Method of Multi-purpose Text Analysis Based on a Combination of Knowledge Bases for Intelligent Chatbot

Andrii Yarovyi and Dmytro Kudriavtsev

Vinnytsia national technical university, Khmelnytsky highway 95, Vinnytsia, 21021, Ukraine

Abstract

The issues related to the text message recognition for intelligent chat-bot information system were analyzed. The issue of multi-purposed text message recognition was noted and resolved. Opportunities for use multiple knowledge bases were presented. Experiments related to use combinations of knowledge bases of different subject areas were completed and shown. Comparison table of effectively usage with analogs was presented.

Keywords 1

Text processing, semantic text analysis, chatbot, terminological knowledge base, intelligent systems.

1. Introduction

Chatbot – is an intelligent information system (IIS), mainly in the form of a program with the ability to process incoming data and provide on its basis the necessary information. The main area of application is support and assistance of the users of mainly one or more selected subject areas. Ability to use elements of artificial intelligence has led to the rapid development of algorithms for processing information in chatbot systems. The processing of the input data in the IIS data is usually implemented in the form of specially designed algorithms, machine learning technology, neural networks, deep learning [1]. Providing user support during an IIS session, the chatbot may have certain usage restrictions that arise when installing the chat messenger in a more global IIS. This chatbots requires reliable and highly skilled processing of input data. Providing this data processing is the provision of data mining tools, data processing algorithms (Big Data), and monitoring of the processing process to improve analysis and reduce the error [2]. Applying all the above-mentioned facilities and functions, most modern chatbots have a high functional potential and recognize up to 99% of incoming information in preselected subject areas. If you consider the chatbot as a fully autonomous intelligent information system, the problem is to apply for a wide range of tasks and compatibility with third-party software.

When choosing the type of chatbot for a more global IIS, the focus is on the subject area in which should be understood chatbot. Accordingly, the degree of understanding and complexity of the implementation of the data storage is directly proportional to the terminology base of the subject area. When combining several subject areas or choosing an interdisciplinary domain, the degree of understanding falls significantly and increases the complexity of maintaining the relevance of the data storage chatbot. The use of several repositories simultaneously or in parallel is a rather promising solution to this problem, since it ensures the division of the terminology database into smaller aggregates, which is easier to accompany and provide input requests to the user of the IIS [3]. But despite the benefits, the key disadvantage of this method is to increase the complexity of processing and providing information to the user due to possible conflicts between data storages. When choosing a stand-alone IIS chatbot, the impact on its characteristics is largely independent of other IIS's and the

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: a.yarovyy@vntu.edu.ua (A. Yarovyi); dmytro.kudriavtsev@vntu.edu.ua (D. Kudriavtsev);
ORCID: 0000-0002-6668-2425 (A. Yarovyi); 0000-0001-7116-7869 (D. Kudriavtsev);

ability to automatically update the repository with relevant data. Based on these features, the processing of such an IIS requires detailed analysis and the use of reliable tools and data processing tools [4].

2. Theoretical preparation

As for the functionality of the chatbot, they may include support for the user of the IIS, the capabilities of third-party services used by the chatbot, as well as the intellectual analysis of the user's actions. User support is the exchange of information packets between the user and the IIS in the form of text, image, audio, and video messages.

Each of these types of packets of information must have its own device for analysis of incoming data of the IIS and be independent on the analysis of other types of data in the created IIS. In accordance with the minimum requirements for the functioning of the IIS chatbot, the system must adequately and timely recognize the type of information packet coming from the user.

Another important criterion for chatting is the information environment in which the IIS chatbot works. By this criterion, they are divided into open, which involves the use of IIS by several users at the same time, and closed, which can be used by only one user, regardless of the duration of the session.

The possibility of simultaneous operation of the IIS at once with several users is at the same time the most difficult stage of implementation of the IIS chatbot. It usually uses parallel computing and distributed systems technologies. The session format for this type of chatbot is missing. Fixing the duration of the user with the IIS is usually analyzed by the duration of the breaks between the presentation of input information by the user. By monitoring the processes of information exchange, reports and metadata are formed. Analyzing these reports, it is possible to conclude on the level of processing of information, uncertain situations, when the most relevant information in the data storage IIS does not exceed the permissible threshold and other forms of work IIS chatbot.

Uncertain situations arise in case of insufficient storage space or its low informational value, as well as in cases where the type of incoming information is incompatible with the possible identification of the selected IIS chatbot. But the most frequent problem of occurrence of uncertain situations is the essential difference of the input information from the terminology database of the chosen subject area or areas. Observing these cases, one can conclude that the implementation of additional functions is necessary.

Also, the structure of stored data for each subject area must be the same for fast search with a minimum of the getting data logic changes. Formatting data needs to be simple and clearly understandable. The most preferable structures are the hash table and key-value dictionary. Implementing part is related to end-user development. Pay attention to the size of each table in the terminology base. In the experiment section will be presented several examples of terminology knowledge bases for different subject areas.

When using several terminological databases, optionally related subject areas, there is a need to enter the threshold of sensitivity. After all, when using multiple data stores at the same time to select the best result, each repository will provide its own best answer with the corresponding recognition factor. In this case, the data storage response that is not related to the subject of the session should be rejected as inappropriate. It is for this purpose that this threshold is introduced, which aims to reject such answers and to avoid such situations in general. The results of adding the threshold of sensitivity are presented in table 1.

