UDC 004.421
DOI: 10.15587/1729-4061.2019.179036

DEVELOPMENT OF THE ALGORITHM OF KEYWORD SEARCH IN THE KAZAKH LANGUAGE TEXT CORPUS

A. Akanova
Master of Informatics
Department of Computer Engineering
and Software
Saken Seifullin Kazakh Agro Technical University
Zhenis ave., 62, Nur-Sultan, Kazakhstan, 010000
E-mail: akerkegansaj@mail.ru

N. Ospanova
PhD, Associate Professor, Head of Department
Department of Information Technology
S. Toraighyrov Pavlodar State University
Lomova str., 62, Pavlodar, Kazakhstan, 140008
E-mail: nazirs_n@mail.ru

Y. Kukharenko
PhD, Associate Professor, Head of Department
Department of Information
communication Technology
M. Kozybayev North Kazakhstan State University
Pushkin str., 86, Petropavlovsk,
Kazakhstan, 150000
E-mail: genylapteva@mail.ru

G. Abildinova
PhD
Department of Information Technology
L. N. Gumilyov Eurasian National University
Satpaev str., 2, Nur-Sultan, Kazakhstan, 010008
E-mail: gulmira_2181@mail.ru

Питання семантичного аналізу тексту займає особливе місце в комп'ютерній лінгвістиці. Дослідники даної області мають підвищений інтерес до розробки алгоритму, використання якого дозволить підвищити якість обробки корпусу тексту та ймовірнісне визначення змісту тексту. Результати дослідження застосувань методик, підходів, алгоритмів для семантичного аналізу тексту у комп'ютерній лінгвістиці в міжнародній і казахстанській науці призвели до розробки алгоритму пошуку ключових слів в тексті казахською мовою. Першим етапом алгоритму було складання еталонного словника ключових слів для корпусу тексту українською мовою. Вирішенням цієї проблеми стало застосування алгоритму Портера (стеммера) для корпусу текстів казахською мовою. Реалізація стеммера дозволила виділити унікальні основи слів і отримати еталонний словник, який згодом проіндексували. Наступний крок – це збір навчальних даних із корпусу текстів. Для обчислення ступеня семантичної близькості між словами кожному слову присвоюється вектор відповідних йому словоформ еталонного словника, в результаті якого виходить пара – ключове слово і вектор. І останнім кроком алгоритму є навчання нейронних мереж. При навчанні застосовується метод зворотного поширення помилок, що дозволяє провести семантичний аналіз корпусу тексту і отримати ймовірнісну кількість слів, близьку до очікуваної кількості ключових. Цей процес дозволяє автоматизувати обробку текстового матеріалу шляхом створення цифрових навчальних моделей ключових слів. Алгоритм використовується для розробки нейрокомп'ютерної системи, що буде проводити автоматичну перевірку текстових робіт учнів онлайн курсів. Унікальністю алгоритму пошуку ключових слів є застосування навчання нейронної мережі для текстів казахською мовою. У Казахстані вченими в області комп'ютерної лінгвістики було проведено ряд досліджень на основі застосування морфологічного аналізу, лемматизації та інших підходів і реалізовані лінгвістичні інструменти (в основному словники-перекладачі). Область застосування навчання нейронних мереж для синтаксичного аналізу казахської мови залишається відкритим питанням в казахстанській науці.

Розроблений алгоритм передбачає вирішення однієї з проблем в отриманні ефективного семантичного аналізу тексту казахською мовою

Ключові слова: ключове слово, алгоритм Портера, семантичний аналіз, нейронна мережа

Received date 04.07.2019
Accepted date 23.09.2019
Published date 31.10.2019

1. Introduction

In modern research [1–3] in the field of computational linguistics using artificial intelligence, a special place is occupied by the development of methods and tools for automated text processing. The first systems consisted mainly of large bilingual dictionaries, where the words of the source language gave one or more words of another language, taking into account syntax rules. These systems were subsequently “considered complex and stressed the need for developing systematic methods, which led to the creation of syntactic ordering rules”.

Research in computational linguistics has reached the level of high intelligent technology. Most studies are aimed at solving the problems of machine translation, indexing, abstracting, classification and categorization of documents in full-text search. Computational linguistics combines knowledge of computer science and linguistics. The complexity of natural language modeling covers morphological, syntactic, phonological levels of language. The main problem in this area is the creation of artificial intelligence systems for natural language processing. Computational linguistics studies the creation and use of electronic text corpora, creation of electronic dictionaries, thesauruses, ontologies, machine translators, information extraction from texts, automatic abstracting and building knowledge management systems.

