Вестник Сыктывкарского университета.
Серия 1: Математика. Механика. Информатика. 2022.
Выпуск 3 (44)
Bulletin of Syktyvkar University.
Series 1: Mathematics. Mechanics. Informatics. 2022; 3 (44)

Informatics

Original article

Selecting a Solution Method for the Problem of Automating the Classification of Texts Related to Industrial Safety Audits

Yuriy V. Golchevskiy[1], **Lidiya P. Shilova**[2]
[1]Pitirim Sorokin Syktyvkar State University, e-mail: yurygol@mail.ru
[2]Semantic machines, e-mail: shilovalp@bk.ru

Abstract. The importance of solving problems arising from text classification in to-day's world is undeniable, due to the fact that a huge amount of textual in-formation of different kinds is generated, which needs some processing and analysis.

The purpose of the paper is to find the best way to automate the classi-fication of industrial safety audits on the example of a large industrial enter-prise. Existing solutions and tools in the field of text classification problems were investigated. This work was carried out using the Scikit-Learn library. Based on a sample of 28,000 industrial safety au-dits, which were evenly divided into 14 classes, several different methods provided by the library were tested. During the analysis of the results, the linear method was proposed as the most accurate and fastest method investi-gated. Although this method does not provide the full level of reliable classi-fication required from a practical point of view, the results can noticeably simplify and speed up staff work.

Keywords: Machine Learning, Text Classification, Industrial Safety Audits

For citation: Golchevskiy Yu. V., Shilova L. P. Selecting a Solution Method for the Problem of Automating the Classification of Texts Related

to Industrial Safety Audits. *Vestnik Syktyvkarskogo universiteta. Seriya 1: Matematika. Mekhanika. Informatika* [Bulletin of Syktyvkar University, Series 1: Mathematics. Mechanics. Informatics], 2022, No 3 (44), pp. 21–32. https://doi.org/10.34130/1992-2752_2022_3_21

ИНФОРМАТИКА

Научная статья

УДК 004.42
https://doi.org/10.34130/1992-2752_2022_3_21

Выбор метода решения задачи автоматизации классификации текстов, связанных с аудитами промышленной безопасности

Юрий Валентинович Гольчевский[1], **Лидия Павловна Шилова**[2]
[1]Сыктывкарский государственный университет им. Питирима Сорокина, e-mail: yurygol@mail.ru
[2]ООО «Смысловые решения», e-mail: shilovalp@bk.ru

Аннотация. Важность решения проблем, возникающих при классификации текстов, неоспорима в связи с тем, что в современном мире генерируется огромное количество текстовой информации различного рода, которая нуждается в некоторой обработке и анализе. Целью данной работы является поиск наилучшего способа автоматизации классификации аудитов промышленной безопасности на примере крупного промышленного предприятия. В ходе исследования были изучены существующие решения и инструменты в области задач классификации текстов. Работа была выполнена с использованием библиотеки Scikit-Learn. На основе выборки из 28 тысяч аудитов промышленной безопасности, которые были равномерно разделены на 14 классов, было протестировано несколько различных методов, предоставляемых библиотекой. В ходе анализа результатов линейный метод был предложен как наиболее точный и быстрый из исследованных. Хотя этот метод не обеспечивает полного уровня надежной классификации, требуемого с практической точки зрения, результаты могут заметно упростить и ускорить работу персонала, решающего представленные задачи.

Ключевые слова: машинное обучение, классификация текстов, аудит промышленной безопасности

Для цитирования: Гольчевский Ю. В., Шилова Л. П. Выбор метода решения задачи автоматизации классификации текстов, связанных с аудитами промышленной безопасности // *Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика.* 2022. Вып. 3 (44). С. 21—32. https://doi.org/10.34130/1992-2752_2022_3_21

Introduction

Machine learning is now widespread in many aspects of human activity. For example, a computer can recognize images in photos, make medical diagnoses, solve legal issues, make predictions about the situation in the stock markets, and this is only a small part of the tasks solved by modern intelligent systems [1–5]. Many people use such systems, sometimes without knowing it, for example, when receiving search engine results, encountering contextual advertising on the Internet, using spam filtering and in many other situations. Programs, like humans, learn by analyzing data in different areas. At the same time, each problem requires its own specific set of data to learn and its own model [6].

