Development of a System for Finding Extremist Texts

Vadim Moshkin [1], Dmitry Fadeev [1] and Nadezhda Yarushkina [1]

[1] *Ulyanovsk State Technical university, Severny venets street, Ulyanovsk, 432027, Russia*

Abstract

The paper describes an algorithm for finding extremist texts in social networks by classifying them. The algorithm includes 3 stages: preprocessing, filtering using dictionaries and text classification based on the Bayesian algorithm. The architecture of the information system and the brief results of the experiments are described.

Keywords

text analysis, social network, Bayesian classifier, classification

1. Introduction

Currently, there are more cases of bringing social media users to criminal and administrative responsibility for published (post) and disseminated (repost) information that is extremist or terrorism-related.

The relevance of this work is due to the tightening of Russian legislation in the field of information dissemination on the Internet and social networks. At the same time, the user of a social network often does not have the correct idea that his text message may be illegal from the point of view of legislation on combating extremism, terrorism and other offenses.

In this regard, the task of developing intelligent algorithms for automatically checking a user's text messages on a social network for the presence of materials prohibited for publication (in particular, extremist) is urgent.

2. Relevance and algorithms used

The definition of a dangerous type of news on the Internet by topics and risks is applied using interdisciplinary tools (psycholinguistics, computer science, psychology, cognitive science, political science, sociology, cultural studies, journalism) in studies on similar topics [1, 2].

Many works have compared classical mathematical classification models for defining and classifying texts with the tonality of calling for extremism. For example, the study [3] compares the effectiveness of the following algorithms:

- Naive Bayesian classifier;
- Support Vector Machine;
- Decision tree;
- Random forest.

To solve the problem of detecting dangerous posts in this study, a new multilingual vocabulary-lexicon was first created, which consists of extreme lexicons of different levels:

- moderate,
- neutral,

Russian Advances in Fuzzy Systems and Soft Computing: Selected Contributions to the 10th International Conference «Integrated Models and Soft Computing in Artificial Intelligence» (IMSC-2021), May 17–20, 2021, Kolomna, Russian Federation

EMAIL:
ORCID:

- low extreme
- high extreme.

The collected multilingual data are categorized into different classes of extremism using the multilingual extreme vocabulary.

The classification is performed using various supervised and unsupervised algorithms, and it is concluded that supervised algorithms perform better than unsupervised algorithms. In supervised algorithms, the linear support vector classifier showed the highest accuracy of 82%. In unsupervised algorithms, the classification achieves an accuracy of 26%.

The solution described in study [4] is presented as a sentiment analysis based on a naive Bayesian classifier that can help in language learning.

In the study [5], an experimental dataset in Russian was created. The work compared various classification methods (polynomial naive Bayesian method, logistic regression, linear SVM, random forest and gradient boosting) and assessed the contribution of differentiating features (lexical, semantic and psycholinguistic) to the quality of the classification. Experimental results show that psycholinguistic and semantic features are promising for detecting extremist text.

Many works on the classification of extremist orientation in texts are based on the semantic analysis of the resources of social networks. For marking up texts in supervised machine learning, the presence of class labels is assumed [6].

There are a number of thesauri, specially marked up with regard to the emotional component. These dictionaries have a semantic or graphical structure, which allows you to track the connections between words, their belonging to any group, and so on. These thesauri are needed by computer programs when processing natural language or analyzing the sentiment of text.

In the WordNet-Affect thesaurus, special emotional markers are used, which make it possible to determine synsets by emotional coloration by the gradation of additional marks from negative to positive [7, 8].

The linguistic ontology RuTez is an evolved thesaurus based on the Russian language. It is a hierarchical network of concepts that differs from the approach in WordNet. This ontology was specially created for automatic text processing [9].

SenticNet is another semantic thesaurus for dealing with sets of emotional concepts [10]. The SenticNet thesaurus is presented as an Internet service with an open API for interaction. This thesaurus supports many languages, including Russian.