Computational tasks and problems in computational linguistics are discussed by scientists at conferences organized by the Association for Computational Linguistics (ACL – aclweb.org). In addition, the actual platform for discussing new research and results is the Dialogue International Conference on Computational Linguistics (dialog-21.ru) and the International Conference on Computational Linguistics and Intelligent Text Processing (cicling.org).

One of the goals of scholars studying text mining, text processing automation, semantic relations of words and sentences in texts, is to create an intelligent tool for evaluating essays, written works and other written creative works of learners. One of the solutions to such problems is an effective cognitive tool for automated text processing – Automated Essay Scoring [4]. There are four types of AES systems that are widely used by testing companies, universities and public schools: Project Essay Grader (PEG), Intelligent Essay Assessor (IEA) (Measurement Incorporated (MI), USA), E-rater (ETS, USA) and IntelliMetric (Vantage Learning, USA) [5].

From the analysis of the studies, we see that widely studied natural languages in computational linguistics are English, Russian, German, Chinese, French, Spanish, Turkish. But one of the little-studied is the Kazakh language. Lack of knowledge of computational linguistics problems of Kazakh language text processing is one of the reasons for beginning research on the development of a keyword search algorithm. The Kazakh language belongs to agglutinative languages, for which many keyword search and dictionary compiling algorithms are already available, especially for the Kipchak and Turkish languages. But deep neural network learning when processing Kazakh language text corpora has not been applied. Thus, the use of deep neural network learning for keyword search, full-text search for Kazakh language text corpora emphasizes the relevance of the topic in the field of computational linguistics.

2. Literature review and keyword search problem statement

In the field of computational linguistics, a lot of research has been carried out, which resulted in translation dictionaries, thesauruses, Internet search engines. The basis of these results is the development and application of methods and algorithms of semantic text analysis, extraction and use of knowledge for intelligent computer analysis. Research in the field of automated text processing begins with a study of the structure of natural language, which includes some types of text analysis: pre-semantic, graphematic, syntactic, fragmentation, morphological lemmatization. The application of the above-mentioned analyses for the Russian language can be seen on the website at.com.

In [6], an evolutionary neurodynamic basis for developing a learning process based on visual recording and extraction of neural weights through neurodynamic experiments while passing the reference content was presented. Also, in [7], the model for extracting uncontrolled relations is considered, which is called relation distribution presentation. Relation distribution presentation is aimed at the automated study of entity vectors and further assessment of semantic similarity between the entities. The study [8] proposes a new technique for recognizing individual question words from the speech query of a South Indian language. In this study, Fourier transform (FFT) and discrete cosine transform (DCT) are used for feature extraction, and artificial neural network (ANN) is used for classification and recognition.

The creation of a thesaurus-based intelligent search engine was considered in [9] and an approach to creating semantic metrics and establishing a semantic relation between certain terms was proposed. The paper [10] presents a connection classification approach using context-semantic functions and LFNN-based incremental learning algorithm for text classification. The proposed method allows the classifier to dynamically study the model in a dynamic database. This learning process uses the Back Propagation Lion (BPLion) neural network, including a fuzzy constraint and Lion algorithm (LA) for possible weight selection. A study on natural language processing by automated correction was proposed for Chinese texts in [11]. An algorithm for the automated verification of text proofreading was presented.

The ontology method is often used to implement data classification and data relationship in semantic analysis in automated text processing [12]. In [13], a multi-agent approach with the interaction of two agents was considered: the first corresponds to significant units of extracted information and the second rule agent implementing replenishment of the given ontology based on the semantic-syntactic language model. The study [14] used semantic networks in the extraction and visualization of knowledge, verb graphs with relational graphs to implement first-order logic.

In the early 2000s, scientists began to more explore and apply latent semantic analysis in small-scale corpora for automated assessment of academic essays [15]. Also, latent semantic analysis was applied [16] to implement the method of automated text summarization used to assess the relevance of a sentence.

Many tasks of semantic text analysis, such as text search, text summarization and text comparison, depend on extracting weight keywords from the text corpus. In [17], a graph text model is proposed that allows estimating frequency characteristics of text words taking into account the location of word pairs. Given this data model, the paper proposes an algorithm for determining text keywords. The algorithm takes into account Russian language words that satisfy two conditions: the word consists of at least 4 letters; the word is recognized by the morphological analyzer as a noun. The main goal of keyword extraction in computational linguistics is to determine the semantic relation of words in different text corpora. For example, researchers [18] proposed an algorithm for keyword extraction from patent documentation (PKEA-Patent Keyword Extraction Algorithm), based on a distributed skip-gram model for patent classification. To achieve the goal, standard reference data sets and a self-made patent data set were used to evaluate PKEA performance.