For input data were used several data sources with text sentences, average size of each sentence is equal to 15-17 words. Number of sentences for this comparison is equal to 18536. Total number of keywords for this comparison is equal to 62737. Correctness calculated as

$$\begin{gathered} C = 2 - \frac{(Keywords - Phantom)}{Total\ keywords}, \text{ if } \frac{(Keywords - Phantom)}{Total\ keywords} > 1 \text{ else} \\ C = \frac{(Keywords - Phantom)}{Total\ keywords}, \text{ if } \frac{(Keywords - Phantom)}{Total\ keywords} < 1 \end{gathered} \quad (1)$$

Table 1
Influence of adding threshold

Threshold, %	Keywords found, N	Phantom found, N	Correctness, %
5	120528	8920	22,101
10	94877	3105	53,719
15	81059	1007	72,401
20	**56343**	**259**	**89,396**
25	53952	227	85,635
35	49013	193	77,817

As presented on the Table 1, the best threshold of sensitive is near 20%. Phantom keywords are attending to the words group which is not belonging to the semantic kernel but accepting by standard semantic analysis algorithm as keyword. Semantic kernel in this case calculated by improved semantic text analysis [4]. If we will use threshold below the 20 percent, the threshold value of frequency for semantic text analysis needs to not depends on the threshold of sensitive. In this case correctness is too small to be selected.

When we tried to use the terminological knowledge bases of the two subject areas instead of the one, the accuracy of recognition from the user was slightly reduced by an average of 2-3% with a threshold value of 20% for correspondence of information. In the last research, three terminological knowledge bases were also used and were found situations when information belonged to both terminological knowledge bases, which in turn caused the uncertainty of the topic of dialogue [4]. In this case, the accuracy of recognition of textual information of the user decreased by 8-10% in the worst case, when the subject areas had significant similarity in terminological knowledge bases. The datasets used in this research were taken from the Kaggle informational resource [5-7]. Due to the decrease in recognition accuracy, there was a problem checking the quality of the terminological knowledge base and its similarity to others used in chat-bot IIS. Additionally, was found that the total processing time of input information requires detailed analysis and optimization if more than two terminological knowledge bases are used.

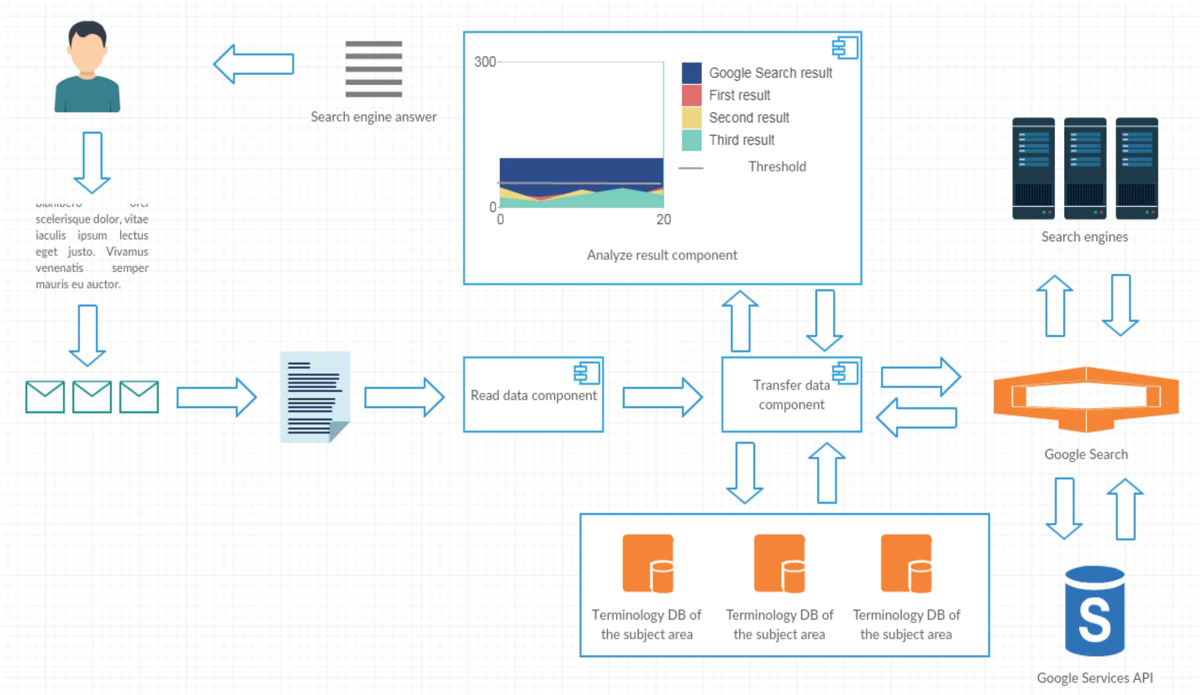


Figure 1: Model of a user's session in the chat-bot IIS

The main purpose of this work is proceeding the research on the use of multiple subject areas using a recurrent neural network. As well as resolving found issues and compare results of using different methods of text analysis.

The high-level structure of the chatbot as a prototype of multi-purpose text analysis is presented below on Figure 2.