Deep learning technologies for the classification of extremist texts are considered in [11]. This work uses machine learning, in particular neural networks and deep learning, to classify text as containing “extremist” or “benign” (that is, non-extremist) content. This robust method allows you to effectively learn how to classify extremist multilingual text of various lengths. This research also involved creating a high quality training and testing dataset. The resulting dataset should facilitate further research on this topic.

The goal of [12] is to use a deep learning algorithm to detect radicalization, in contrast to existing works based on machine learning algorithms. A feed-forward neural network based on LSTM is used to detect radical content. The researchers collected a total of over 16,000 entries from a variety of online sources containing news, articles and blogs. These entries are annotated by subject matter experts in three categories: radical, non-radical, and irrelevant, which are then applied to the LSTM-based network to classify radical content. An accuracy of 85.9% was achieved using the proposed approach.

3. Syntax tree optimization algorithm

An algorithm was developed to search for prohibited texts in social networks in this study. The proposed algorithm includes the following stages:

1. Pre-processing of text messages extracted from social networks.

The preprocessing includes carrying out graphematic and morphological analysis and lemmatization of words using the grammatical dictionary of the Lucene system [13].

2. Surface classification based on the thesaurus of dangerous phrases and statements.

A word comparison algorithm was developed to solve this problem. Since a word enters the input of the lemmatization algorithm, and a set of words serves as the output, the problem of comparing a phrase from a text with a key phrase so that the list of all lemmas of the word is sequentially compared with the list of all lemmas of the keyword and proceeds to comparing the following lemmas of the word in the text and phrase.

Thus, the analyzed text can be represented as a model

$$T = \{W_r, W_e, S\}$$

where W_r is a set of words in Russian,

W_e - many words in English,

S is a set of characters.

Moreover, W_r can be represented as

$$W_r = \{w_{r\,1},\ w_{r\,2},\ \dots\ w_{r\,n}\}$$

After processing the text by the lemmatizer, each word $w_{r\,i}$ will be represented by a set of lemmas

$$w_{r\,i} = \{l_{r\,1}, l_{r\,2},\ \dots\ , l_{r\,m}\}$$

With exhaustive search, the comparison of the lemma variants grows rapidly. For example, the sentence in Russian “В ближайшем будущем все будет более технологичным” is shown in Figure 1.