For searching and determining semantically related words, some researchers [19] used the cuckoo search optimization algorithm in combination with the response generator algorithm to increase the semantic accuracy of the sentences found.

At the moment, there are many studies on keyword search, different methods, approaches, as well as a learning algorithm for training to classify positive or negative examples of key phrases, have been developed. To this end, the GenEx algorithm has been specially developed, which is reflected in [20] and includes specialized knowledge of the procedural domain, having the greatest success in keyword extraction than a conventional algorithm. As a result of the literature review, it can be concluded that each of the studies reflects work in the field of automated text processing, organization of semantic text analysis, using different algorithms and models of keyword extraction. However, the issue of compiling a dictionary of key phrases and a base of keywords-word forms of the required natural language remains unaddressed in these works. And the main point for a semantic analysis of the text in the Kazakh language was the compilation of a dictionary of key phrases. Key phrases are structural units of the text, which to some extent are important components in text transmission. And most often, sets of keywords and phrases usually contain the most important information for understanding the meaning of the text and form a general idea.

As a result of the study, it can be said that at the moment there are unresolved issues related to the development of intelligent tools for semantic analysis of texts in the Kazakh language. The reason is objective difficulties associated with the lack of algorithms for keyword search in the Kazakh language text corpus. An option to overcome these difficulties may be to develop an algorithm using deep neural network learning for the Kazakh language. Deep neural network learning and methods for its implementation were proposed in [21], where they use this approach to understand video semantics. In [22, 23], deep neural network learning is used for end-to-end text detection to restore and improve images. However, the use of deep neural network learning to search for keywords for semantic text analysis was not reflected in the studied works.

Deep neural network learning is widely used for graphic image processing [24]. Deep learning of artificial neural networks, which was first used in 2006 has taken an important place in computational linguistics [25]. To date, neural network learning methods have been developed allowing to quickly and efficiently train networks consisting of one hundred or more layers [26].

All this suggests that it is advisable to conduct a study on the development of an algorithm for keyword search in the Kazakh text corpus using deep neural network learning. The development of the algorithm was required for further development of a neurocomputer system with checking Kazakh text works of learners.

3. The aim and objectives of the study

The aim of the study is to develop an algorithm for keyword search in the Kazakh text.

To achieve the aim, the following objectives were set:

– to bring the text corpus into machine-readable form, which includes the definition of words/word forms of the Kazakh language with the help of Porter stemmer;

– to collect data and conduct neural network learning.

4. Bringing text corpora into machine-readable form

A dump of Wikipedia database in the Kazakh language as of April 2019 was used as a text corpus. For further work, it was necessary to compile a dictionary of unique word forms.

The original dump file was previously cleared of XML function words and repetitions. As a result, a dictionary of 1062058 words was obtained. To reduce it, a stemmer based on the Porter algorithm was developed and applied.

The stemming algorithm (Porter stemmer) is often used in approaches to complex word identification [27], file manipulation, search and scripts for specific applications [28], it does not use bases of word stems, but only, sequentially applying a series of rules, strips endings and suffixes based on language features, and therefore works quickly.

The idea of the Porter stemmer is that there is a limited number of inflectional and derivational suffixes. The Porter stemmer uses a set of existing suffixes (with complex compound suffixes broken into simple ones) and manually defined rules. The implementation of the stemmer for Turkish, Romanian, Armenian, Catalan, Greek, Lithuanian languages is available in [29]. It can be seen that the stemmer has not been implemented for the Kazakh language.