The importance of solving the problems of text classification has increased sharply due to the fact that in the modern world a huge amount of textual information of different plan (technical, scientific, creative and other directions) is generated. Classification as a problem is one of the rapidly developing fields and has wide application horizons in information processing and data mining. Various architectures, approaches and algorithms have been proposed in scientific publications, e.g. [7–9].

The task of classifying documents related to industrial safety is no exception. For example, in [10] a method based on automatic classification of construction accident messages, which can be useful in developing risk management strategies is proposed, in [11] the authors apply classification of texts based on the use of ontologies and propose an ontology of construction safety domain and its corresponding knowledge base.

According to our calculations, at the researched enterprise, which is being automated, a user spends from 10 to 30 seconds to fill in one field of an industrial safety audit in an electronic document. More than 70 thousand industrial safety audits can be generated in a year. If the classification process were to be automated, about 200 to 600 hours of work time could be saved.

This is the purpose of this study, which is to find the best way to automate classification of industrial safety audits on the example of one enterprise for further routing of such documents. In solving this task the problems of text classification were studied, several different methods of classification of industrial safety audits were applied, the obtained results were compared and conclusions about the effectiveness of the considered methods were drawn.

Methods

Classification refers to machine learning “with a teacher”, which requires partitioned training data. The classification algorithm has to assign a document to one of the classes, the list of which is known in advance. As a process, classification typically proceeds as follows: preprocessing, object design, dimension decomposition, model selection and model evaluation. In [12] an overview of each step and a review and comparison of classification algorithms are provided, while in [13–15] authors discusses methods and problems associated with different approaches to data mining, including machine learning-based text analysis.

An important part of text classification is text preprocessing. In [16] preprocessing of the native language text is defined as bringing the text into a form that is suitable for further work. That is, the first step is to obtain a set of texts that are used as the research base.

A total of 28 000 industrial safety audits were downloaded from the existing database and divided uniformly into 14 classes. By class, we mean the “Theme” field of the document. All audit text fields (“Justification”, “Location”, “Observation”, etc.) were merged into one text field. Thus, the resulting file represented a table, where the first “theme” column is the security audit class (topic) and the second “description” column is the document text fields. The text in the “description” column was stripped of punctuation marks and numbers, reduced to lower case and each word written in its initial form. For clarity, Figure 1 shows a word cloud (in Russian) of the “description” field without preprocessing (left) and with processing (right), and Figure 2 shows an example of prepared data.

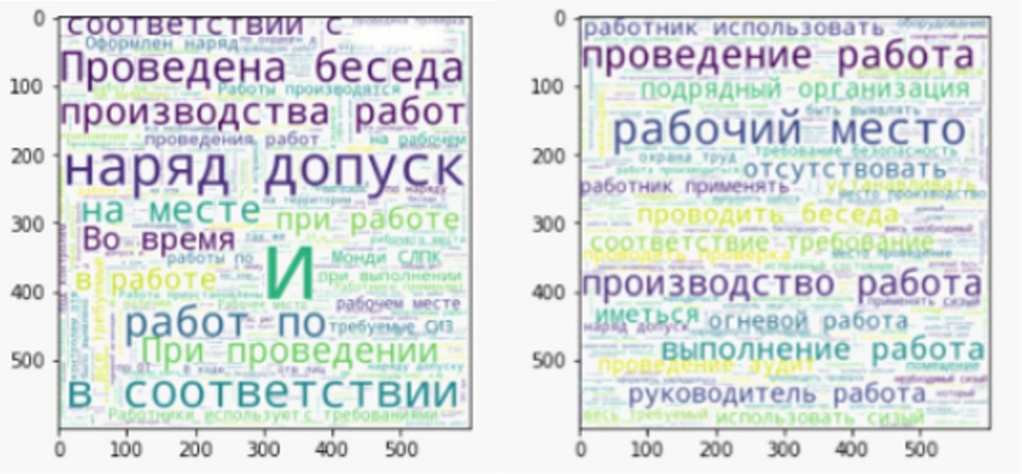


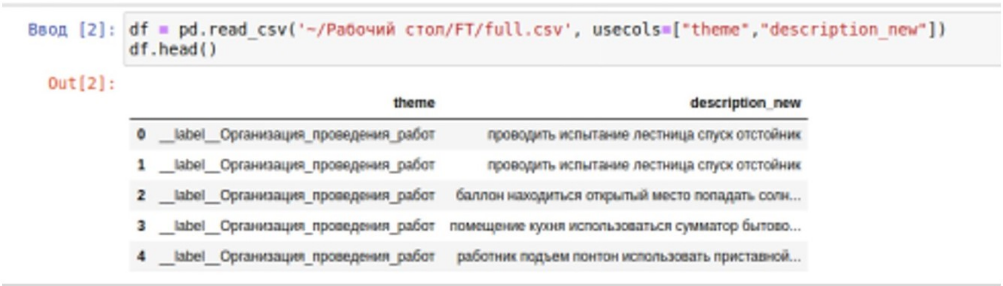
Fig. 1. Word cloud before preprocessing (left) and after processing (right)

