Terra Linguistica. 2024, Том 15, № 3. С. 109–123.
Terra Linguistica. 15 (3) (2024) 109–123.

Research article
UDC 004.8 + 81'33
DOI: https://doi.org/10.18721/JHSS.15310

DATASET PREPROCESSING EFFECTS ON BI-LSTM-BASED CONCEPT TAGGING OF TEXT TOKENS

O.I. Babina ✉, **A.Yu. Zinoveva**, **E.D. Nerucheva**

South Ural State University,
Chelyabinsk, Russian Federation

✉ babinaoi@susu.ru

Abstract. The paper considers the problem of natural language dataset preprocessing to improve the neural network model performance. The aim of the study is to find out the dataset preprocessing parameters that ensure higher performance of the model aimed at correlating textual input (a sequence of lexical units) with semantic, or conceptual, classes, i.e. concept tagging. Our methodology includes: a) modeling conceptual annotation of textual units, b) experimenting with textual dataset preprocessing options. The model that we propose takes as input tokens (in lowercase) representing words and multi-component lexical units (phrases), some of which are domain concept related. Since each token may refer to several conceptual classes, the concept tagging task is treated as a multi-label classification problem. In this research, we deal with the corpus of news reports on terrorist attacks in English. We experimented with preprocessing the corpus-based dataset by: a) lemmatizing tokens, b) removing stop words, and c) including sentence separators as individual tokens in the model vocabulary. The multi-label classification model used for the training experiments was a neural network that constructs sequences of lexical unit embeddings and feeds them into a bidirectional long short-term memory (Bi-LSTM) model. The experimental results show that the dataset preprocessed according to all the above-mentioned procedures demonstrated the highest micro-, macro- and weighted averaged F1-scores. The per-class F1-score on the test dataset reaches 88% for the class characterized by high frequency and low lexical variability in the training, validation, and test samples. The novelty of the paper lies in the proposed approach to content analysis of news reports on terrorist attacks using the proposed multi-label classification model. New results were obtained during experimenting with the differently preprocessed corpora of news reports on terrorist attacks. The proposed method may be used for content analysis of news reports specific to other subject areas.

Keywords: semantic tagging, natural language processing, Bi-LSTM, multi-label classification, data preprocessing, news corpus, terrorism.

Citation: Babina O.I., Zinoveva A.Yu., Nerucheva E.D., Dataset preprocessing effects on Bi-LSTM-based concept tagging of text tokens, Terra Linguistica, 15 (3) (2024) 109–123. DOI: 10.18721/JHSS.15310

Научная статья
DOI: https://doi.org/10.18721/JHSS.15310

ВЛИЯНИЕ ПРЕДВАРИТЕЛЬНОЙ ОБРАБОТКИ НАБОРА ДАННЫХ НА КОНЦЕПТУАЛЬНУЮ РАЗМЕТКУ ТЕКСТОВЫХ ТОКЕНОВ НА ОСНОВЕ ДВУНАПРАВЛЕННОЙ LSTM

О.И. Бабина ✉, **А.Ю. Зиновьева**, **Е.Д. Неручева**

Южно-Уральский государственный университет,
Челябинск, Российская Федерация

✉ babinaoi@susu.ru

Аннотация. В статье рассматривается проблема предварительной обработки набора данных на естественном языке для повышения качества работы нейросетевой модели. Цель исследования – выяснить параметры предварительной обработки набора текстовых данных, обеспечивающие более высокие показатели качества модели, направленной на соотнесение текстового ввода (последовательности лексических единиц) с семантическими (концептуальными) классами, т.е. на концептуальную разметку текста. Наша методология включает в себя: а) моделирование концептуального аннотирования текстовых единиц; б) экспериментирование с вариантами предварительной обработки набора текстовых данных. Специфика модели концептуального аннотирования, которую мы предлагаем, состоит в том, что она принимает на вход токены (в нижнем регистре), представляющие собой слова и многокомпонентные лексические единицы (словосочетания), некоторые из них аннотированы концептами предметной области. Поскольку каждый токен может относиться к нескольким концептуальным классам, задача разметки концептов ставится как задача классификации по нескольким меткам. В данном исследовании мы используем в качестве материала корпус новостных сообщений о террористических актах на английском языке. Мы экспериментировали с предварительной обработкой набора данных на основе корпуса путем: а) лемматизации токенов; б) удаления стоп-слов; в) включения разделителей предложений в качестве отдельных токенов в словарь модели. Модель классификации с несколькими метками, используемая для экспериментов с обучением, представляла собой нейронную сеть, которая строит последовательности эмбеддингов лексических единиц и передает их на обработку в последовательно расположенные двунаправленные слои долгой краткосрочной памяти (Bi-LSTM-слои). Результаты экспериментов показывают, что набор данных, предварительно обработанный в соответствии со всеми вышеупомянутыми процедурами, продемонстрировал самые высокие микро-, макро- и средневзвешенные значения показателя F1. Поклассовая оценка F1 достигает на тестовом наборе данных значения 88% для класса, характеризующегося большой употребительностью и низкой лексической вариативностью в обучающей, проверочной и тестовой выборках. Новизна работы заключается в предложенном подходе к контент-анализу новостных сообщений о терактах с использованием предложенной модели классификации по нескольким меткам. Новые результаты были получены в ходе экспериментов с различными предварительно обработанными корпусами новостей о терактах. Предложенная методика может быть масштабирована для проведения контент-анализа новостных сообщений, специфичных для других предметных областей.

Ключевые слова: семантическая разметка, обработка естественного языка, двунаправленная LSTM, классификация по нескольким меткам, предобработка данных, корпус новостных текстов, терроризм.

Для цитирования: Бабина О.И., Зиновьева А.Ю., Неручева Е.Д. Влияние предварительной обработки набора данных на концептуальную разметку текстовых токенов на основе двунаправленной LSTM // Terra Linguistica. 2024. Т. 15. № 3. С. 109–123. DOI: 10.18721/JHSS.15310

Introduction

Nowadays, the need to automate text analysis is increasing, since information is mainly distributed in the form of texts in the natural language. At the same time, the number of texts increases, which leads to the practical impossibility to obtain analytical data based on manual text data processing.