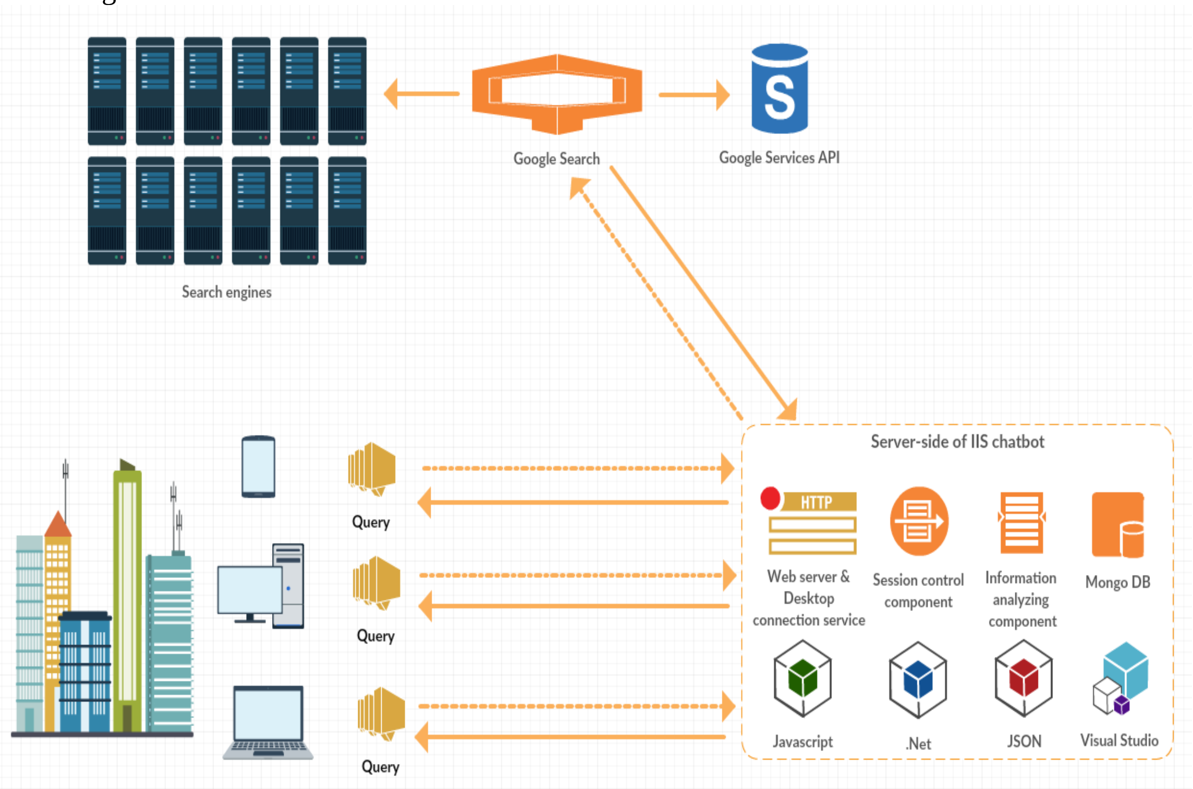

Search engines
Google Search
Google Services API
Query
Query
Query
Server-side of IIS chatbot
HTTP
Web server & Desktop connection service
Session control component
Information analyzing component
Mongo DB
Javascript
.Net
JSON
Visual Studio

Figure 2: Architecture model of IIS chatbot

3. Main research

Comparing the development of natural language processing with the development of artificial intelligence, it is worth emphasizing the affinity of the directions and the direct application of artificial intelligence for the semantic parsing of text information from the flow for the allocation of important elements (constructions, terms). The selection of intellectually valuable elements from the information stream is formed by searching among the information of structures already identified by the repository as constructs which belonging to the terminology database of the chosen subject area or regions. The described search is carried out by applying the intellectual means of information analysis. For IIS chatbot such a tool was chosen neural network for the task of intellectual analysis of natural language. For the problem of intellectual analysis of the text, two most common types of the neural network were considered. The first type is a machine learning algorithms, the feature of which is the minimum pre-processing of the data before use [8]. The disadvantage of this type is the complexity of learning and minor effectiveness if the source data will be changed in the future. The second type is recurrent neural networks, which also belong to the class of deep neural networks and whose tasks include recognition of natural text and speech recognition. Unlike machine learning algorithms, the connection between nodes in recurrent neural networks forms an oriented cycle [8, 9]. This creates the internal state of the network, which allows it to manifest a dynamic behavior in time, based on which internal memory is formed. Due to the effect of internal memory, the recurrent neural network is resistant to the dynamic change of the input stream of information and the processing of arbitrary input sequences [10-12]. Focusing on the problem of natural language recognition, the recurrent neural network was selected.

The recurrent neural network (RNN) has undergone many modifications and methods for a period from its creation to the present, leading to a complete set of neural network variants for each of the tasks. The most well-known RNNs are the Elman, Jordan networks, the echo-state network, the network using the long-term LSTM method, the two-directional RNN, RNN of continuous time, the Hierarchical RNN, the second order RNN, the RNN of several time scales, the Turing neural machines. Among the presented modifications of RNN, the greatest attention was focused on the use of the method of long-term short-term memory, which best proved itself in the tasks of recognition of natural language. This method was developed and published in 1997 by Hochreiter and Schmidguber [13]. This method avoids the problem of gradient disappearance and prevents the disappearance and occurrence of reversible errors. This is due to the reciprocal propagation through an unlimited number of layers deployed in the space of the RNN. Based on the reciprocal distribution through an arbitrary number of layers, RNN is using a long-term memory method to withstand time gaps and can process text information of any length. Thanks to the capabilities of such an RNN, its widest use was in the field of natural language recognition, where this model of RNN began to outperform traditional recognition models. Examples of its effectiveness are the use of the Baidu search giant since 2014, the use of Google Android and Google Voice Search [14, 15]. Thanks to the power of this RNN, it is successfully used to recognize context-sensitive languages, to model languages and multilingual language processing.