The study then used 20% of the data for the test sample and 80% for the training sample.

Then it is required to represent the document in the form of some numerical model. Most often, a document is represented as a multidimensional vector [17]. One simple vectorization method is the “Bag of words” (BOW) [18].

The accuracy of classification depends on the choice of hyperparameters. Hyperparameters are often selected using a matching method, although there are also special functions that allow automating this. Once the hyperparameters have been selected, the classifier can be started.

After training the classifier on a training sample. The classifier “predicts” classes for the test sample. This builds an error matrix.



```
Ввод [2]: df = pd.read_csv('~/Рабочий стол/FT/full.csv', usecols=["theme","description_new"])
          df.head()
```

Out[2]:

	theme	description_new
0	__label__Организация_проведения_работ	проводить испытание лестница спуск отстойник
1	__label__Организация_проведения_работ	проводить испытание лестница спуск отстойник
2	__label__Организация_проведения_работ	баллон находиться открытый место попадать солн...
3	__label__Организация_проведения_работ	помещение кухня использоваться сумматор бытово...
4	__label__Организация_проведения_работ	работник подъем понтон использовать приставной...

Fig. 2. Example of prepared data

One of the main evaluations of the quality of the classifier is precision, which is calculated as the ratio of true positives to total positives. The second parameter is recall, which is the ratio of true positives to all possible positives. Another characteristic is the weighted average value of accuracy and completeness. This is called the F-value. F varies from 0 to 1, where 1 is the best (ideal) value for F.

The Scikit-Learn library for the Python programming language was used for the study, providing the necessary machine learning algorithms as well as supporting tools and utilities. It provides a standardised API, which makes it user-friendly and easy to use. More information about Scikit-Learn can be found in [19–21].

Vectorization of the texts was done using the CountVectorizer class, which is based on the BOW model. The result was a matrix that contained the number of occurrences of each word. The GridSearchCV class was used for parameter selection – a grid search. The input is a model and various values of hyperparameters (hyperparameter grid). Then, for each possible hyperparameter values combination, the method calculates the error and chooses the combination at which the error is minimal. More details can be obtained in [22].

Results and Conclusions

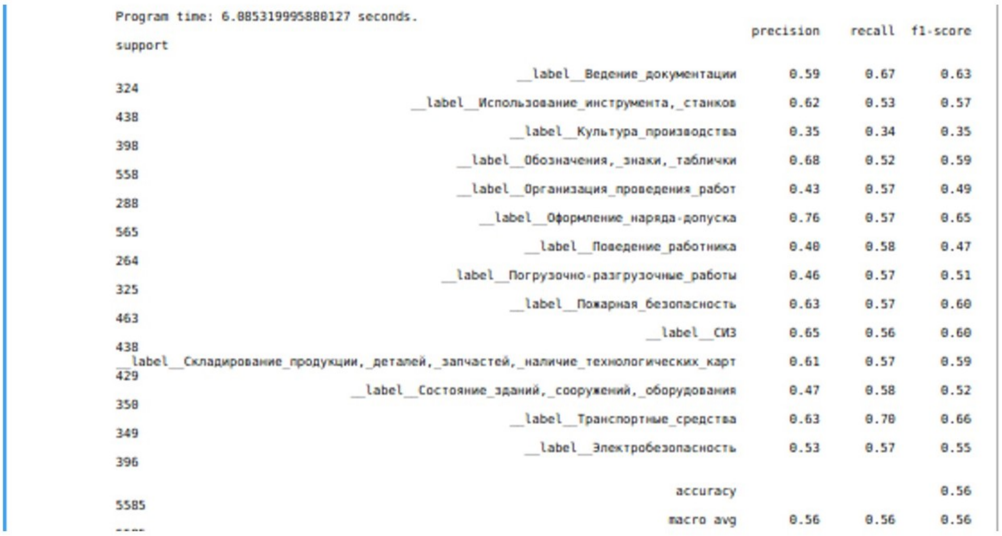
The following methods from Scikit-Learn library were chosen for comparison: MultinomialNB (probabilistic), KNeighborsClassifier (metric), Random-ForestClassifier (logical), LinearSVC (linear) and MLPClassifier.