A popular method to analyze political, social, religious, psychological views, comprehend industrial and economic trends is content analysis (CA) of texts. The purpose of CA is to extract information about the phenomenon of interest based on quantitative and qualitative text analysis and to make replicable and valid inferences from texts (or other meaningful matter) to the contexts of their use [6]. CA today is widely applied in different fields. Thus, it is used in psycholinguistics to analyze the patterns of speech activity; in intercultural communication – to identify cultural dominants and culturally specific parameters of representatives of different cultures; in political science – to monitor the dynamics of social processes and political preferences, as well as to draw conclusions about the religious affiliations of political leaders; in social science – to conceptualize problems of a city, levels of depression among citizens; in mass media – to detect editorial biases of journals and newspapers; in advertising – to build effective strategies for attracting attention to a product, etc.

The quantitative CA procedure involves determining analysis units representing semantic categories relevant to the solution of the task at hand, then establishing the corresponding units of account, and then, by counting the latter, it is possible to estimate the importance or prevalence of certain ideas and opinions, as well as the relationship between the target concepts with subsequent interpretation of the obtained results. Consequently, to make a decision or draw a conclusion using CA, one has to annotate text units with concept-related labels, which will form a ground for calculations and semantic interpretation of the obtained data.

Generally, natural language processing (NLP) tasks, in the essence, correspond to the problem of correctly classifying textual input. In recent years, the development of artificial intelligence techniques has led to an increase in attempts to solve NLP tasks using machine learning and deep learning approaches. Thus, the problem of topic modeling for a set of documents is effectively solved using probabilistic algorithms, such as Expectation Maximization Algorithm, Probabilistic Latent Semantic Analysis, Latent Dirichlet Allocation and its modifications, see, for example, [5, 6]. Other studies also demonstrate decent results in solving the problem of text classification based on the use of Naïve Bayes, Logistic Regression, Support Vector Machine and other classifiers for assigning a label to the text.

Besides traditional machine learning techniques, the neural network approach to solving NLP tasks has been increasingly developed recently. Among the well-known architectures, the recurrent neural network architecture, as well as its bidirectional long short-term memory (Bi-LSTM) modification, have proven their productivity in the context of solving text classification and sequence labeling problems. Many solutions are now freely available online. Thus, the Hugging Face framework[1] combines a set of solutions for NLP tasks, including pretrained models for different languages and pipelines for solving translation tasks, text summarization, question answering, text classification, sequence labeling, etc.

In this paper, we consider the problem of a sequence labeling that arises when there is a need to assign a certain text segment to a class or classes. Often, sequence classification is performed at the sentence or short-text level. The problem of sequence labeling may be set as a problem of:

- binary classification. It is the simplest case, when the text (sequence) needs to be labeled as belonging or not belonging to one class. It is the case when we need to answer a question formulated as follows: ‘Is this text about ecology?’, ‘Do people rate the film positively?’, ‘Is this message spam?’ etc. The answers would be either ‘yes’ or ‘no’, which is what the classifier should predict.
- multi-class classification. It is an extension of the binary classification, when the number of classes is greater than two. For each sample, the classifier selects only one label (class) from the set, so the classes are mutually exclusive.

[1] Hugging Face. Available at: https://huggingface.co/ (accessed 30.03.2023).

• multi-label classification. In this case, the textual input may be associated with a number of labels from a predefined set. In this case, the same sample may be labeled with more than one class at a time. Therefore, the categories in the set intersect.

• multi-task classification. It is the case of multi-class-multi-output classification, when textual inputs are classified using a set of non-binary labels. That is, the number of classes and the number of valid values per class is greater than two.

In this paper, we consider the problem of the CA task, which we treat as a multi-label classification problem at the stage of multi-word unit annotation, or tagging, with content labels. Further in this paper, in Materials and Methods section, we present the dataset structure and preprocessing options we experimented with, and also the architecture of the Bi-LSTM-based classifier used in the experiments. Then, in Results section, we present quantitative data from the experiments based on differently preprocessed datasets. Then, in Discussion section, we provide some insights into the reasons for the results obtained. Finally, in Conclusion section, we summarize the main results and give a brief view on further research directions.

Materials and Methods

The aim of this study is to find efficient ways of textual dataset preprocessing to solve the problem of labeling single- and multi-word units with concepts. The reason for this task is the need to have a per-token semantically annotated data to store CA that is very useful to support decision-making in many domains driven by human opinion. Our hypothesis is that labeling multi-word units (in addition to words), converting text to lowercase, lemmatization, removing stop words and treating end-of-sentence punctuation marks as separate tokens may improve the performance of the concept tagging model.

The basic CA procedures include deciding on concepts, or semantic categories, related to the analysis goal, selecting the units manifesting the concepts, then coding the text using concept-related labels based on finding concept manifestations in the text, and finally, interpreting the data on qualitative and quantitative concept distribution in the text. The key carriers of semantics in the text are lexical units. The meanings of lexical units and their combinations determine the informative content of the text. Based on these considerations, CA involves identifying lexical units and labeling them with the corresponding semantic category relevant for CA.

Hence, keeping in mind the CA task, our approach to labeling the text consists of mapping lexical units in the text to specific classes, representing domain concepts relevant for CA. We propose using a deep learning framework based on the Bi-LSTM classifier to solve the task. A model performance is determined by a set of parameters, including dataset preprocessing, which we will discuss in more detail.

Datasets

The material for our study is a tagged corpus of terrorism domain e-news in English, in which lexical units (words and phrases) are labeled with domain-relevant classes. The corpus was complied using a conceptual annotation platform [12] based on the domain ontology and ontolexicon containing single- and multi-word units (tokens) automatically extracted from the corpus using the LanaKey tool [11] and mapped onto the domain ontology classes.