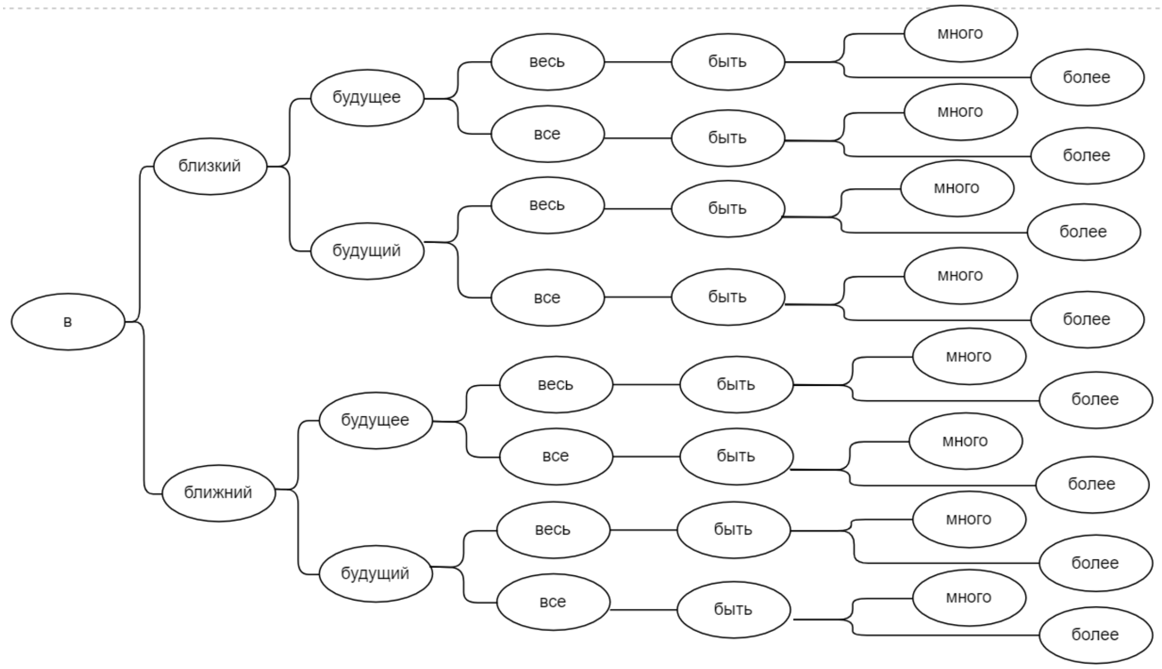


Figure 1: An example of a sentence syntax tree

The result is sixteen sentences to check. And to check for the presence of a lemmatized phrase, you need to store a similar tree in the database of key phrases and check each branch of the phrase tree with each branch of the text tree.

The developed algorithm splits the proposal into nodes. Each node stores a list of lemmas for comparison. Key phrases are also broken down into nodes. The first node of text is then compared to the first node of the key phrase.

The comparison is performed by subtracting the list of lemmas of the keyword node from the list of lemmas of the text node. If the size of the list remains the same, there is no match and the algorithm goes to the second text node and compares it with the first keyword node. If there is a match, the algorithm simultaneously moves to the next nodes of the text and the key phrase.

Reaching the end of a key phrase means a complete match, which means its presence in the text. Thus, there is no branching of both the lemmatized text and the key phrase. There remains the principle of work of character-by-character text comparison, which covers all possible variants of word forms. A schematic of the nodal representation of a lemmatized text is shown in Figure 2.

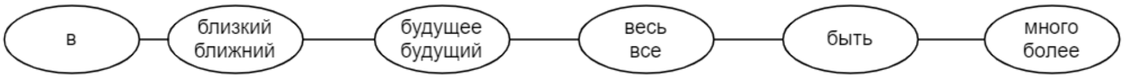


Figure 2: Scheme of the nodal representation of the proposal

If a key phrase matches a passage, the entire text is considered dangerous and is marked with the appropriate flag. This approach separates the most secure posts from the suspicious ones. A suspicious post in this study is a post with a dangerous phrase or its forms.

The result of this stage is a set of sentences and short texts that are previously marked as dangerous.

4. Classification of texts by the method of a naive Bayesian classifier

The bag of words model was chosen. According to this model, any text message is represented as a set of words and phrases [14]. According to the presence in the text of certain words that implicitly correspond to the class, classification occurs.

The probability of assigning a text di to class k is determined by the following model:

$$P\left(k|d_i\right)=\frac{P\left(d_i|k\right)*P\left(k\right)}{P\left(d_i\right)}, \text{ where}$$

P(d$_i$ |k) is the probability of finding document di in the set of documents k;

P(k) is the unconditional probability of class *k* in the training set;

P(d$_i$) is the unconditional probability of the text di in the text corpus of the training set [15].

The most likely class for the text is determined using an estimate of the posterior maximum:

$$k_{map}=\arg\max_{k\in K}\frac{P\left(d_i|k\right)*P\left(k\right)}{P\left(d_i\right)}$$

Since P(d$_i$) = const within the text corpus and taking into account that

$$\mathrm{P}\left(d_i|\mathrm{k}\right)\approx\ \mathrm{P}\left(w_1|\mathrm{k}\right)*\ \mathrm{P}\left(w_2|\mathrm{k}\right)*\ldots\ \ *\mathrm{P}(w_n\mid\mathrm{k})\ =\prod_{s=1}^{n}\mathrm{P}(w_s\mid\mathrm{k}),$$

we get::

$$k_{map}\ =\ \arg\max_{k\in K}\left[\ \mathrm{P}\left(\mathrm{k}\right)*\ \prod_{s=1}^{n}\mathrm{P}(w_s\mid\mathrm{k})\right]$$