Considering the model of RNN using the LSTM method as a modification of the usual RNN, the LSTM elements serve as nodes, representing the simplest RNN with the possibility of reciprocal distribution.

The main advantages of using LSTM method for RNN.

- Withstand time gaps
- Not depends on the text length
- Simple structure

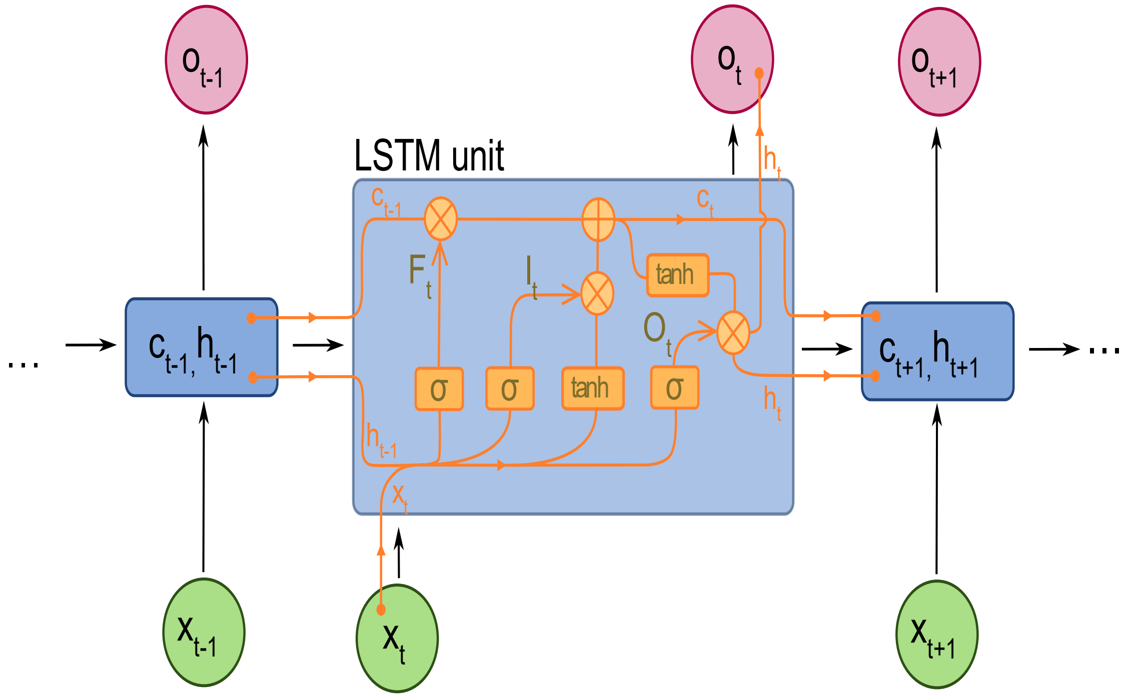


Figure 3: Model of a LSTM cell in RNN [16]

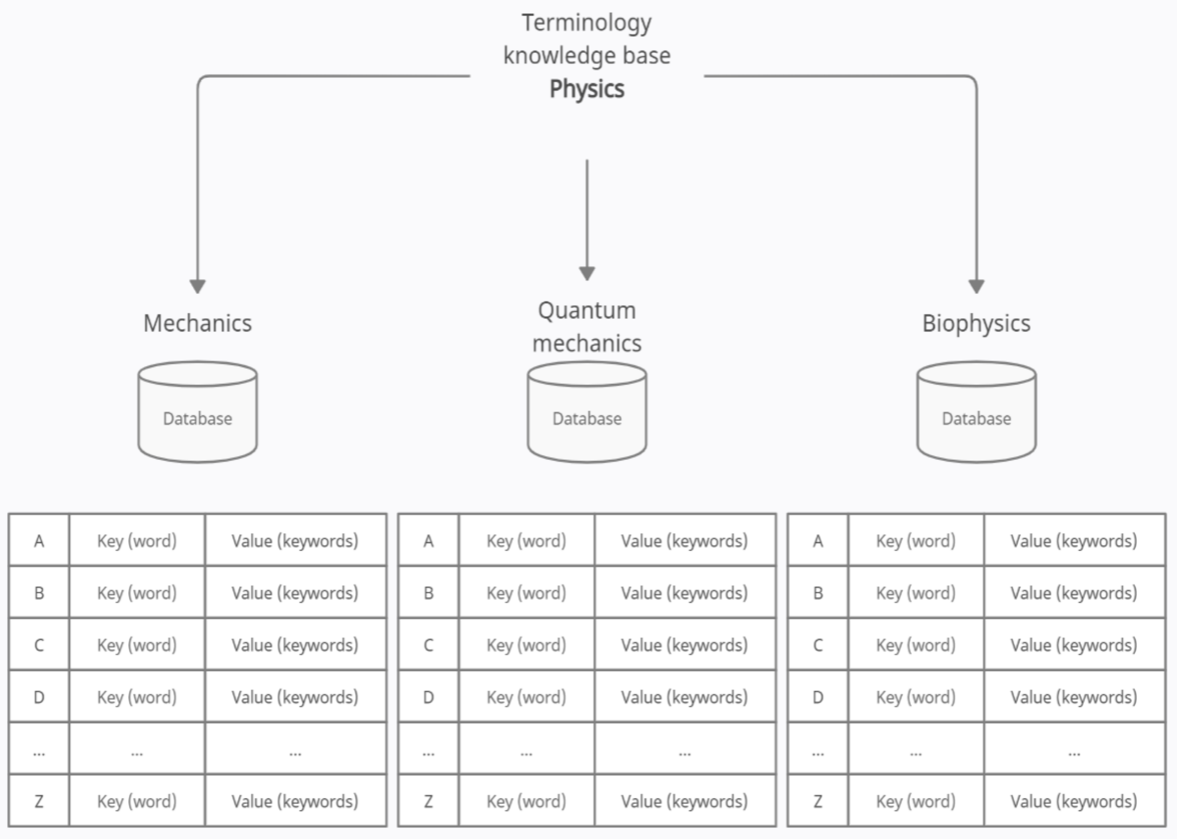
By the experiment method for IIS chatbot, this RNN was selected with a set of characteristics: number of entrance inputs is equal to the size of words vector for analysis (inputs of the input layer), outputs number is equals to the number of used terminology databases (last layer neurons), 3 hidden layers of 1024 neurons in each. The activation function of the neuron is a sigmoid and number of epochs is individual for each experiment, but as you will see in the experiments section, the average epochs for training is about 16-18 thousand.