The choice of extracted keywords, including multi-word units, as features is due to the following reasons. First, multi-word units are domain-specific and, since the words that comprise them are presented in the microcontext of other words which function as meaning specifiers for each other, tend to be monosemantic, which leads to a solution to the problem of multiple interpretations of individual words (when they are used as features) by the model. This means that multi-word units as features would eliminate ambiguity and, thus, contribute to increasing the accuracy of the model. Second, a sufficiently large number of concepts are verbalized by set phrases constructed from two or more words (e.g., *terrorist attack*, *took place*, etc.). Moreover, it sometimes happens that the components of such multi-word units can hardly be referred to a terrorism domain concept when used independently, while as part of a phrase they form an unambiguous concept (e.g., *have blown themselves up*, *knife wielding*

man, *suicide belt*, etc.). Therefore, the use of multi-word units as features would contribute to a higher memorability of the model. Third, due to the applied metrics and algorithm of keyword extraction used in the LanaKey tool, the multi-word units are formed not just as mere frequent n-grams, but as syntactically well-formed word combinations, functioning in the corpus mainly as part of extracted phrases, which ensures the optimal choice of lexical units for modeling the domain through mapping to lexical dimensions. Hence, the choice of words or n-grams as features is intelligent and depends on the lexical unit functioning properties in the domain corpus. Thus, the domain-specific single- and multi-word units used as features ensure a complete and accurate semantic modeling of the domain by using monosemantic cliched phrases for constructing the vector space.

Below is an example of a sentence from the corpus that was tagged automatically using the ontolexicon-based platform and then manually verified, resulting in the 'gold standard' tagged corpus:

{Ten people}~A {were arrested}~P~RW~I {in Germany}~L~N {over}~O {suspicions}~I {they}~O {were}~B {planning}~K {a}~O {terrorist attack}~T, {report}~D {German}~N {news agency}~S {DPA}~S

As can be seen from the example, the tokens in the 'gold standard' corpus are enclosed in curly brackets and labeled with codes of semantic classes: *A* is the concept of 'Agent (perpetrator) of an Attack', *P* – 'Damage as a Consequence of an Attack', *RW* – 'Counter Terrorism Measures', etc. The full list of concepts, including 23 classes, and their codes are given in the Appendix (Table 5).

The 'gold standard' corpus consists of 1237 sentences of e-news on terrorist attacks for the period of 2019–2021. The corpus size is 22420 tokens (including single- and multi-word tokens). The complete vocabulary list of the corpus consists of 5819 different lexical single- or multi-word units. The list of the most frequent tokens related to the domain is given in Table 1.

The 'gold standard' corpus was preprocessed and reformatted to compile the dataset in the following order:

1. The corpus was split into tokens. Each token is explicated by a single- or a multi-word unit.
2. All tokens are lowercase to eliminate the graphical distinction between the same tokens at the beginning and in the middle of a sentence.
3. The set of labels related to the domain was binarized.

In particular, the labels of each token were transformed into a multi-dimensional label vector with zeros and ones at the corresponding positions, where "one" in a certain position means that this token is mapped onto the class coded in this position. The positions of the vector correspond to domain-relevant classes labeled with their codes. The label vector dimension for datasets is 23, which corresponds to the number of domain-relevant classes. The vector of zeros stands for the token belonging to common words that are not characteristic of the domain.

Sample data from the preprocessed corpus is shown in Fig. 1.

During this research, we experimented with different options for dataset preprocessing, which included:

1. Optional lemmatization with WordNetLemmatizer class from Natural language toolkit (NLTK) package[2]. One option is the case where each token in the dataset is lemmatized (see column 'lemma' in Fig. 1). For multi-word tokens, each word in the token is lemmatized individually and then concatenated into a string of lemmas. In this case, the model vocabulary consists of lemmatized tokens. Another option is to compile the vocabulary from the text forms used in the corpus.
2. Extracting sentence separators (periods, question marks) at the end of each sentence into a separate token or ignoring sentence separators during preprocessing the dataset (see token 55 in Fig. 1).
3. Stop words removal. The datasets were prepared in two forms: first, as a raw sequence of tokens as observed in the corpus (as shown in Fig. 1); second, as a sequence of tokens, from which stop words were removed (e.g., tokens 47, 49, 51 in Fig. 1 are excluded from the datasets with the stop words removed).

These optional preprocessing allowed further experimenting with the vocabulary size and the embedding layer, constructing embeddings based on the following principles:

[2] NLTK. Available at: https://www.nltk.org/ (accessed 30.03.2023).

	token	lemma	A	BW	C	CR	D	DA	E	EW	...	N	OW	P	RW	S	T	UW	X	Y	Z
45	At least	at least	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
46	two	two	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
47	other	other	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
48	persons	person	0	0	0	0	0	0	0	0	...	0	0	1	0	0	0	0	0	0	0
49	are	be	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
50	missing	missing	0	0	0	0	0	0	0	0	...	0	0	1	0	0	0	0	0	0	0
51	a	a	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
52	security source	security source	0	0	0	0	0	0	0	0	...	0	0	0	0	1	0	0	0	0	0
53	told	told	0	0	0	0	1	0	0	0	...	0	0	0	0	0	0	0	0	0	0
54	AFP	afp	0	0	0	0	0	0	0	0	...	0	0	0	0	1	0	0	0	0	0
55	.	[SEP]	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

Fig. 1. Sample data from the preprocessed dataset

Table 1. List of the most frequent token-label pairs in the corpus

Token	Label	Frequency
said	D	277
people	P	100
attack	T	88
died	P	58
explosion	T	45
told	D	39
terrorist attack	T	36
statement	S	34
injured	P	34
took place	R	33
city	L	31
wounded	P	31
suspect	A, I	27
incident	T	27

1) considering word distribution either within a sentence or across sentences. In the first case, the batches of tokens fed into the model's embedding layer are formed from the tokens of a single sentence only; in the latter case, the batches are created as n-grams, which include neighboring words both within the current sentence and from the adjacent sentence, that is, a batch might consist of the last k tokens from the previous sentence combined with the first n-k tokens from the following sentence in the text (in this approach, separation tokens are ignored);

2) taking or not taking into account the function words in the context of notional lexical units in the embeddings;

3) experimenting with various label vector representations.

Thus, the materials of the research consists of eight datasets, each of which was preprocessed according to the first three above-mentioned steps. Various features of the datasets are shown in Table 2.

