Comparison of methods for automatic classification of Russian-language texts

E.N. Karuna, P.V. Sokolov

Saint Petersburg Electrotechnical University “LETI”, 197376 St. Petersburg, Russian Federation

Email: zhenya.karuna@mail.ru

Abstract. In this paper, a comparative analysis of methods for automatic classification of text information using various machine learning algorithms based on artificial neural networks is carried out. The paper presents results of the text classification considering various ways of a vector space formation, such as bag-of-words, n-gram, word embedding, and various architectures of neural networks. The research is based on the text corpus in Russian.

1. Introduction

A large amount of textual data stored both in open sources and in closed databases of enterprises requires high-quality tools to analyze this data, and in particular, to automatically categorize this data. One of the problems of the text classification tasks using machine learning is that textual data cannot always be uniquely attributed to one of the specific classes by the quantitative composition of all words in the document. The solution to these problems is the development of a classification system that takes into account the mutual arrangement of words within the text. At the moment, a fairly large number of materials are available on the study of systems for the text information classification, but at the same time, there are practically no studies devoted to systems for automatic classification of data consisting of a Russian-language set of texts, while the features of the Russian language grammar can significantly affect the results of such systems. In the present report, a comparative analysis of the automatic classification methods is carried out, and conclusions on the results of the operation of the systems under study for the Russian-language data corpus are given.

Algorithms for the textual data classification carry out automatic determination of the thematic category of the text. The subset of automatic classification tasks includes such tasks as: determining the emotional coloring of statements, spam filtering, identifying authors of texts, sorting letters, messages, news, and other tasks. The solution to the classification problem can be divided into two main stages: text data preprocessing and machine learning algorithm. The first stage is necessary to transform the original text into a set of features, which is a numeric vector or matrix. At the second stage, one of the machine learning algorithms, which can determine a text thematic category using the set of text features in the form of a numeric vector or matrix, is implemented.

2. Text data preprocessing

At this stage, it is necessary to perform a number of tasks: removing non-alphabetic characters from a text, splitting a text into a list of tokens, removing stop words, performing the operation of reducing words to their word stem. Removing non-alphabetic characters and stop words is a standard procedure
allowing clearing the text from unnecessary elements that have little effect on the general theme of the text. To reduce a word to its stem, one of two methods is used: stemming and lemmatization, and, in the first case, a rough reduction of a word ending is performed according to a certain algorithm, in the second case, the word is reduced to its base form, in accordance with a language grammar.

The texts subject matter determination requires a pre-marked corpus of data that will be used to train the algorithm. In this work, a corpus of more than 102 thousand news articles of the Internet edition "Lenta.ru" in Russian is used. The news articles are divided into 10 categories. Each news article, in addition to its body text and headline, has a topic tag, and each topic is a particular category. The entire collection of texts is divided into 2 groups: training and test, with the size of the training group equals 80% of the total number of texts. Research is carried out with some common neural network models [1]. Below, some of them are considered separately.

3. Models of neural networks

3.1. Feedforward neural network

In artificial neural networks of the FFNN (Feedforward neural network) type, the signal is transmitted strictly from the input layer to the output layer, and the connection between these layers is a fully connected network. For feedforward networks, a single numeric vector is fed into the input, and each vector is responsible for one text from the data corpus. The most common way to compose an input vector of text is BOW (bag-of-words). This method consists in creating a single dictionary consisting of a list of tokens, and the most relevant features that are used to compose vectors are identified on the basis of the created dictionary. Typically, one feature is a single word or phrase. The vector should consist of numerical values characterizing the quantitative measure of the presence of words from the dictionary in the text. In this work, two statistical measure types are used to calculate values of the TF and TF-IDF vectors, equations 1 and 2, respectively:

$$tf_i = \frac{n_{id}}{\sum_{j=1}^{imax} n_{jd}}$$  (1)

$$tfidf_i = tf_i \times \ln \left( \frac{|D|}{|d \mid i \in d|} \right)$$  (2)

where $tf_i$ is the frequency of using the word $i$ in the document $d$, $n_{id}$ is the number of words $i$ in the document $d$, $imax$ is the dictionary size, $\sum n_{jd}$ – is the number of words in the document $d$. $D$ is the set of all documents in the corpus, $|D|$ is the number of documents in the collection, $|d|_{i \in d|} \mid$ is the number of documents in which the word $i$ occurs.

The dimensions of vectors are limited and user-defined. The studies of the classification accuracy are carried out using various statistical measures of vector formation, various sizes of the input vector, and various ways of defining the word stem. Also, the results of classification when creating the dictionary from both single words (unigrams) and combinations of 2 or 3 words (bigrams and trigrams, respectively) are obtained.

An example of a feedforward neural network that allows working with the bag-of-words model is shown in Figure 1.

Chapter 4 additionally presents results of the thematic analysis of textual data using different methods for operations of defining the word stem and calculating vector elements for the bag-of-words model.

For the following experiments, an approach known as vector representation of words, in which each word is a separate vector, is used [2]. The advantage of this approach is that it allows using machine learning algorithms that take into account not only the quantitative composition of all words in the text, but also the relative position of words within the entire text. Various methods are used to form a vector space containing a set of vectors for a large number of known words in the language. It is important to note that there are ready-to-use pre-trained data samples that contain a large vocabulary
of terms and a set of vectors for each word. In this work, a pre-trained model of vector representation of words is used.

### 3.2. Convolutional neural network

The CNN (convolutional neural network) model is actively used to solve pattern recognition problems. When solving such problems, the input data is a fixed size matrix. The matrix can be built by concatenating the vectors obtained using the vector word representation algorithm. The CNN model performs alternating convolution operations on a numerical matrix for activating neurons and subsampling operations to reduce the matrix dimension. As a result, the matrix is transformed into a vector, which is fed to the feedforward neural network input.