In the last expression *P(w$_s$|k)* is the probability of finding a word / phrase w$_s$ in the texts k [16]. Hence:

$$k_{map}\ =\ \arg\max_{k\in K}\left[\ \log\mathrm{P}\left(\mathrm{k}\right)+\ \sum_{s=1}^{n}\log\mathrm{P}(w_s\mid\mathrm{k})\right]$$

An alternative option is to optimize the classification algorithm. It was decided to replace each word in the text with its more abstract version - a hyperonym. This approach will allow you to build more on the meaning of the word itself than on its letter form. Moreover, at the previous stage, no preprocessing of the text was performed and all words were in their arbitrary forms.

This improvement will not only help bring words to normal form, but also help to combine synonyms into a single element, the presence of which will more accurately determine the class of the text resource.

The lemmatizer "Pymorphy2", which has a Russian-language dictionary, was chosen as a text preprocessing tool. Library "WikiWordnet" is a lexical database for the Russian language, based on the Russian-language Wikitonary. It allows you to get synsets - the general concept of a word, its concept. From the found synset, you can get hyperonyms.

Word preprocessing in this case is necessary due to the condition that WikiWordnet works only with the initial form of the word. In the algorithm for forming a bag of words, a condition has been added that allows you to take not the original word (or a phrase for bigrams) from the text, but its hyperonym.

5. Implementation of the system for finding texts of extremist character

To implement the proposed algorithm, a text classification module in the python programming language was developed, which is a separate REST service.

The overall system architecture is shown in Figure 3:

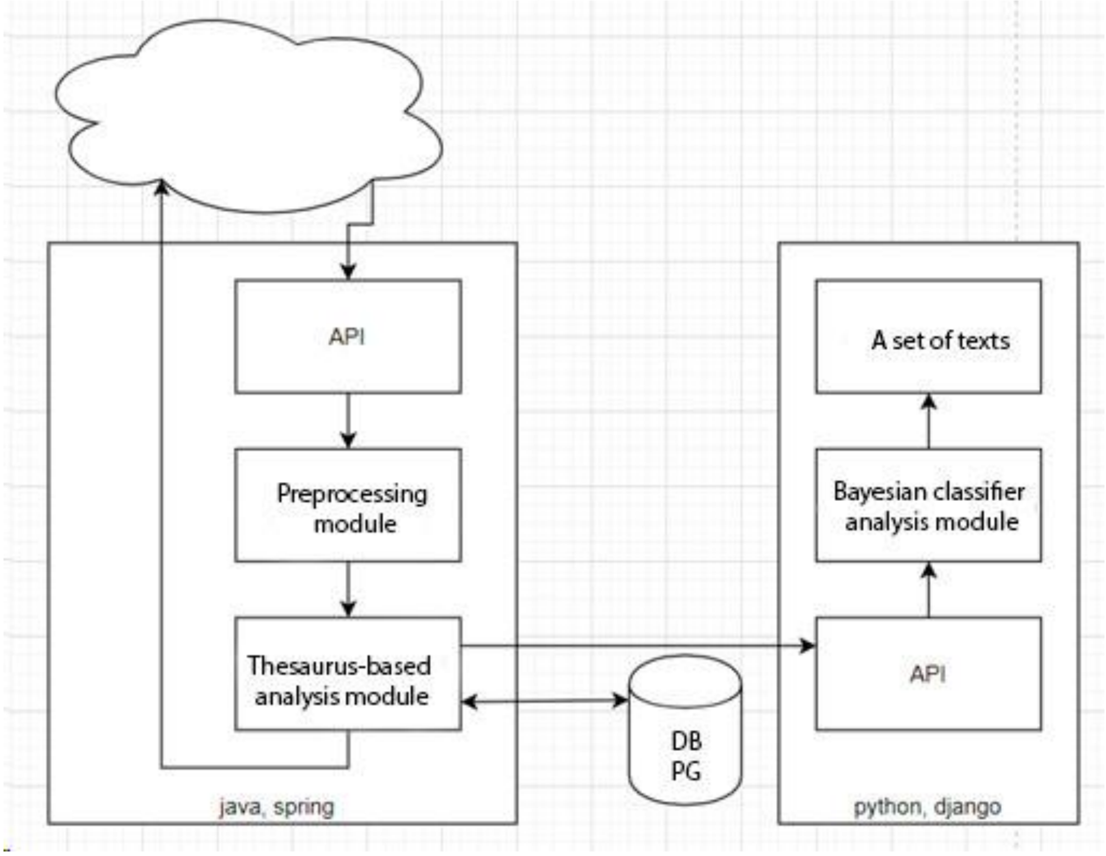
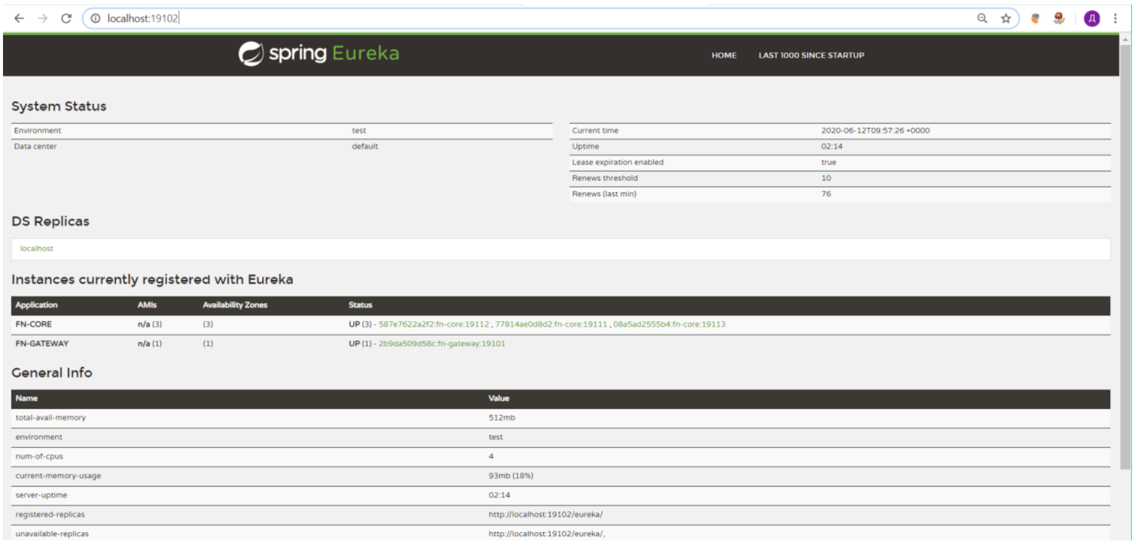


Figure 3: Architecture of the text classification system

The application has a user interface implemented by the Eureka tool for managing and administering microservices. Screenshots of the interfaces are shown in Figures 4 and 5. They show information about all services in the cloud.



System Status

Environment	test
Data center	default

Current time	2020-06-12T09:57:26 +0000
Uptime	02:14
Lease expiration enabled	true
Renews threshold	10
Renews (last min)	76