4. Experiments

After performing theoretical calculations, considered the results of practical implementation in the form of graphs and tables for different input data. For a detailed analysis of the results, more than 30 experiments were conducted on combined data from 5 subject areas and more than 20 terminological databases. Each of the terminological databases is presented in the figures below. A graph of the dependence of processing time on the number of subject areas is given in Figure 1. Part of the experiments is shown in sections 4.2-4.4. Each experiment contains few screens with diagrams and table of input data and results.

4.1. Terminological knowledge base structure

The data organization are presented as key-value dictionary with sub-levels of value which store in shared distributed and document-oriented database – MongoDB [17]. The performance of this database is not included as valuable parameter of research experiments. For reproduce the results you may use another database. Each terminology knowledge base includes up to five database which differ by the impact of relations of the subject area. Each database includes 26 collections of words (1 collection for each letter of English alphabet). The model of this storing is presented on the Figure 4.



A	Key (word)	Value (keywords)	A	Key (word)	Value (keywords)	A	Key (word)	Value (keywords)
B	Key (word)	Value (keywords)	B	Key (word)	Value (keywords)	B	Key (word)	Value (keywords)
C	Key (word)	Value (keywords)	C	Key (word)	Value (keywords)	C	Key (word)	Value (keywords)
D	Key (word)	Value (keywords)	D	Key (word)	Value (keywords)	D	Key (word)	Value (keywords)
...	...	...	...	...	...	...	...	...
Z	Key (word)	Value (keywords)	Z	Key (word)	Value (keywords)	Z	Key (word)	Value (keywords)

Figure 4: Example of storage the terminology knowledge base

In this structure you can fast search needed keywords for selected words and create a tree structure solution for the whole searching text with all crossed links between keywords. The value of the analysis is increasing with every newfound relation between keywords of the parts of sentences. Duplication in keywords will be avoided by organizing the words in the keyword set in the small groups (up to ten related words, in order of decreasing the compatibility).

4.2. Experiment (two knowledge bases)

For this experiment was used two knowledge bases: cars and cooking. Each of knowledge base contains two databases with up to 80000 terms for both knowledge bases. Number of epochs for training is 16396. Detailed information is presented on the Table 2.

Table 2
Results of text analysis for two subject areas

Title	Average found times per text	Average processing time, s	Terms, N	Correctness, %	Total processed texts, N
Buses (Car)	0,193	0,487	11034	78,3	40000
Cars (Car)	0,375	0,450	23006	82,0	40000
Motorcycle (Car)	0,036	0,503	5974	74,9	40000
Baking (Cooking)	0,241	0,488	22406	71,4	40000
Salat (Cooking)	0,156	0,421	3244	81,2	40000
Vegetables (Cooking)	0,289	0,447	15779	93,2	40000