The distribution of tokens labeled by 23 domain-relevant classes in the datasets is shown in Fig. 2. This distribution is the same for all eight datasets as separators and stop words, consisting mostly of function words, are semantically irrelevant to the domain, and consequently, are labeled as belonging to the class *O* (*Other*). Therefore, the dataset modifications involving changes in stop words, separators and lemmatization parameters do not affect the distribution.

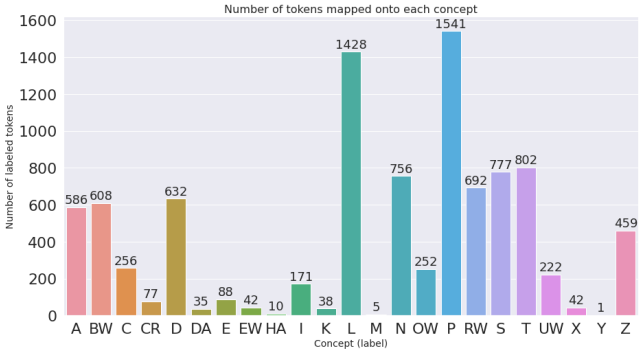

Fig. 2. Label distribution in the corpus

Table 2. Dataset features used in the experiments

Dataset No.	Tokens are lemmatized	Separators are taken into account	Stop words are eliminated
Dataset 1	Yes	No	Yes
Dataset 2	No	No	Yes
Dataset 3	Yes	Yes	Yes
Dataset 4	No	Yes	Yes
Dataset 5	Yes	No	No
Dataset 6	No	No	No
Dataset 7	Yes	Yes	No
Dataset 8	No	Yes	No

As can be seen from the example above, some tokens in the 'gold standard' corpus are tagged with multiple ontology classes, which is due to conceptual syncretism (see [15] for more details on conceptual syncretism). For example, the token '*in Germany*' is labeled as activating the concepts *Location* (*L*) and *Nation* (*N*). Though the vast majority of tokens in the 'gold standard' corpus are unambiguous and labeled with one tag, some tokens are mapped to multitags, consisting of two or, in rare cases, three or four tags (see Fig. 3).

The number of tokens in Datasets 1 through 4 (cases with stop words removed) that are labeled in the 'gold standard' corpus with one and more tags is shown in Fig. 3. The other four datasets differ only in the height of the first bar of the histogram. This bar shows the number of tokens that cannot be assigned to domain-relevant ontology classes, e.g., function words or common vocabulary. Such tokens are labeled with the vector of zeros in the datasets.

If we discard the irrelevant tokens reflected by the first bar, tokens in the datasets are labeled predominantly with one domain-relevant tag. However, as the histogram in Fig. 3 shows, there are cases of double and triple class associations. Thus, the problem of concept tagging turns into a problem of multi-label classification: each token has to be labeled with a vector of zeros and ones in the corresponding positions.

Deep learning model for concept tagging

Our approach consists of constructing a classification model that takes as input a sequence of embeddings for consecutive tokens (lexical units), each embedding being computed by token distribution in the 'gold standard' corpus. We trained the models with a window of five consecutive tokens. The embedding sequences were fed to a series of Bi-LSTM layers, then to a hidden dense layer and finally to a dense classifier, assigning one or more classes to the middle token. Each layer is followed by dropouts to prevent overfitting and batch normalization, which helps improve efficiency of the model. The seed model architecture is given in Fig. 4. The number of trainable parameters in the model is 17349655.

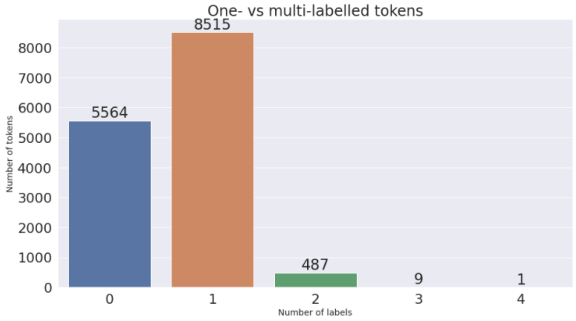

Fig. 3. Distribution of tokens with one and more labels (Datasets 1 through 4)