DS Replicas

localhost

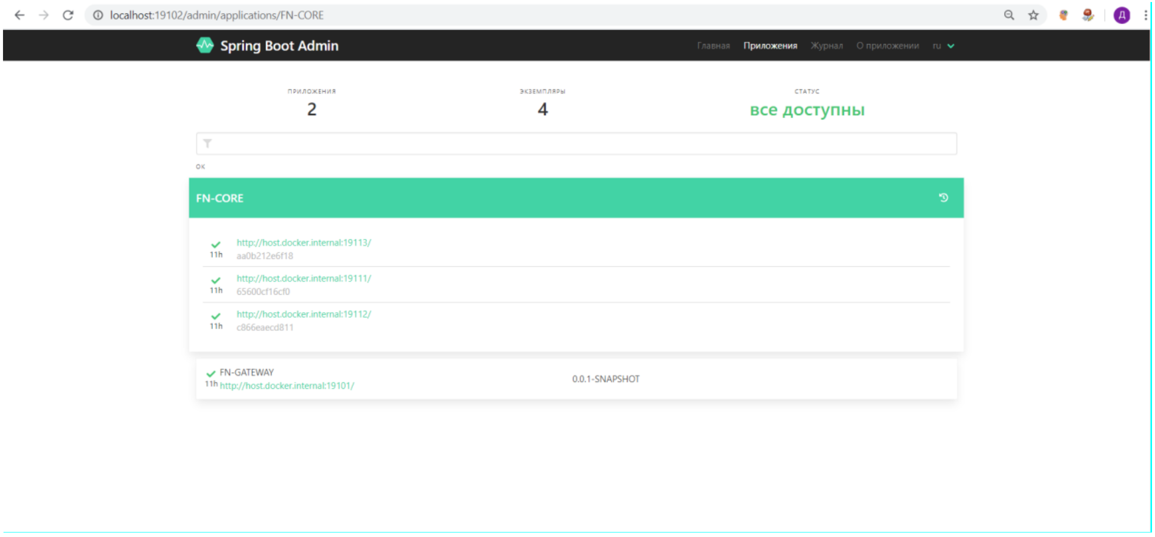
Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
FN-CORE	n/a (3)	(3)	UP (3) - 587e7622a2f2:fn-core:19112 , 77814ae0d8d2:fn-core:19111 , 08a5ad2555b4:fn-core:19113
FN-GATEWAY	n/a (1)	(1)	UP (1) - 2b9da509d58c:fn-gateway:19101

General Info

Name	Value
total-avail-memory	512mb
environment	test
num-of-cpus	4
current-memory-usage	93mb (18%)
server-uptime	02:14
registered-replicas	http://localhost:19102/eureka/
unavailable-replicas	http://localhost:19102/eureka/,

Figure 4: The Eureka interface. Main menu



ПРИЛОЖЕНИЯ	ЭКЗЕМПЛЯРЫ	СТАТУС
2	4	все доступны

ОК

FN-CORE

✓ 11h	http://host.docker.internal:19113/ aa0b212e6f18
✓ 11h	http://host.docker.internal:19111/ 65600cf16cf0
✓ 11h	http://host.docker.internal:19112/ c866eaecd811

✓ FN-GATEWAY 11h http://host.docker.internal:19101/	0.0.1-SNAPSHOT

Figure 5: The Eureka interface. Administration menu.

To train the naive Bayesian classifier, we used annotated texts obtained in previous studies. Of these, two corpuses of texts were formed, 190 documents of dangerous posts and 820 documents of suspicious posts. The division into training and test sets was 9 to 1, respectively.

67 key phrases were found out of 1,277 phrases included in the expert thesaurus. The ten most common phrases are shown in Table 1.

Table 1.

Common Key Phrases

Phrase	Number of posts with the phrase
Strikes	49
Terror	52
Liquidation	58
Kabbalah	61
Gypsies	71
Society	85
Drug addicts	123
Hitler	134
Confrontation	151
Maidan	181

The following results were obtained as a result of experiments:

- The accuracy of the classifier on the test sample was just over 84%.
- During load testing, the waiting time with 100 requests changed from 17 seconds to 28 seconds. The increase in processing time for one request increased by 0.11 seconds, which is acceptable.

6. Conclusion

The system developed as part of the study is a set of microservices. It allows for a three-stage classification of texts. The classification accuracy was increased without a significant increase in processing time, which will allow implementing additional verification steps to refine the results of determining the class of a text resource.

Further development of the project includes the solution of the following tasks:

- Expansion of the thesaurus of key phrases;
- Experimenting with different neural network architectures to find the most efficient model.

7. Acknowledgements

This study was supported by the Russian Foundation for Basic Research (Grants No. 18-47-732007 and 18-47-730035).