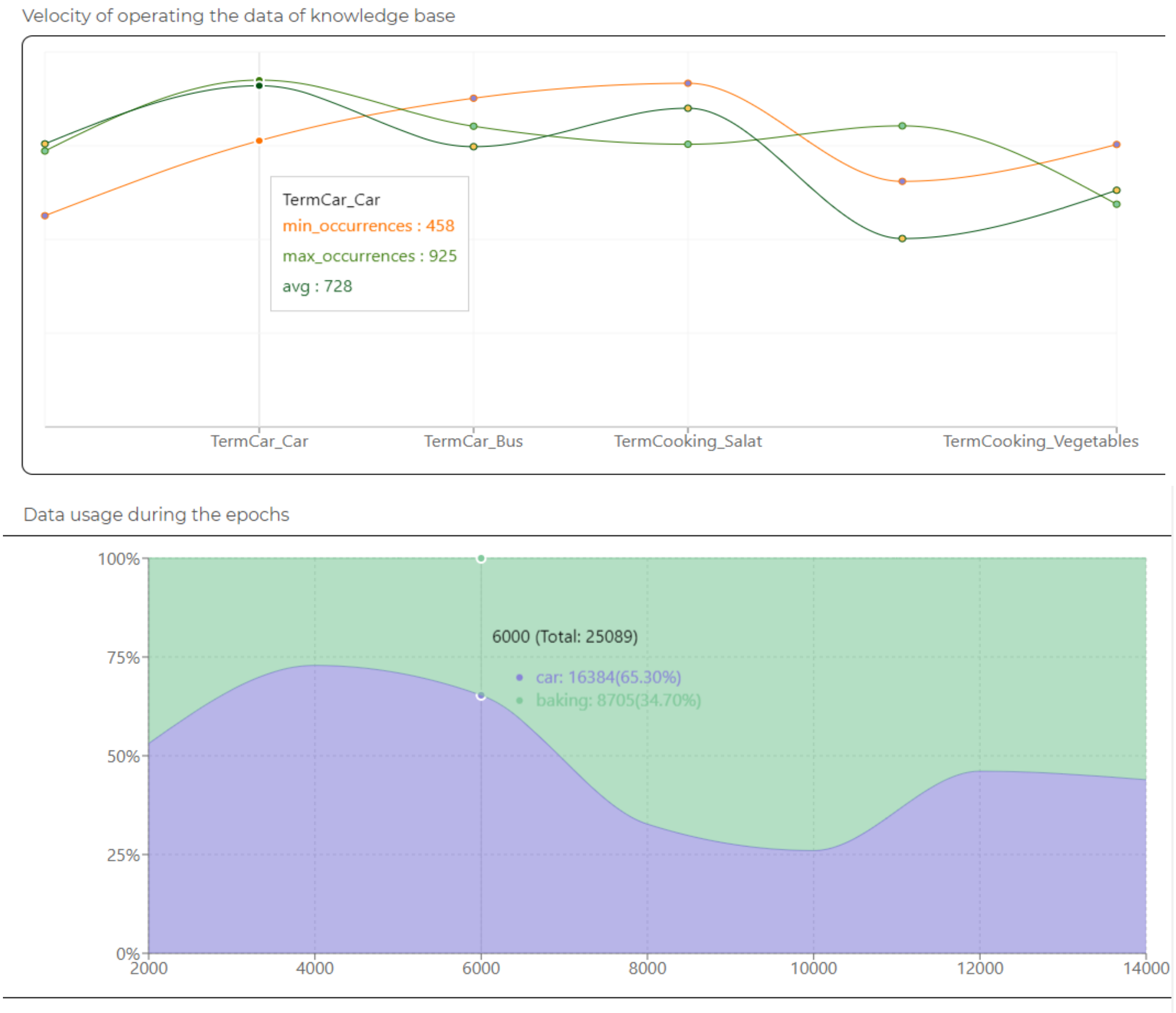


Figure 5: Diagrams of training the RNN with LSTM

4.3. Experiment (three knowledge bases)

For this experiment was used three knowledge bases: cars, sports, and business. Each of knowledge base contains three databases with up to 130000 terms for all knowledge bases. Number of epochs for training is 17992. Detailed information is presented on the Table 3.

Table 3
Results of text analysis for three subject areas

Title	Average found times per text	Average processing time, s	Terms, N	Correctness, %	Total processed texts, N
Buses (Car)	0,032	0,604	11034	76,1	100000
Cars (Car)	0,194	0,725	23006	79,5	100000
Motorcycle (Car)	0,015	0,536	5974	74,2	100000
Tennis (Sports)	0,083	0,703	4306	80,3	100000
Cybersport (Sports)	0,252	0,596	31053	79,6	100000
Biathlon (Sports)	0,099	0,54	6210	76,3	100000
Documents (Business)	0,288	0,782	27083	82,2	100000
Laws (Business)	0,221	0,746	14007	78,7	100000
Government (Business)	0,186	0,658	5384	85,8	100000