The algorithm consists of five steps. At each step, the inflectional or derivational suffix is stripped and the rest is checked for compliance with the rules (for example, for Russian words, the stem must contain at least one vowel). If the resulting word satisfies the rules, the next step is taken. Otherwise, the algorithm selects another suffix to be stripped. According to the official project website, the maximum inflectional suffix is stripped in the first step, the letter "i" in the second, the derivational suffix in the third, suffixes of superlative forms, "ь" and one of the two "н" in the fourth. The fact that the Porter stemmer does not use any dictionaries and stem bases is advantageous for speed and application range (it copes well with nonexistent words) and at the same time disadvantageous in terms of stemming accuracy. The algorithm often cuts off the word more than necessary, complicating the synthesis of the normal form according to the resulting stem: аманшылы –>ама (the really unchanging part is аман). When implementing the Porter algorithm for the Kazakh language, we give all suffixes and endings of the Kazakh language in the program code:

```
    _re_all=re.compile(
    r»(шалық|шелік|даған|деген|таған|теген|лаған|леген|»
    r»дайын|дейін|тайын|тейін|кент|хана»r»ңдар|
ңдер|дікі|тікі|нікі|атын|етін|йтын|йтін|»
    r»гелі|қалы|келі|ғалы|шама|шеме|»
    r»мын|мін|бын|бін|пын|пін|мыз|міз|быз|біз|пыз|піз|-
сың|сің|»
    r»сыз|сіз|ңыз|ңіз|дан|ден|тан|тен|нан|нен|нда|нде|-
дың|дің|тың|»
    r»тің|ның|нің|дар|дер|тар|тер|лар|лер|бен|пен|мен|стан|»
    r»дай|дей|тай|тей|дық|дік|тық|тік|лық|лік|паз|»
    r»ғыш|гіш|қыш|кіш|шек|шақ|шыл|шіл|нші|ншы|дап|-
деп|»
    r»тап|теп|лап|леп|дас|дес|тас|тес|лас|лес|ғар|гер|қар|-
кер|дыр|»
    r»дір|тыр|тір|ғыз|гіз|қыз|кіз|ған|ген|қан|кен|»
    r»ушы|уші|лай|лей|сын|сін|бақ|бек|пақ|пек|мақ|мек|
йын|йін|йық|йік|»
    r»сы|сі|да|де|та|те|ға|ге|қа|ке|на|не|»
    r»ді|ты|ті|ны|ні|ды|ба|бе|па|пе|ма|ме|»
    r»лы|лі|ғы|гі|қы|кі|ау|еу|ла|ле|ар|ер|»
    r»ып|іп|ша|ше|ші|шы|са|се|»
    r»и|й|ы|і)$»)
```

By stemming 1062058 dump words, the word is assigned to the variable: word="орналасқан" and the stemmer is started. Consequently, there is a process of stripping affixes of words in the database by the stemmer, some examples of the process are given in Table 1.

```
“”””

unittest.main()
“””

stemmer=Stemmer()
word=”орналасқан”
word=stemmer.stem(word)
print(word)
```

Table 1

Process of stripping affixes of words in the database by the stemmer

Original text	Expected version	Stemmer result
Орналасқан	Орналас–қан	Орналас
Деректердің	Дерек+тер+дің	Дерек
Соншалықты	Сонша+лық+ты	Соншалық
Сапалықты	Сапа+лық+ты	Сапалық
Сүңг+и+т+ін	Сүңг+и+т+ін	Сүңгит
Таң+ғы	Таң+ғы	Таң

The resulting dictionary was indexed for further use by the algorithm.

To extract keywords from the text corpus, a domain dictionary approach was used [30]. A dictionary of essential terms that the algorithm will operate with was compiled from the dump file. For this, the topics of about 224,000 articles were used. This approach focuses on the formation of a dictionary, which should be based on terms of the subject area carefully selected by an expert. Some of the topics on personalities, settlements and others that are not suitable as keywords were not included in the dictionary. Thus, the length of the dictionary of essential terms was 50,000 elements.

5. Neural network learning

We know that a dictionary of 1,062,058 words is redundant, which significantly increases machine time for neural network learning.

According to the calculations, a closer dictionary size for efficient work was determined. After highlighting unique word stems, the size of the dictionary was reduced to 135,120 elements.

For the experiment, a perceptron-type neural network was created. The number of perceptron receptors is equal to the length of the word stem dictionary, i. e. 135,120. In the output layer, the number of neurons is equal to the length of the essential term dictionary, i. e. 50,000.

Choosing the number of elements of the hidden layer, it was taken into account that the task of the neural network is to generalize the input data array. Kevin Swingler [31] recommends using a narrowing neural network in such conditions, that is, a network with fewer neurons in the hidden layer than in the input one.

For example, for the limit error e=0.1, a learning sequence 10 times the number of weights must be used. This dependence is described by the formula:

$$n \geq \frac{\omega}{\varepsilon}. \quad (1)$$

According to the formula (1), where the number of training examples (n) is equal to the product of the number of connections (ω) by the error reciprocal (1/ε), as a result of the reduction, the ratio of the number of connections (ω) to the number of errors (ε) is obtained.