8. References

[1] E. Riloff et al., Sarcasm as Contrast Between a Positive Sentiment and Negative Situation. Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP), 704–714 (2013)

[2] I. S. Karabulatova, P. V. Barsukov, I. V. Akhmetov, O. V. Mamatelashvili, F. F. Khizbullin “Network Wars” as a New Type of Devitiaon Processes in the Modern Electronic and Information Society in the Context of Social and Economic Security. MJSS, 6, 6, S.3, 150-159 (2015)

[3] M. M. Polekhina, V. A. Limonzeva, I. S. Karabulatova, M. S. Vyhrystyuk. The Evolution of the Concept of "Terror" / "Terrorism" in Modern Scientific Knowledge as a Factor in Ensuring the Security of Modern Society. In the: Astra Salvensis, 6 (12), 695-704 (2018)

[4] Troussas C. et al. Sentiment analysis of Facebook statuses using Naive Bayes classifier for language learning //IISA 2013. – IEEE, 2013. – C. 1-6.

[5] Devyatkin D. et al. Exploring linguistic features for extremist texts detection (on the material of Russian-speaking illegal texts) //2017 IEEE International Conference on Intelligence and Security Informatics (ISI). – IEEE, 2017. – C. 188-190.

[6] Bogdanov AL, Dulya IS Sentiment analysis of short Russian-language texts in social media // Bulletin of the Tomsk State University. Economy. - 2019. - No. 47.

[7] WordNet-Affect: http://wndomains.fbk.eu/wnaffect.html.

[8] Victoria Bobicev, Victoria Maxim, Tatiana Prodan, Natalia Burciu, Victoria Angheluú Emotions in words: developing a multilingual WordNet-Affect // Technical University of Moldova, 168, Stefan cel Mare bd., Chisinau, Republic of Moldova: http://lilu.fcim.utm.md/RoRUWNAffect.pdf.

[9] RuTez – NLPub: https://nlpub.mipt.ru/PyTeз.

[10] S. M. Zenkevich. Author's designations of prosodic characteristics of emotional speech in the novel "Wives and daughters"by E. Gaskell. Professionally-oriented language teaching: reality and prospects: materials of the scientific practice conferences. Saint Petersburg: Saint Petersburg state University of Economics publishing house, 248-253 (2020)

[11] Johnston A. H., Weiss G. M. Identifying sunni extremist propaganda with deep learning //2017 IEEE Symposium Series on Computational Intelligence (SSCI). – IEEE, 2017. – C. 1-6.

[12] Kaur A., Saini J. K., Bansal D. Detecting Radical Text over Online Media using Deep Learning //arXiv preprint arXiv:1907.12368. – 2019.

[13] Nadezhda Yarushkina, Aleksey Filippov, Vadim Moshkin, Gleb Guskov, Anton Romanov Intelligent Instrumentation for Opinion Mining in Social Media / Proceedings of the II International Scientific and Practical Conference "Fuzzy Technologies in the Industry – FTI 2018" // Ulyanovsk, Russia, 23-25 October, 2018. pp. 50-55

[14] V. Moshkin Unification of Social Media Data When Building a Graph Knowledge Base, 2020 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon), Vladivostok, 2020, pp. 1-6, doi: 10.1109/FarEastCon50210.2020.9271313.

[15] P.D. Ermakov, R.V. Fedyanin. Research of machine learning methods in the problem of automatic sentiment determination of texts in natural language. - Moscow: MSTU im. N.E.Bauman, 2015, pp. 600-614.

[16] N. Yarushkina, A. Filippov, M. Grigoricheva, V.Moshkin The Method for Improving the Quality of Information Retrieval Based on Linguistic Analysis of Search Query In: Rutkowski L., Scherer R., Korytkowski M., Pedrycz W., Tadeusiewicz R., Zurada J. (eds) Artificial Intelligence and Soft Computing. ICAISC 2019. Lecture Notes in Computer Science, vol 11509. Springer, Cham pp 474-485 https://doi.org/10.1007/978-3-030-20915-5_43.