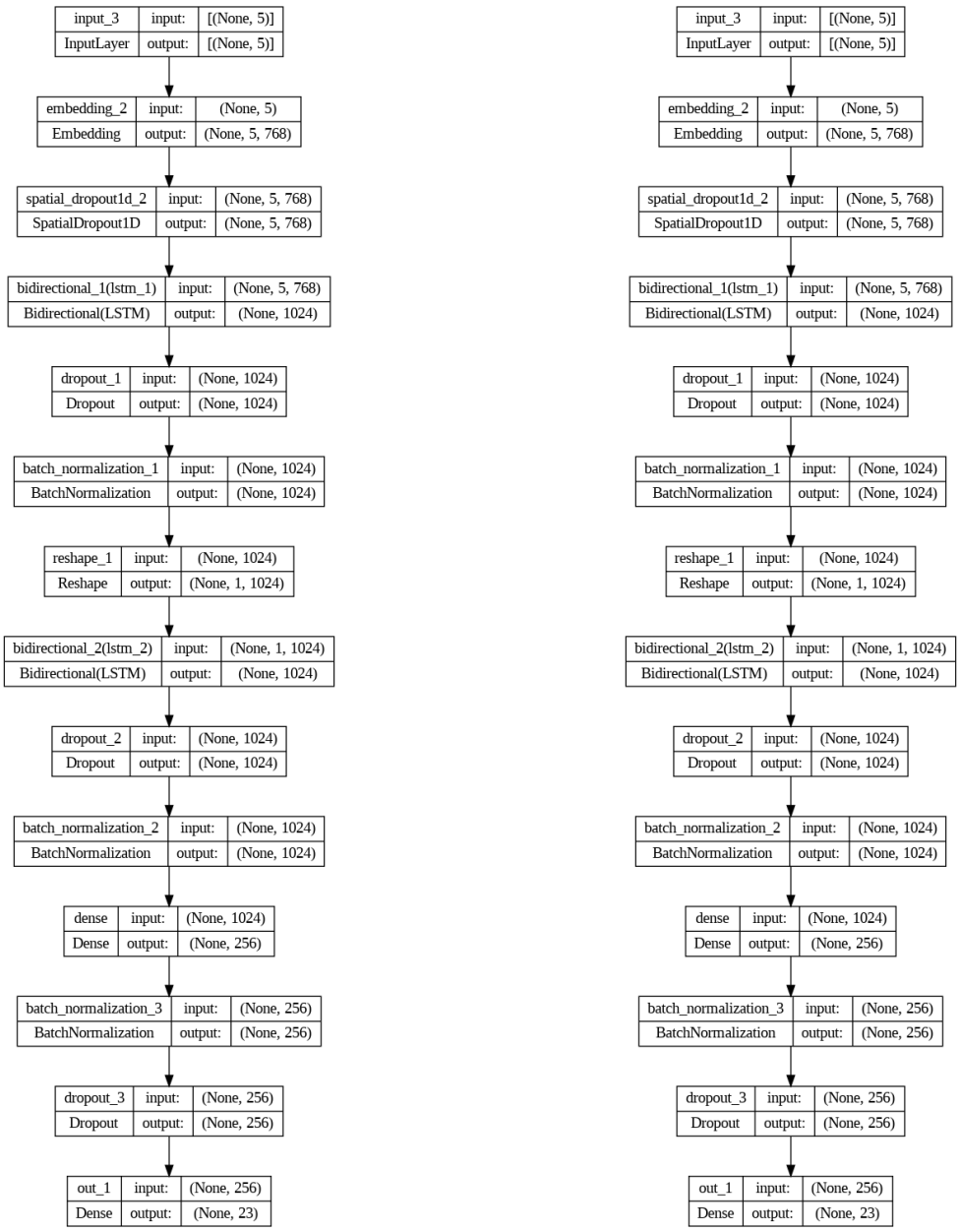

Fig. 4. Neural network architecture for the concept tagging task

The output of this model is a 23-dimensional array, its dimensions representing probabilities for each of the labels marking domain-relevant classes in the middle token.

The datasets were split into training, validation and test data subsets, with the validation portion being 0.2 and the test portion being also 0.2. The models for each dataset were trained for 25 epochs and then evaluated on the test datasets using the metrics described below to determine the best dataset preprocessing option. The best dataset was then evaluated for the quality of each class prediction. The results are presented in the following section.

Results

As we have shown, the problem of concept tagging is a multi-label classification problem. Keeping that in mind, we found it useful to evaluate the model performance on the test datasets based on aggregated metrics, including micro-, macro- and weighted average F1-score for each dataset and then F1-score for each class individually. The F1-score calculation is based on the precision and recall, which in turn are calculated based on the number of true positive, true negative, false positive and false negative predictions of the model. We calculated the F1-score using equations (1)–(3):

$$F1\text{-}score = 2 \cdot precision \cdot recall \big/ \left(precision + recall \right), \quad (1)$$

$$precision = TP \big/ \left(TP + FP \right), \quad (2)$$

$$recall = TP \big/ \left(TP + FN \right), \quad (3)$$

where *F1-score* is the model-performance metric calculated as the harmonic mean of precision and recall; *precision* is a metric that measures the model quality in terms of its accuracy in predicting positives; *recall* is a metric that measures the model quality in terms of its sensitivity to positives in the dataset; *TP* is the number of true positive predictions; *FP* is the number of false positive predictions; *FN* is the number of false negative predictions of the model.

The micro averaged F1-score is calculated based on the sum of true positive, false positive and false negative predictions across all the classes, and then the global precision, recall and F1-score are calculated based on these values. The macro averaged F1-score is calculated as the arithmetic mean of all F1-scores for each class. The weighted average is calculated as the mean of all F1-scores for each class weighted by the class support, i.e. the proportion of actual occurrences of the class in all domain-relevant tokens in the test dataset. The metrics calculation results are given in Table 3.

Dataset 3, which includes texts with lemmatized tokens and removed stop words, as well as with sentence borders taken into account, turned out to be the best based on F1-score metrics. For this dataset, we also calculated confusion matrices and areas under the precision-recall curve for each class to determine the thresholds. The threshold for each class was calculated based on the threshold that produces maximum F1-score for each class. The precision-recall curves for each class are shown in Fig. 5.

The selected thresholds for each class are marked with large dots on the curves, which correspond to the per-class precision-recall tradeoffs maximizing F1-score metrics. The legends show areas under the curve and the thresholds calculated for each class based on F1-score maximization.

The model-produced predictions for test dataset evaluated by per-class precision, recall and F1-score metrics are shown in Table 4. Classes labeled as *M* and *Y* are omitted as they were found to be unsupported in the test dataset. Classes are sorted by F1-score value in descending order.

The data demonstrate low recall against the background of often rather high precision for classes that are hardly present in the training dataset (e.g., *Direction of Attack* (DA), *Adversary's Plans* (K), *Goal of Attack* (T) etc.). Classes representatively supported in the training dataset differ in the values of their scores. This may be due to monosemy or, conversely, ambiguity of tokens: classes represented in the corpus by

tokens that are ambiguously related to different classes in different contexts (*Counter-Terrorism* (RW), *Consequences-Damage* (P), etc.) have lower precision and recall values, which indicates insufficient sensitivity of the model due to the 'noise' produced by multiple references the token to concepts in the training data.