Hence, using more connections than the learning set can fill harms the generalizing ability, which was revealed by comparing experimentally constructed “learning curves” (Fig. 1, x – errors in percentage, y – number of experiments), corresponding to the maximum generalizing ability. At the same time, 65,000 neurons were found in the hidden network layer.

For learning, a sample was prepared, where each essential term corresponded to the text vector model (2, 0, 4, ..), where the first element corresponds to the number of occurrences of the first word stem from the dictionary, etc. For convenience, as data for neural network learning, the occurrences were normalized in the range from 0 to 1.

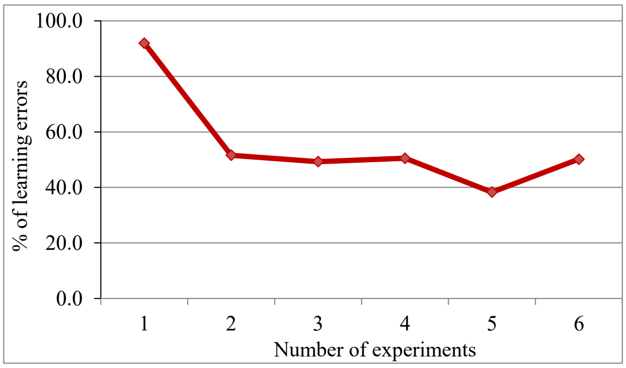

Fig. 1. Learning error size (%) for each experiment

For neural network learning, it is necessary to perform a large number of iterations with the text corpus, which can be implemented using the error backpropagation algorithm. In this case, a learning set of text corpora with pre-known keywords is used, as a result of errors minimization, we reveal the difference between the output values of the neural network and the input ones of keywords. This iterative gradient algorithm is used to minimize the error of the multilayer perceptron and obtain the desired output. Learning the selection of the best options, which occurs by comparing the vector model of the studied text with the threshold values of the search model is carried out. The vector model involves matching each document with the frequency range of words and, accordingly, the vector in the lexical space. In the search process, the frequency portrait of the query is considered as a vector in the same space and the most relevant documents are determined by the degree of proximity (distance or angle between the vectors). In more advanced vector models, the dimension of space is reduced by discarding the most common or rare words, thereby increasing the significance percentage of keywords. Next, the relevance of each keyword in the text corpus is estimated by matching with the value vector. The probabilities of their attributing to the key ones are determined in accordance with the constructed model for approximating the indicator to the expected result.

The difference between the vector values of these parameters for key and non-key words is determined. Next, the probability of attributing each word to the group of keywords is estimated and its threshold is set, i. e., the model is learned.

6. Result of the algorithm for keyword search in the Kazakh language text corpus

By means of the Porter stemmer, the dictionary to search for keywords in the Kazakh language, which includes the base of Kazakh word stems and the terminological dictionary for neural network learning was created. This base will be used to develop a system of semantic text analysis, for remote check of electronic text works of learners.

For example, the following text corpus is chosen.

Text 1. *Нейрожелі тәжірибелік мәліметтерді сақтауға және қолдануға табиғи бейімділігі бар параллельді таратылған процессорлар жиыны. Ол екі жағдайда миға ұқсас:*

1) Білім қоры желіні оқыту үрдісінде қалыптасады.

2) Синаптикалық салмақ ретінде анықталған нейрон аралық бірігу күштері есте сақтау үшін қолданылады.

Салмақ – Жер бетіне жақын тұрған денеге әсер ететін ауырлық күшінің сандық шамасы: P=mg, мұндағы m - дене массасы, g - еркін түсу үдеуі (немесе ауырлық күшінің үдеуі). Дененің массасы тұрақты шама, ал g мәні Жер бетіндегі ендікке және теңіз деңгейінен есептелетін биіктікке байланысты (мысалы, Алматы үшін g=9,804 м/с^2*) өзгеретіндіктен, оған сәйкес дененің салмағы да өзгереді.*

Table 2 gives the word forms (1, 2 columns) from the dictionary of keywords and essential terms (3, 4 columns).

In neural network learning, dictionaries with stems of 135,120 words and the terminological dictionary of 50,000 words were used. The neural network consisted of one hidden layer.

The structure of the neural network is as follows.