MultinomialNB method represents an implementation of Naive Bayes. KNeighborsClassifier represents an implementation of the k-nearest neighbour method. The basic idea is that if some point A is very far from point B, and B is very close to point C, then A and C are also far away from each other and there is no need to calculate the distance between them. The data structure used is in the form of trees. In the current model the option auto is selected for the algorithm parameter, i.e. the method is selected automatically; the parameters are set so that the closer neighbours have more influence.

RandomForestClassifier is an implementation of decision tree method, while LinearSVC method is based on reference vector method.

In order not to complicate the conclusions presented, the chosen parameters for the above methods will not be given in detail in this paper. The MLPClassifier method was run with default parameters due to long running.

Summary tables were obtained for each of the methods. As an example, the results obtained for KNeighborsClassifier and LinearSVC methods are shown in Figures 3 and 4 respectively.



```
Program time: 6.085319995880127 seconds.
                                                                          precision    recall  f1-score
support

                                      __label__Ведение_документации         0.59      0.67      0.63
324
                         __label__Использование_инструмента,_станков         0.62      0.53      0.57
438
                                      __label__Культура_производства         0.35      0.34      0.35
398
                               __label__Обозначения,_знаки,_таблички         0.68      0.52      0.59
558
                               __label__Организация_проведения_работ         0.43      0.57      0.49
288
                                  __label__Оформление_наряда-допуска         0.76      0.57      0.65
565
                                       __label__Поведение_работника         0.40      0.58      0.47
264
                             __label__Погрузочно-разгрузочные_работы         0.46      0.57      0.51
325
                                      __label__Пожарная_безопасность         0.63      0.57      0.60
463
                                                       __label__СИЗ         0.65      0.56      0.60
438
__label__Складирование_продукции,_деталей,_запчастей,_наличие_технологических_карт         0.61      0.57      0.59
429
                    __label__Состояние_зданий,_сооружений,_оборудования         0.47      0.58      0.52
358
                                      __label__Транспортные_средства         0.63      0.70      0.66
349
                                       __label__Электробезопасность         0.53      0.57      0.55
396

                                                           accuracy                             0.56
5585
                                                          macro avg         0.56      0.56      0.56
```

Fig. 3. Result of applying the KNeighborsClassifier method

Table

Table of results for the different methods

Method	Learning time, in secs	F
MultinomialNB	0.9	0.63
KNeighborsClassifier	6.9	0.56
RandomForestClassifier	36.6	0.65
LinearSVC	8.9	0.65
MLPClassifier	1573.5	0.58

A summary table with precision, recall and F-value (f1-score) estimates was also obtained. The data are presented in Table 1. The most important value for our study is the average F-value for all classes.

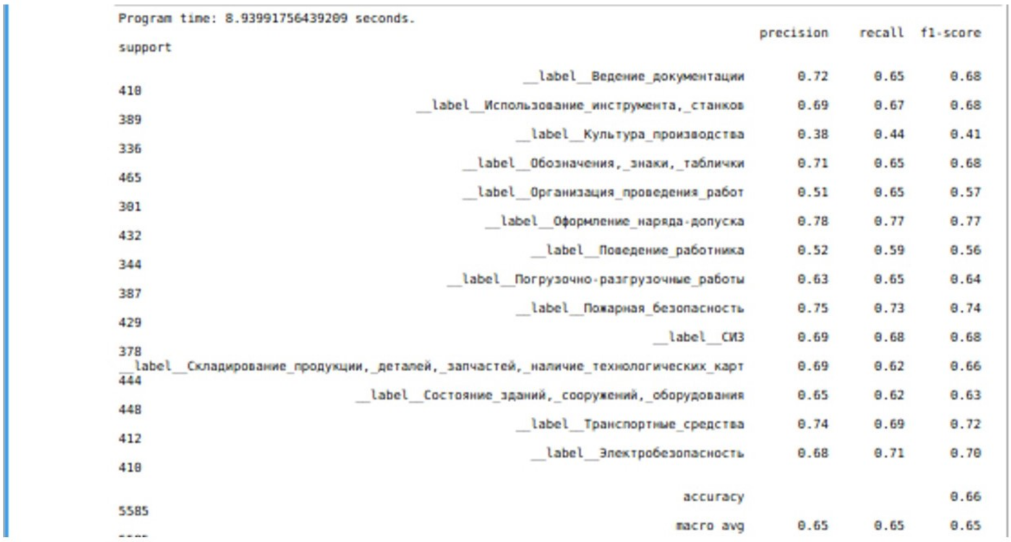
It was found that RandomForestClassifier and LinearSVC methods have the best F-value. Nevertheless, LinearSVC method is faster than RandomForestClassifier.

