with side-information. California: NIPS, 2003. P. 521-528.

[24] Senellart P., Blondel V. D. Automatic Discovery of SimilarWords. Survey of Text Mining II. London: Springer, 2008. P. 25-44.

[25] WordNet. URL: https://wordnet.princeton.edu/.

[26] Textterra. URL: http://www.ispras.ru/proceedings/isp_26_2014_1 /isp_26_2 014 _1_421/.

[27] K Nearest Neighbor Algorithm. URL: http://www.d.umn.edu/~deoka001/downloads/K_Nearest _Neighbor _ Algorithm.pdf.

[28] Introduction to k-nearest neighbors : Simplified. URL: https://www.analyticsvidhya.com/blog/2014/10/introduction-k-neighbours-algorithm-clustering/.

[29] Kim Hyun-Chul Bayesian Kernel and Mutual k-Nearest Neighbor Regression. Beijing: IGARSS 2016. P. 32-40.
[30] Ruan J. A Fully Automated Method for Discovering Community Structures in High Dimensional Data. Department of Computer Science. San Antonio: One UTSA Circle, 2007. P. 7-15.
[31] Karyaeva M.S. Linguistic and Statistical Analysis of the Terminology for Constructing the Thesaurus of a Specified Field. Modeling and Analysis of Information Systems. Jaroslavl': JaGU, 2015. P. 18-36. (In Russian).