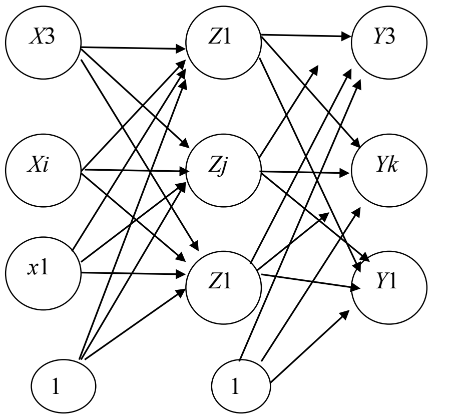

Fig. 2. Error backpropagation neural network with one hidden layer

Here, neurons representing the network outputs (*Y*) and hidden neurons may have a bias (1). These biases serve as weights on the connections emanating from the neurons, at the output of which there is always 1. In addition, the arrows in the figure show the movement of information during the phase of data propagation from inputs to outputs. In the learning process, signals propagate in reverse direction. As mentioned earlier, network learning includes three stages: feeding learning data to the network inputs, error backpropagation, and weight correction. During the first stage, each input neuron (*X*) receives a signal and broadcasts it to each of the hidden neurons (*Z*). Each hidden neuron then calculates the result of its activation function (network function) and sends its signal to all output neurons. Each output neuron (*Y*), in turn, calculates the result of its activation function, which is nothing more than the output signal of this neuron for the corresponding input data. In the course of learning, each neuron at the network output compares the calculated value of *Y* with the teacher-provided *T* (target value), determining the corresponding error value for the given input pattern. Based on this error, *Qk* (*k*=1, 2...) is calculated. *Qk* is used in error propagation from *Y* to all network elements of the previous layer (hidden neurons associated with *Yk*), as well as later when the weights of the connections between the output and hidden neurons change. Similarly, *Qj* (*j*=1, 2...) is calculated for each hidden neuron *Zj*. Despite the fact that error propagation to the input layer is not necessary, *Qj* is used to change the weights of the connections between the hidden layer neurons and input neurons. After all *Q* have been determined, the weights of all connections are adjusted simultaneously.

Table 2

Word forms from the dictionary of keywords and essential terms

No.	word forms	No.	word forms	No.	terms	No.	terms
1	*Нейрожелі*	8	*Тұр*	1	*Нейрожелі*	8	*сандық шама*
2	*тәжірибе*	9	*Жақын*	2	*заңдылық*	9	*бірігу күштері*
3	*мәлімет*	10	*Күші*	3	*ғылым*	10	*әдіс-тәсілдері*
4	*сақта*	11	*Сан*	4	*Ауырлық күш*	11	*дәлелдеу*
5	*қолдан*	12	*байланыс*	5	*Синаптикалық салмақ*	12	*Тұрақты шама*
6	*бейім*	13	*тұрақты*	6	*дене салмағы*	13	*Дене массасы*
7	*дене*	14	*заңдылық*	7	*Білім қоры*	14	*Жер беті*

Thus, after obtaining the numerical data of the weights, the correlation coefficient was calculated, which amounted to 0.99 %. This result showed a linear dependence of input and output data of the neural network, which indicates the likelihood of correspondence between the number of words in the obtained dictionary and the number of words in the Kazakh language dictionaries. Hence, significant deviations in neural network learning are not observed, so, the desired result is achieved. If you compare with the number of words of the “Explanatory dictionary of the Kazakh language” – 106,000, the dictionary is more suitable for the Kazakh language word base.

7. Discussion of the results of research on the development of a keyword search algorithm

After applying the Porter stemmer to search for keywords in the Kazakh text corpus, a dictionary of word stems of 135,120 word forms and a reference dictionary of 50,000 keywords (or terminological dictionary) were obtained.

For parsing the text in the Kazakh language, the Porter algorithm was chosen. Due to this, dictionaries for keyword search were created. In the works on the development of linguistic processors for the Kazakh language, lexical-morphological and morphological text analyses were considered, where subject ontology and the dictionary of suffixes and affixes were used [32, 33]. Hence, the study of semantic analysis of Kazakh texts using a neural network and compiling a keyword dictionary using the Porter algorithm are relevant.

This study is applicable only for text information processing and the algorithms are not applicable for another format of information (images, video, audio), which is one of its drawbacks. Based on this study, a neurocomputer system is developed that will allow semantic text analysis and determining whether the text corresponds to a given topic. The neurocomputer system will include a semantic analyzer of the Kazakh text, which can be used in online courses to check the text works of learners in educational institutions [34].

By filling the system with dictionaries of other natural languages, it can be applied to other languages.

In connection with the transition of the Kazakh language to the Latin alphabet, the adaptability of this study to the Latin alphabet should be studied in the future.