In the future, it is planned to conduct additional research related to the choice of other methods of text vectorization. For example, methods based on neural networks that convert words into “meaningful vectors” [23], as well as the use of ensembles of methods [24].

This study investigated the stages of classification and the possibilities of using automation to classify texts related to industrial safety audits. A comparison of industrial safety audit classification methods based on the application of methods and tools provided by the Scikit-Learn library was performed. During the analysis of the results, it was proposed to use LinearSVC method as the most accurate and fastest of the investigated ones.

The LinearSVC method was implemented in a corporate document analysis system for further testing on real data.

Although this method does not provide the full level of reliable classification required from a practical point of view, the results of its work can significantly simplify and speed up the work of company employees involved in processing of industrial safety audits, which was found as a result of testing the use of this approach.



```
Program time: 8.93991756439209 seconds.
                                                                                 precision    recall  f1-score
support

                                                  __label__Ведение_документации       0.72      0.65      0.68
418
                                    __label__Использование_инструмента,_станков       0.69      0.67      0.68
389
                                                 __label__Культура_производства       0.38      0.44      0.41
336
                                           __label__Обозначения,_знаки,_таблички       0.71      0.65      0.68
465
                                           __label__Организация_проведения_работ       0.51      0.65      0.57
381
                                              __label__Оформление_наряда-допуска       0.78      0.77      0.77
432
                                                   __label__Поведение_работника       0.52      0.59      0.56
344
                                         __label__Погрузочно-разгрузочные_работы       0.63      0.65      0.64
387
                                                 __label__Пожарная_безопасность       0.75      0.73      0.74
429
                                                                   __label__СИЗ       0.69      0.68      0.68
378
__label__Складирование_продукции,_деталей,_запчастей,_наличие_технологических_карт       0.69      0.62      0.66
444
                          __label__Состояние_зданий,_сооружений,_оборудования       0.65      0.62      0.63
448
                                                 __label__Транспортные_средства       0.74      0.69      0.72
412
                                                   __label__Электробезопасность       0.68      0.71      0.70
418

                                                                       accuracy                           0.66
5585
                                                                      macro avg       0.65      0.65      0.65
```

Fig. 4. Result of applying the LinearSVC method

References

1. «Post Bank»: we saved hundreds millions rubles using biometrics. Available at: https://bloomchain.ru/newsfeed/k-kontsu-2019-goda-vseh-klientov-pochta-banka-budut-identifitsirovat-po-biometrii (accessed: 2021/10/05). (in Russ.)

2. Loan scoring and fight against swindlers: AI in Russian banking sector. Available at: https://aiconference.ru/en/article/kreditniy-skoring-i-borba-s-moshennikami-ob-ii-v-bankovskoy-sfere-rossii-96820 (accessed: 2021/10/05).

3. Neurohive – Neural Networks. Available at: https://neurohive.io/en/ (accessed: 2021/10/05).

4. **Hannun A. Y., Rajpurkar P., Haghpanahi M. et al.** Cardiologist-level arrhythmia de-tection and classification in ambulatory electrocardiograms using a deep neural net-work. *Nat Med 25*, 2019. Pp. 65–69. DOI: 10.1038/s41591-018-0268-3.

5. **Koshy R., Padalkar A., Nikam N., Jain V.** Easy verdict: Digital assistant to resolve criminal litigation *10th International Conference on Advances in Computing, Con-trol, and Telecommunication Technologies*, 2019. Pp. 17–23.

6. **Domingos P.** The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World. Basic Books, New York, 2015.

7. **Mou S., Du P., Cheng Z.** A brain-inspired information processing algorithm and its application in text classification. *Expert Systems with Applications*, Vol. 177, 2021. DOI: 10.1016/j.eswa.2021.114828.