Table 3. Aggregate metrics of learning for test datasets based on different dataset preprocessing (The largest F1-scores in the column are shown in bold)

Dataset No.	Micro averaged			Macro averaged			Weighted averaged		
	precision	recall	F1	precision	recall	F1	precision	recall	F1
Dataset 1	.42	.47	.44	.51	.30	.37	.71	.47	.56
Dataset 2	.40	.46	.43	.44	-27	.33	.72	.46	.56
Dataset 3	.71	.59	**.64**	.73	.47	**.56**	.85	.60	**.70**
Dataset 4	.28	.42	.34	.47	.28	.34	.73	.42	.53
Dataset 5	.24	.44	.31	.42	.26	.31	.69	.44	.53
Dataset 6	.23	.44	.30	.40	.24	.30	.70	.44	.53
Dataset 7	.57	.59	.58	.64	.42	.50	.85	.59	.69
Dataset 8	.19	.43	.26	.42	.24	.30	.67	.43	.52

Errors in classification are likely due to ambiguous tokens, such as the token '*Arab people*' that could be referred to an *Agent* (A) in some contexts and to an *Object of Attack* (Z) in others. Another cause of errors may be due to unknown tokens that appear in the test dataset and are absent from the training set.

Discussion

The results obtained concern the solution of the problem of token tagging on a semantic basis. This task quite often arises in state-of-the-art research related to textual input classification based on topic modeling, identification of web-user's interest, authors' demographic parameters including age, gender, education, etc., see, for example, [4, 7–9, 14]. At the same time, the machine learning approach plays an essential role in solving problems related to terrorist activity (also considered in this paper as a practical application of CA), including event classification in terrorism domain [4], topic modeling to predict terrorist motives [2], data analysis to predict factors contributing to the growth of terrorism [1], prediction of possible terrorist attacks [3], prediction of a terrorist attack related parameters (such as weapon type, attack type, etc.) based on studying features from the Global Terrorism Database [10] and other techniques aimed at preventing violence and security threats. The presented solutions based on learning from natural language texts provide F1-score for different models up to 78%.

This paper considers the problem of finer-grained semantic classification applied to domain-related words and phrases, which would allow CA of terrorism domain e-news by tracking the combinations and frequencies of labels in a text sequence. This task is more challenging than that of entire text categorization, since several labels need to be recognized in the same context and, in addition, they need to be arranged in a sequence, i.e., associated with a specific token.

The approach proposed here is to compile the vocabulary of domain-relevant *multi-word* units coded in the input sequences as a single token. These tokens function as a unit for constructing embeddings in the Bi-LSTM-based classification model, which is a key distinction of our model. This approach allows us to refine the semantics of the units used to model the domain, and thus contributes to improving the accuracy of the model.

The research results show that the hypothesis about the possibility of improving the metrics through preprocessing the corpus did add to weighted averaged F1-score metrics, which varied from 53% for

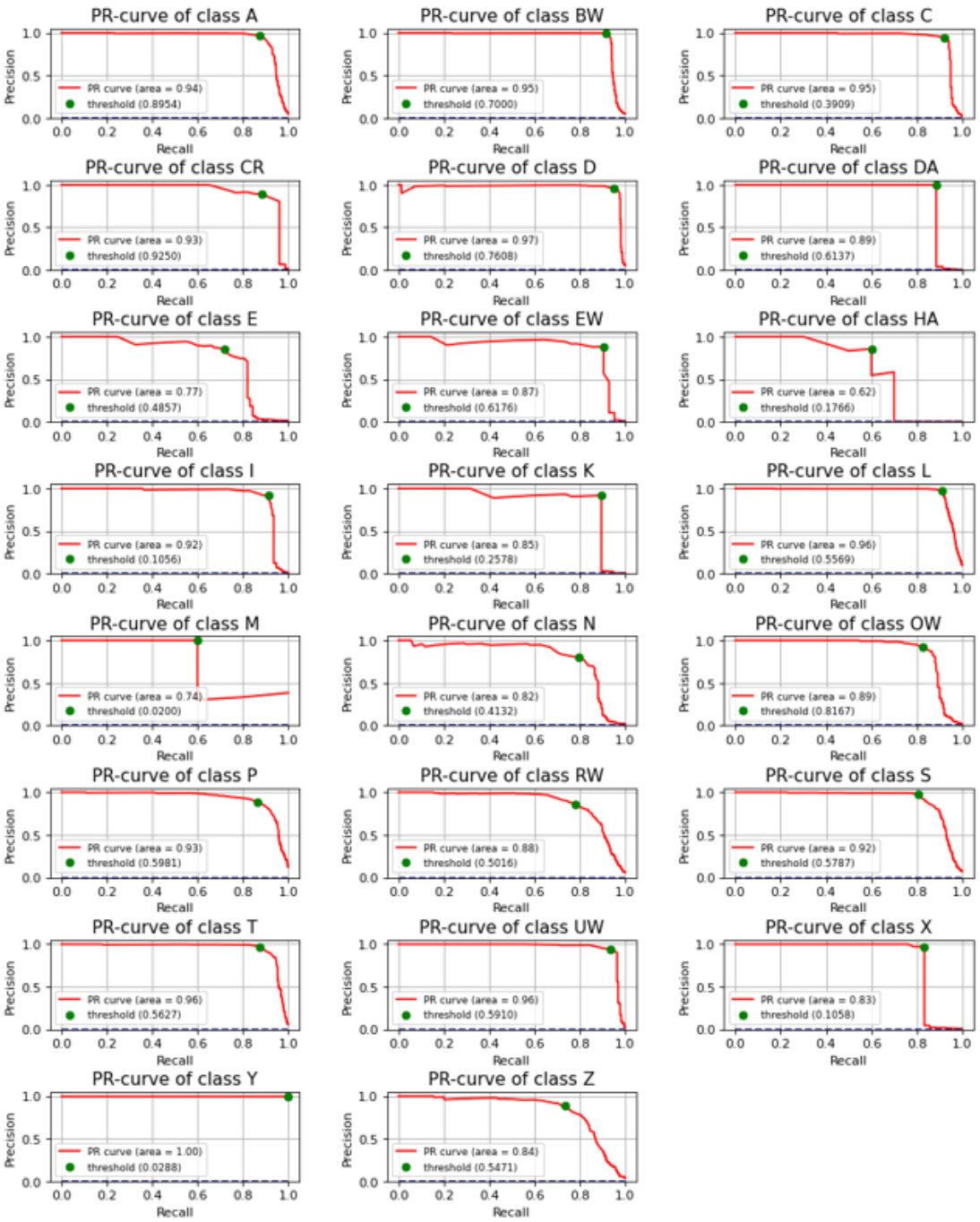

Fig. 5. Precision-recall curves and threshold values for Dataset 3

the minimally preprocessed data (Dataset 6) to 70% for *data preprocessed by stop words removal, lemmatization and taking into account separators* (Dataset 3). In case of per-class predictions, the highest predictive ability was achieved for the class *Declaration* (F1-score of 88%), which could be explained by the following factors:

1) the training set of class examples was supported by a sufficiently large set of samples (510 units) in the corpus;

2) the vocabulary size of the class is comparatively lower than that of other classes and includes the most frequent lexical units in the corpus;

Table 4. Per-class model prediction evaluation

Concept	Code	Test Dataset				Support in training dataset
		Precision	Recall	F1-score	Support	
Declaration	D	0.93	0.84	0.88	123	510
Terrorist Organization	UW	0.97	0.78	0.86	40	182
Time	BW	0.99	0.69	0.81	127	481
Means of Attack	C	0.92	0.69	0.79	49	207
Type of Attack	T	0.89	0.69	0.78	149	653
Consequences-Damages	P	0.86	0.69	0.77	310	1231
Agent	A	0.96	0.61	0.74	109	477
Cause	EW	0.88	0.64	0.74	11	31
Location	L	0.97	0.60	0.74	284	1144
Claim Responsibility	CR	0.79	0.69	0.73	16	61
Assumption	I	0.88	0.54	0.67	28	143
Direction of Attack	DA	1.00	0.44	0.62	9	26
Adversary's Plans	K	1.00	0.38	0.55	8	30
Counter-Terrorism	RW	0.69	0.46	0.55	134	558
Source	S	0.70	0.45	0.55	139	638
Goal of Attack	X	1.00	0.33	0.50	12	30
Nation	N	0.59	0.37	0.46	51	158
Object of Attack (Target)	Z	0.54	0.40	0.46	94	365
Other Terrorist Activities	OW	0.77	0.26	0.38	39	213
Threats	E	0.44	0.20	0.28	20	69
Have Means of Attack	HA	0.00	0.00	0.00	3	7

3) units belonging to the class tend to be unambiguous in the corpus, i.e. units are labeled predominantly with one tag only.

The lowest prediction results were observed, as expected, for the classes that are of low frequency in the corpus, as well as for the classes, whose units vary their association to a conceptual class in different context.

Generally, the experimental results show quality comparable to the results obtained for the task of the whole text classification and, as a consequence, provide the possibility to finer-grained per-unit classification, which should provide data for quantitative CA and as well as room for further research.

Conclusion

This paper presents the results on experimenting with textual dataset preprocessing to study a model for concept tagging of text tokens. The proposed approach treats the dataset as a set of single- and multi-word tokens that may be labeled with one or more domain-relevant conceptual classes. Experiments dealt with varying the options of token preprocessing, including optional lemmatization, stop words removal and taking into account sentence separators in the textual input.

Experimental results show that a dataset with lemmatized words, removed stop words and consideration of sentence separators provides the best F1-score values, when assessed on a micro-, macro- and weighted average basis.

The practical value of this research lies in the possibility of applying proposed approach to the preparation of datasets, when tagging large corpora in natural languages. Tagged corpora may further serve as a basis for CA of discourses, which is in demand in various domains, including counter-terrorism security, politics, business, economics, mass media, etc.

Appendix

Table 5. List of conceptual classes used in corpus annotation (adapted from [13])

Concept	Code	Definition	Lexical examples
ADVERSARY'S PLANS	K	Intended activities of a terrorist or a terrorist group.	alleged terrorist attack plot, planning a terrorist attack, preparing, plotting
AGENT	A	The perpetrator of the attack.	violent extremist, terrorist group, fighter, terrorism suspect
ASSUMPTION	I	Assumptions of "good guys" about a probable terrorist group behind the attack or a suspect.	suspect, being suspected, reportedly, apparently
CAUSE	EW	To make smth happen	resulting in, causing, as a result of, as a result of two explosions
CHARACTER OF ATTACK	M	The concept indicates whether the victims of the attack were numerous or one person was the only target.	deadliest, large-scale, macabre, powerful
CLAIM RESPONSIBILITY	CR	To claim responsibility for an attack.	claiming responsibility for the attack, claiming, pleading guilty, responsible
CONSEQUENCES DAMAGE	P	All negative outcomes of the terrorist attack, such as human victims, destroyed objects, terrorists' destiny, and the condition of those.	victim, being killed, being injured, resulting in no injuries
CONSEQUENCES RECOVERY	Y	Actions aimed at repairing a physical and moral damage of the attack, e.g., condolences, etc.	quick recovery
COUNTER-TERRORISM	RW	Actions aimed at preventing terror attacks or deliver justice to terrorists	heightened security cordon, detective, detaining, investigation
DECLARATION	D	To say, to declare, to announce (the concept is normally linked to verbs and adverbial phrases that mean the transfer of information).	saying, adding, telling, reporting
DIRECTION OF ATTACK	DA	To target smth or smb	against, targeting, directly targeting
GOAL OF ATTACK	X	The goal terrorists are trying to achieve by committing the attack. It can also be used to indicate the rea-son for the attack as sometimes it is hard to distinguish between them.	sowing seeds of discord, weakening the offensive, stoking sectarian tensions, diverting attention from Mosul
HAVE MEANS OF ATTACK	HA	To have a weapon or a weapon-like object (the concept is normally linked to verbal phrases that mean the process of application of MEANS OF ATTACK).	being laden, being armed with, wielding, being filled with
LOCATION	L	The country, region, city, district, or geographical entity where the attack took place.	Turkey, city of Kobani, Iraqi capital of Baghdad, around mosques
MEANS OF ATTACK	C	The weapons or weapon-like objects (e. g., a truck) used to commit the attack, also functional weapon parts, such as explosives, bullets, etc.	land mine, knife, suicide car bomb, assault rifle
THREATS	E	Terrorist's actions related to threatening, causing fear in victims, and their aftereffects	threat, start of the offensive, mind control, fresh spate of violence
NATION	N	The origin of terrorist and victims; it should not be confused with LOCATION, which only covers the places where particular attacks were committed.	Franco-Moroccan, Australian, Palestinian, Algerian
OBJECT OF ATTACK (TARGET)	Z	The animate or inanimate object the attack is directed to, which is hurt or damaged in the attack.	church, shopper, crowded street, government official