8. Conclusions

1. Hence, with the help of the Porter stemmer, a keyword dictionary with a total of 135,120 word forms and a reference dictionary of essential terms, which includes 50,000 words were created. The given number of word stems and the reference dictionary of keywords are a probabilistic approximation to the number of words in the explanatory dictionary of the Kazakh language, as a result their approximate error is 50–90 %. This is a good result allowing to determine the approximate number of keywords.

2. As a result of learning data preparation, we got a pair: a keyword and a vector of corresponding word forms. As well as a fixed difference in the values of word form vectors for key and non-key words. After that, the probability of assigning each word to the group of key ones is estimated and its threshold is set, that is, the model is learned. Weights were determined taking into account the displacement of neurons in the inner layer of the neural network, while correlation analysis showed a linear dependence of input and output data of the neural network with a value of 0.99.

References

1. Bassiou, N. K., Kotropoulos, C. L. (2014). Online PLSA: Batch Updating Techniques Including Out-of-Vocabulary Words. IEEE Transactions on Neural Networks and Learning Systems, 25 (11), 1953–1966. doi: https://doi.org/10.1109/tnnls.2014.2299806
2. Borschev, V. B., Partee, B. H. (2014). Ontology and Integration of Formal and Lexical Semantics. Proceedings of the international scientific conference on computational linguistics “Dialogue”. Available at: http://www.dialog-21.ru/digests/dialog2014/materials/pdf/BorschevVBParteeBH.pdf
3. Turdakov, D. Y., Astrakhantsev, N. A., Nedumov, Y. R., Sysoev, A. A., Andrianov, I. A., Mayorov, V. D. et. al. (2014). Texterra: A framework for text analysis. Programming and Computer Software, 40 (5), 288–295. doi: https://doi.org/10.1134/s0361768814050090
4. Attali, Y., Burstein, J. (2006). Automated Essay Scoring With E-rater® V.2. Journal of Technology, Learning, and Assessment, 4 (3). Available at: https://ejournals.bc.edu/index.php/jtla/article/view/1650/1492
5. Dikli, S. (2006). Automated Essay Scoring. Turkish Online Journal of Distance Education, 7 (1), 49–62. Available at: https://www.researchgate.net/publication/26415982_Automated_Essay_Scoring
6. Rai, A., Kannan, R. J. (2018). Differed Restructuring of Neural Connectome Using Evolutionary Neurodynamic Algorithm for Improved M2M Online Learning. Procedia Computer Science, 133, 298–305. doi: https://doi.org/10.1016/j.procs.2018.07.037
7. Chen, Z., Huang, Y., Liang, Y., Wang, Y., Fu, X., Fu, K. (2017). RGloVe: An Improved Approach of Global Vectors for Distributional Entity Relation Representation. Algorithms, 10 (2), 42. doi: https://doi.org/10.3390/a10020042
8. Sukumar A., R., Sukumar A., S., Shah A., F., Anto P., B. (2010). Key-Word Based Query Recognition in a Speech Corpus by Using Artificial Neural Networks. 2010 2nd International Conference on Computational Intelligence, Communication Systems and Networks. doi: https://doi.org/10.1109/cicsyn.2010.56
9. Lytvyn, V., Moroz, O. (2013). Contextual search method based on the thesaurus of knowledge domain. Eastern-European Journal of Enterprise Technologies, 6 (2 (66)), 22–27. Available at: http://journals.uran.ua/eejet/article/view/18700/17065
10. Ranjan, N. M., Prasad, R. S. (2018). LFNN: Lion fuzzy neural network-based evolutionary model for text classification using context and sense based features. Applied Soft Computing, 71, 994–1008. doi: https://doi.org/10.1016/j.asoc.2018.07.016
11. Zhang, H., Jun, Y. (2009). An Algorithm of Text Automatic Proofreading Based on Chinese Word Segmentation. 2009 International Conference on Computational Intelligence and Software Engineering. doi: https://doi.org/10.1109/cise.2009.5364024
12. Kalinichenko, L. A. (2012). Effective support of databases with ontological dependencies: Relational languages instead of description logics. Programming and Computer Software, 38 (6), 315–326. doi: https://doi.org/10.1134/s0361768812060059
13. Garanina, N. O., Sidorova, E. A. (2015). Ontology population as algebraic information system processing based on multi-agent natural language text analysis algorithms. Programming and Computer Software, 41 (3), 140–148. doi: https://doi.org/10.1134/s0361768815030044
14. Bessmertny, I. A. (2010). Knowledge visualization based on semantic networks. Programming and Computer Software, 36 (4), 197–204. doi: https://doi.org/10.1134/s036176881004002x
15. Jorge-Botana, G., León, J. A., Olmos, R., Escudero, I. (2010). Latent Semantic Analysis Parameters for Essay Evaluation using Small-Scale Corpora*. Journal of Quantitative Linguistics, 17 (1), 1–29. doi: https://doi.org/10.1080/09296170903395890
16. Mashechkin, I. V., Petrovskiy, M. I., Popov, D. S., Tsarev, D. V. (2011). Automatic text summarization using latent semantic analysis. Programming and Computer Software, 37 (6), 299–305. doi: https://doi.org/10.1134/s0361768811060041