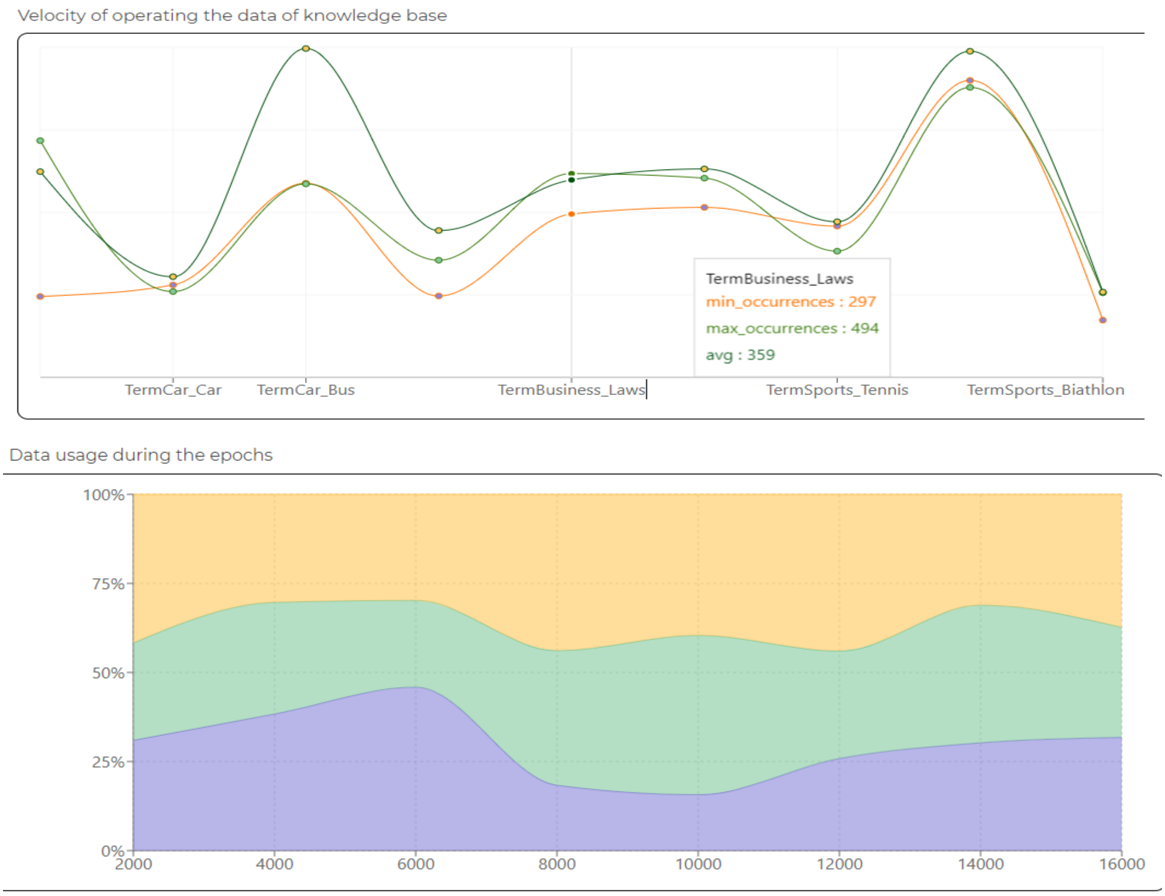


Figure 6: Diagrams of training the RNN with LSTM

4.4. Experiment (four knowledge bases)

For this experiment was used four knowledge bases: cars, history, banking, and business. Each of knowledge base contains three databases with up to 215000 terms for all knowledge bases. Number of epochs for training is 18463. Detailed information is presented on the Table 4.

Table 4
Results of text analysis for four subject areas

Title	Average found times per text	Average processing time, s	Terms, N	Correctness, %	Total processed texts, N
Buses (Car)	0,057	0,735	11034	76,1	150000
Cars (Car)	0,152	0,791	23006	79,5	150000
Motorcycle (Car)	0,013	0,701	5974	78,6	150000
Rome (History)	0,183	0,853	25796	85,2	150000
Austria (History)	0,297	0,820	16402	79,3	150000
Germany (History)	0,268	0,874	38671	84,7	150000
Banks (Banking)	0,186	0,815	12057	81,4	150000
Credit (Banking)	0,239	0,916	18539	77,8	150000
Transactions (Banking)	0,106	0,975	18663	79,1	150000
Documents (Business)	0,183	0,862	27083	81,6	150000
Laws (Business)	0,174	0,819	14007	69,2	150000
Government (Business)	0,153	0,825	5384	82,1	150000

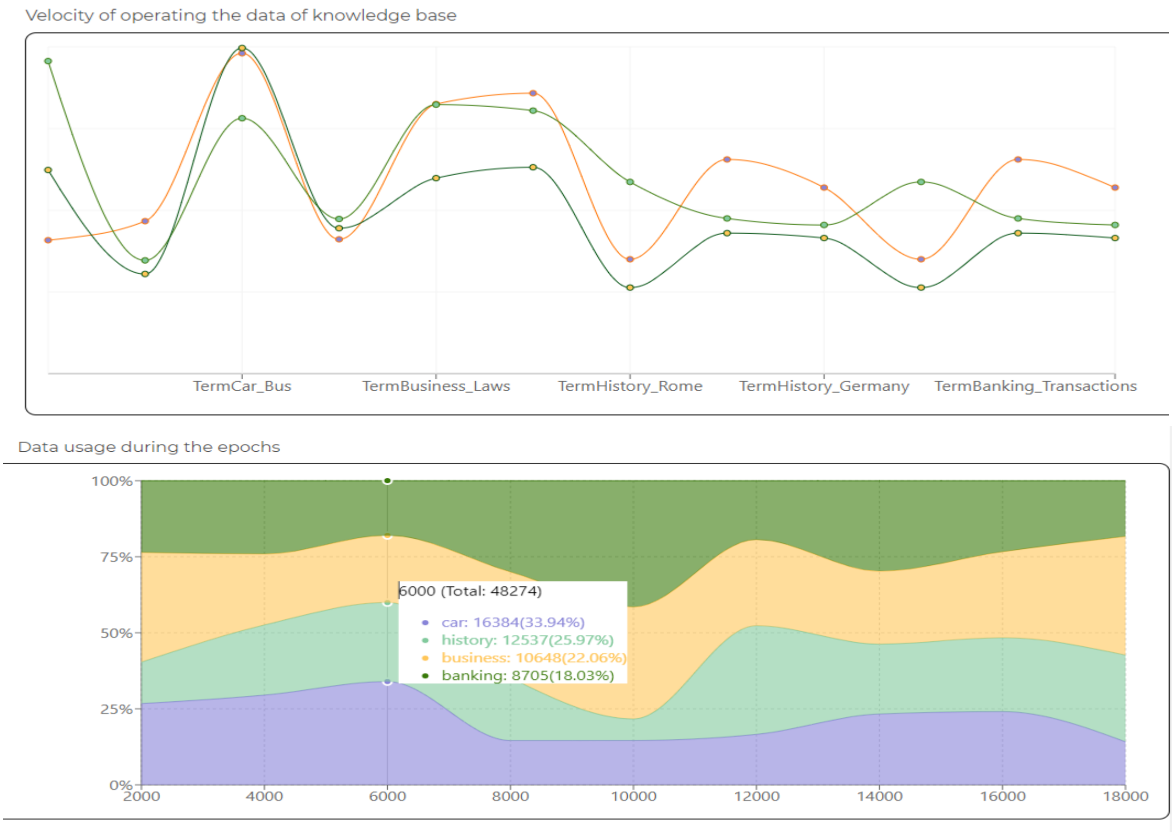


Figure 7: Diagrams of training the RNN with LSTM

5. Comparison results

The experiments of using different combinations of subject areas and sizes of data are successfully presented that the main goal of multi-purposed text analysis was reached and for now the comparison table with similar methods of text analysis is presented below on Table 5.