8. **Asim M., Javed K., Rehman A., Babri H. A.** A new feature selection metric for text classification: eliminating the need for a separate pruning stage. *International Journal of Machine Learning and Cybernetics*, 12(9), 2021. Pp. 2461–2478. DOI: 10.1007/s13042-021-01324-6.

9. **Shimomoto E. K., Portet F., Fukui K.** Text classification based on the word subspace representation. *Pattern Analysis and Applications*, 24 (3), 2021. Pp. 1075-1093. DOI: 10.1007/s10044-021-00960-6.

10. **Zhang J., Zi L., Hou Y. et al.** A C-BiLSTM approach to classify construction accident reports. *Applied Sciences (Switzerland)*, 10(17), 2020. DOI: 10.3390/APP10175754.

11. **Chi N.-W., Lin K.-Y., Hsieh S.-H.** Using ontology-based text classification to assist Job Hazard Analysis. *Advanced Engineering Informatics*, 28(4), 2014. Pp. 381–394. DOI: 10.1016/j.aei.2014.05.001.

12. **Zhan, T.** Classification Models of Text: A Comparative Study. *IEEE 11th Annual Computing and Communication Workshop and Conference, CCWC 2021*, 2021. Pp. 1221-1225. DOI: 10.1109/CCWC51732.2021.9375918.

13. **Li Y., Dai G., Li G.** Feature selection method of text tendency classification. *Proceedings – 5th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD-2008*, 2008. Pp. 34–37. DOI: 10.1109/FSKD.2008.263.

14. **Flach P.** *Machine Learning: The Art and Science of Algorithms That Make Sense of Data.* Cambridge University Press, New York, 2012.

15. **Rani M.S., Sumathy S.** Analysis on various machine learning based approaches with a perspective on the performance. *Innovations in Power and Advanced Computing Technologies*, i-PACT 2017, 2017. Pp. 1–7. DOI: 10.1109/IPACT.2017.8244998.

16. **Bengfort B., Bilbro R., Ojeda T.** *Applied Text Analysis with Python: Enabling Lan-guage-Aware Data Products with Machine Learning.* 1st edn. O'Reilly Media, Inc., 2018.

17. **VanderPlas J.** *Python Data Science Handbook: Essential Tools for Working with Data. 1st edn.* O'Reilly Media, Inc., 2017.

18. **Chollet F.** *Deep Learning with Python.* Manning Publications Co, 2017.

19. **Géron A.** *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. 2nd edn.* O'Reilly Media, Inc., 2019.

20. **Buitinck L., Louppe G., Blondel M. et al.** API design for machine learning software: experiences from the scikit-learn project. *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, 2013. Pp. 108–122.

21. **Rashka S., Mirjalil V.** *Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow. 2nd edn.* Packt Publishing, 2017.

22. **Brink H., Richards J. W., Fetherolf M.** *Real-World Machine Learning. 1st edn.* Manning Publications Co, 2016.

23. fastText. Library for efficient text classification and representation learning. Available at: https://fasttext.cc/ (accessed: 2021/10/05).

24. **Opitz D., Maclin R.** Popular ensemble methods: An empirical study. *Journal of Artifi-cial Intelligence Research*, 11, 1999. Pp. 169–198. DOI: 10.1613/jair.614.

Сведения об авторах / Information about authors
Юрий Валентинович Гольчевский / Yuriy V. Golchevskiy
к.ф.-м.н, доцент, заведующий кафедрой прикладной информатики / Ph.D. in Physics and Mathematics, Associate Professor, Head of Applied Informatics Department
Сыктывкарский государственный университет им. Питирима Сорокина / Pitirim Sorokin Syktyvkar State University
167001, Россия, г. Сыктывкар, Октябрьский пр., д. 55 / 167001, Russia, Syktyvkar, Oktyabrsky Ave., 55

Лидия Павловна Шилова / Lidiya P. Shilova
аналитик / analyst
ООО «Смысловые машины» / Semantic machines
167026, Россия, г. Сыктывкар, пр-кт Бумажников, д. 2 / 167026, Russia, Syktyvkar, Boumazhnikov Ave., 2

Статья поступила в редакцию / The article was submitted 15.08.2022
Одобрено после рецензирования / Approved after reviewing 21.09.2022
Принято к публикации / Accepted for publication 28.09.2022