17. Grigoryeva, E., Klyachin, V., Pomelnikov, Y., Popov, V. (2017). Algorithm of Key Words Search Based on Graph Model of Linguistic Corpus. Vestnik Volgogradskogo Gosudarstvennogo Universiteta. Serija 2. Jazykoznanije, 16 (2), 58–67. doi: https://doi.org/10.15688/jvolsu2.2017.2.6
18. Hu, J., Li, S., Yao, Y., Yu, L., Yang, G., Hu, J. (2018). Patent Keyword Extraction Algorithm Based on Distributed Representation for Patent Classification. Entropy, 20 (2), 104. doi: https://doi.org/10.3390/e20020104
19. Kanagarajan, K., Arumugam, S. (2018). Intelligent sentence retrieval using semantic word based answer generation algorithm with cuckoo search optimization. Cluster Computing. doi: https://doi.org/10.1007/s10586-018-2054-x
20. Turney, P. D. (2000). Learning Algorithms for Keyphrase Extraction. Information Retrieval, 2 (4), 303–304. doi: https://doi.org/10.1023/A:1009976227802
21. Kulhare, S. (2017). Deep Learning for Semantic Video Understanding. A Thesis for the Degree of Master of Science in Computer Engineering. Rochester. Available at: https://pdfs.semanticscholar.org/d195/9ba4637739dcc6cc6995e10fd41fd6604713.pdf
22. Ibrahim, A. S. (2017). End-To-End Text Detection Using Deep Learning. Blacksburg. Available at: https://vtechworks.lib.vt.edu/handle/10919/81277
23. Lin, X. V., Wang, C., Zettlemoyer, L., Ernst, M. D. (2018). NL2Bash: A Corpus and Semantic Parser for Natural Language Interface to the Linux Operating System. International Conference on Language Resources and Evaluation. Available at: https://homes.cs.washington.edu/~mernst/pubs/nl2bash-corpus-lrec2018.pdf
24. Dictionary Based Annotation at Scale with Spark, SolrTextTagger and OpenNLP. Available at: https://databricks.com/session/dictionary-based-annotation-at-scale-with-spark-solrtexttagger-and-opennlp
25. Bingel, J., Bjerva, J. (2018). Cross-lingual complex word identification with multitask learning. Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications. doi: https://doi.org/10.18653/v1/w18-0518
26. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., et. al. (2014). Caffe. Proceedings of the ACM International Conference on Multimedia - MM '14. doi: https://doi.org/10.1145/2647868.2654889
27. Hinton, G. E., Osindero, S., Teh, Y.-W. (2006). A Fast Learning Algorithm for Deep Belief Nets. Neural Computation, 18 (7), 1527–1554. doi: https://doi.org/10.1162/neco.2006.18.7.1527
28. Snowball. Available at: https://snowballstem.org/
29. He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi: https://doi.org/10.1109/cvpr.2016.90
30. Swingler, K. Applying Neural Networks. A practical Guide. Available at: http://matlab.exponenta.ru/neuralnetwork/book4/3_2.php
31. Sharipbaev, A. A., Bekmanova, G. T., Ergesh, B. J., Buribaeva, A. K., Karabalaeva, M. H. (2012). The intellectual morphological analyzer based on semantic networks. Open Semantic Technologies for Intelligent Systems.
32. Koybagarov, K. Ch., Musabaev, R. R., Kalimoldaev, M. N. (2014). Razrabotka lingvisticheskogo protsessora tekstov na kazahskom yazyke. Problemy informatiki, 3, 64–72.
33. Akanova, A., Ospanova, N., Abildinova, G., Ulman, M. (2016). Assessment tools for evaluating knowledge of online students. Proceedings of the 13th International Conference Efficiency and Responsibility in Education 2016, 9–18. Available at: https://erie.v2.czu.cz/en/r-13629-proceedings-2016