End of Table 5

Concept	Code	Definition	Lexical examples
OTHER TERRORIST ACTIVITIES	OW	Types of terrorist activities that are not literary terror attacks, e.g., terrorism financing, recruiting, involvement in war conflicts, etc., but appear sporadically in terrorism domain e-news and are therefore considered relevant.	suspicious activity, material support, Islamic State sympathizers, screaming Allahu Akbar
SOURCE	S	The sources of the message about the attack, such as newspapers, TV channels, news agencies, or authorities.	Daily Mail, Amaq news agency, witness, regional governor spokesman
TERRORIST ORGANIZATION	UW	The organization responsible for the attack or any organization mentioned in the text.	Islamic State terror group, Daesh, Taliban, al-Qaida
TIME	BW	The time and date of the attack.	this year, Monday, during the New Year holidays, for at least four hours
TYPE OF ATTACK	T	The type of attack, such as an explosion, kidnapping, arson, etc.	terrorist attack, explosion, suicide bomber

Given the obtained results, further research can be related to the assessment of the model's performance depending on further dataset preprocessing options, such as varying vocabulary size of the input by reducing its frequency or by combining semantically close tokens based on a thesaurus. Another prospect concerns experimenting with neural network architectures to improve the quality indicators of the model. We also consider the possibility of turning to pre-trained natural language processing models to assess their potential for the task at hand.

REFERENCES

[1] **Agarwal P., Sharma M., Chandra S.,** Comparison of Machine Learning Approaches in the Prediction of Terrorist Attacks, 2019 Twelfth International Conference on Contemporary Computing (IC3), (2019) pp. 1–7. DOI: 10.1109/IC3.2019.8844904

[2] **Bridgelall R.,** An Application of Natural Language Processing to Classify What Terrorists Say They Want, Social Sciences, 11 (1) (2022) 1–15. DOI: 10.3390/socsci11010023

[3] **Huamaní E.L., Alva-Mantari A., Roman-Gonzalez A.,** Machine Learning Techniques to Visualize and Predict Terrorist Attacks Worldwide using the Global Terrorism Database, International Journal of Advanced Computer Science and Applications, 11 (4) (2020) 562–570. DOI: 10.14569/ijacsa.2020.0110474

[4] **Inyaem U., Meesad P., Haruechaiyasak C., Tran D.,** Terrorism Event Classification using Fuzzy Inference System, International Journal of Computer Science and Information Security, 7 (3) (2010) 247–256. DOI: 10.48550/arXiv.1004.1772

[5] **Karpovich S.N.,** Multi-Label Classification of Text Documents using Probabilistic Topic Model ml-PLSI, SPIIRAS Proceedings, 4 (47) (2016) 92–104. DOI: 10.15622/sp.47.5

[6] **Krippendorff K.,** (1980) Content Analysis: An introduction to its methodology. London: SAGE Publications, Inc. DOI: 10.4135/9781071878781

[7] **Mitrofanova O., Sampetova V., Mamaev I., Moskvina A., Sukharev K.,** Topic Modelling of the Russian Corpus of Pikabu Posts: Author-Topic Distribution and Topic Labelling, Intelligent Memory Systems, (2020) pp. 115–130.

[8] **Mustafa G., Usman M., Yu L., Tanvir Afzal M., Sulaiman M., Shahid A.,** Multi-label classification of research articles using Word2Vec and identification of similarity threshold, Scientific Reports, 11 (2021) 21900. DOI: 10.1038/s41598-021-01460-7

[9] **Oliseenko V., Tulupyeva T.,** Neural Network Approach in the Task of Multi-label Classification of User Posts in Online Social Networks, 2021 XXIV International Conference on Soft Computing and Measurements (SCM), (2021) pp. 46–48. DOI: 10.1109/SCM52931.2021.9507148

[10] **Saidi F., Trabelsi Z.,** A hybrid deep-learning based framework for future terrorist activities modeling and prediction, Egyptian Informatics Journal, 23 (3) (2022) 437–446. DOI: 10.1016/j.eij.2022.04.001

[11] **Sheremetyeva S.,** On Extracting Multi-word NP Terminology for MT, Proceedings of the 13th Annual Conference of the European Association for Machine Translation, (2009) pp. 205–212.

[12] **Sheremetyeva S., Babina O.,** A Platform for Knowledge Assisted Conceptual Annotation of Multilingual Texts, Bulletin of the South Ural State University. Series Linguistics, 17 (4) (2020) 53–60. (In Russ.). DOI: 10.14529/ling200409

[13] **Sheremetyeva S., Zinoveva A.,** Ontological Analysis of E-News: A Case for Terrorism Domain, Proceedings of the 14th International Conference on Interactive Systems, (2019) pp. 130–141.

[14] **Uteuov A.,** Topic model for online communities' interests prediction, Procedia Computer Science, 156 (2019) 204–213. DOI: 10.1016/j.procs.2019.08.196

[15] **Zinoveva A.,** On Resolving Conceptual Ambiguity in an English Terrorism E-news Corpus, Proceedings of the International Conference "Internet and Modern Society" (IMS-2020), (2021) pp. 205–215.

INFORMATION ABOUT AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Olga I. Babina
Бабина Ольга Ивановна
E-mail: babinaoi@susu.ru
https://orcid.org/0000-0002-1733-6075

Anastasia Yu. Zinoveva
Зиновьева Анастасия Юрьевна
E-mail: zinovevaai@susu.ru
https://orcid.org/0000-0002-7658-7376

Ekaterina D. Nerucheva
Неручева Екатерина Дмитриевна
E-mail: neruchevaed@susu.ru

Поступила: 19.04.2024; Одобрена: 30.09.2024; Принята: 02.10.2024.
Submitted: 19.04.2024; Approved: 30.09.2024; Accepted: 02.10.2024.