Table 5
Comparison of different text analysis methods

Title	Average processing time, s	Found keywords, N	Found phantoms, N	Correctness, %	Tests, N
Semantic text analysis	0,602	381064	50973	74,22	150000
Word2vec [18]	0,449	389082	29407	83,43	150000
Multi-purpose text analysis (two subject areas)	0,376	393176	17704	**85,37**	150000
Multi-purpose text analysis (three subject areas)	0,481	395362	14056	**86,52**	150000
Multi-purpose text analysis (four subject areas)	0,619	395163	12953	**89,03**	150000

6. Conclusion

During the research, the main goal of which is to create a method for multi-purpose text analysis was reached. Text analysis for different subject areas can be used for checking the spam messages, for checking the posts on the social network to find necessary information. The main advantages of using the multi-purposed text analysis method find the keywords which can be different by subject areas, the ability to scan user preference in the marketing sphere, making an analysis of the sense of the entered text, and provide a better semantic text analysis. In experiments section was shown that increasing the subject areas count creates an issue relating to the performance of the search, and the best solution is to use up to four subject areas. The correctness level is better than in popular Word2vec method on 2-6 percent.

For the next research, we will make more experiments with subject areas and try to find the relation between different groups of subject areas.

7. References

[1] Serban I. V. et al. “A deep reinforcement learning chatbot” arXiv preprint arXiv:1709.02349 (2017)

[2] Werder K., Heckmann C. S. Ambidexterity in Information Systems Research: Overview of Conceptualizations, Antecedents, and Outcomes, Journal of Information Technology Theory and Application. (2019) T. 20. №. 1.

[3] Pearlson K. E., Saunders C. S., Galletta D. F. Managing and using information systems: A strategic approach. John Wiley & Sons, 2019.

[4] Andrii Yarovyi, Dmytro Kudriavtsev, Serhii Baraban, Volodymyr Ozeranskyi, Liudmyla Krylyk, Andrzej Smolarz, and Gayni Karnakova "Information technology in creating intelligent chatbots", Proc. SPIE 11176, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2019, 1117627 (6 November 2019); https://doi.org/10.1117/12.2537415.

[5] Bhat G., “Chatbot Data” dataset, 2018, URL: https://www.kaggle.com/fungusamongus/chatbot-data

[6] Liling Tan, “Old Newspapers” dataset, 2018, URL: https://www.kaggle.com/alvations/old-newspapers

[7] Jeet J., “US Financial News Articles” dataset, 2018, URL: https://www.kaggle.com/jeet2016/us-financial-news-articles

[8] Polhul, T., & Yarovyi, A. Development of a method for fraud detection in heterogeneous data during installation of mobile applications. Eastern-European Journal of Enterprise Technologies, 2019, T.1(2), 65–75. https://doi.org/10.15587/1729-4061.2019.155060

[9] Guthrie D. Unsupervised detection of anomalous text, University of Sheffield, 2008.

[10] The multiple dimensions of information quality, 2017, URL: https://www.researchgate.net/publication/ 242929284_The_Multiple_Dimensions_of _Information_Quality

[11] Andrii Yarovyi, Raisa Ilchenko, Ihor Arseniuk, Yevhene Shemet, Andrzej Kotyra, Saule Smailova, "An intelligent system of neural networking recognition of multicolor spot images of laser beam profile," Proc. SPIE 10808, Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2018, 108081B (1 October 2018), URL: https://doi.org/10.1117/12.2501691;

[12] A. Yarovii, D. Kudriavtsev and O. Prozor, "Improving the Accuracy of Text Message Recognition with an Intelligent Chatbot Information System," 2020 IEEE 15th International Conference on Computer Sciences and Information Technologies (CSIT), Zbarazh, Ukraine, 2020, pp. 76-79, doi: 10.1109/CSIT49958.2020.9322036.

[13] S. Hochreiter and J. Schmidhuber, Recurrent Neural Networks and LSTM, 1997, URL: https://www.researchgate.net/publication/13853244_Long_Short-term_Memory

[14] Google AI Blog, Neural Networks behind Google Voice, 2015, URL: https://ai.googleblog.com/2015/08/the-neural-networks-behind-google-voice.html

[15] M. Patwary, S. Narang, E. Undersander, J. Hestness, and G. Diamos, Neural Networks in Baidu search engine, 2018, URL: http://research.baidu.com/Blog/index-view?id=103

[16] O. Vinyals, G. Corrado and J. Shlens, Understanding LSTM Networks, 2015, URL: https://colah.github.io/posts/2015-08-Understanding-LSTMs/

[17] Mongo DB Official website, Mongo DB Features and documentation, URL: https://docs.mongodb.com/

[18] V. Bhanawat, Word2vec algorithm, 2019, URL: https://medium.com/@vishwasbhanawat/the-architecture-of-word2vec-78659ceb6638