In the present work, the input text matrix size is \((m \times l)\), where \(l\) is the dimension of the vector space of words, which depends on the chosen method of vector representation of words, and \(m\) is the maximum length of the sentence that is fed to the convolutional neural network input. Since the convolutional neural network requires a matrix of a fixed size as its input, it is necessary to determine in advance the number of words from the text that will be converted into vectors and used in the classification algorithm. If the value \(m\) is greater than the number of words in the text, then the rest of the matrix can be filled with zeros.

The convolution kernel, by which the input matrix is multiplied element by element to perform the convolution operation, must have a width equal to the dimension of the vector \(l\), the length of the convolution kernel must be selected manually. The results of the convolution operation are vectors, and the length and number of vectors depend on the number of convolution kernels and the convolution kernel length. The subsampling operation can be applied to these vectors to feed them to the input of a fully connected network.

Out of many experiments, the best results of a convolutional neural network are obtained using the scheme shown in Figure 2. In this model, 5 different types of convolution kernels are used on the text matrix input, with their sizes from \(1 \times 300\) to \(5 \times 300\) elements, thus allowing taking into account the presence of both specific words in the text and combinations of several words. To reduce the space dimensionality, a subsampling operation is performed on each of the obtained vectors, and then the resulting vector is connected to the fully connected network, which determines the text category using the defined sets of features.
3.3. Recurrent neural network

There is a wide variety of algorithms based on recurrent neural networks, and at the moment one of the most common is the LSTM network (long short-term memory) with BiLSTM (bidirectional long short-term memory) layer, which is a neural network with a recurrent layer consisting of LSTM blocks passing the input data array in the forward and backward directions [3]. A special feature of recurrent neural networks is that connection between neurons is transmitted over time. So, when sending several data sets to the network, the network output is determined not only by the current input, but also by the sent earlier data. The advantage of LSTM networks is that they allow storage of the acquired information for a long time.

![Convolutional neural network model for textual data classification](image)

Figure 2. Convolutional neural network model for textual data classification

3.3. Recurrent neural network

![Recurrent neural network with BiLSTM layer](image)

Figure 3. Recurrent neural network with BiLSTM layer

Figure 3 shows time evolution of the BiLSTM network used in the study. In this neural network each row of the input matrix goes to the corresponding LSTM-block, and all LSTM-blocks are then connected to the fully connected network. The research results for LSTM and BiLSTM neural networks are presented in Table 2.

4. Results of textual data classification

The research result is the building of the neural networks models presented in the previous chapter and the comparison of the classification results of the text corpus. For feedforward networks, an additional comparison of the classification accuracy when using different implementations of the word stem
defining operation and different calculating methods of vector elements for the bag-of-words model is made. The results are shown in Table 1.

Table 1. Comparison of different methods of word stem definition and methods of vector values calculating for the BOW model

| Word stem definition method | Vector elements formation method | Accuracy |
|----------------------------|---------------------------------|----------|
| stemming                   | TF                              | 0,853    |
| lemmatization              | TF                              | 0,859    |
| stemming                   | TF-IDF                          | 0,861    |
| lemmatization              | TF-IDF                          | 0,872    |

According to the Table 1, the best classification accuracy can be achieved by using the lemmatization operation with the TF-IDF statistical measure to count the vector elements. The chosen statistical measure and the stem definition method affect the speed of the algorithm during the preprocessing of texts, but do not affect the network learning rate in any way, which is a more important indicator, thus the comparison of the learning rate is not given in Table 1. Further, these methods for the BOW model are used. Comparison of classification algorithms when using unigrams, bigrams and trigrams is shown in Table 2.

When constructing a convolutional neural network, the architecture of the model shown in Figure 2 has been used. This model provides fairly high classification accuracy. But, it should be noted, this model can be improved, since it has a large number of adjustable parameters. With the correct selection of these parameters, it is possible to achieve higher classification accuracy. The results of the classification accuracy comparison for all algorithms are presented in Table 2.

Table 2. Comparison of classification algorithms models based on neural networks

| Model                        | Accuracy | Training time, s |
|------------------------------|----------|------------------|
| FFNN – BOW – TF-IDF – 1-gramm| 0,872    | 1807             |
| FFNN – BOW – TF-IDF – 2-gramm| 0,891    | 1932             |
| FFNN – BOW – TF-IDF – 3-gramm| 0,868    | 1865             |
| CNN                          | 0,894    | 39563            |
| BiLSTM                       | 0,905    | 15892            |
| LSTM                         | 0,901    | 12685            |

As it is shown in Table 2, the best classification accuracy is provided by a neural network algorithm with a bidirectional LSTM layer, while the difference between this algorithm and the other is insignificant. This algorithm has training time significantly longer than the training time of any algorithm configuration of feedforward network. In addition to high speed, bag-of-words models are quite accurate and simple to implement, without any pre-trained models requirements. A neural network based on a convolutional neural network has a fairly high accuracy, comparable to recurrent networks, but it requires a very long training time.

Figure 4 compares thematic analysis algorithms for each topic. In this figure it can be seen which topics cause difficulties for the algorithms and how well the models cope with them.

The most difficult topic for classification models is the topic "Business". This is due to the fact that this topic on the words composition and connection strongly overlaps with the topic "Economics". At the same time, all bag-of-words based algorithms show very similar accuracy.
5. Conclusion
In this work, some common algorithms for automatic classification of textual data on the news corpus of Russian-language texts are demonstrated.

The results show that bag-of-words based algorithms are easy to configure and they also have a high learning rate. But from the very structure of the algorithms it is clear that they have low accuracy when working with texts with complex semantics. These algorithms are best suited for situations when it is necessary to quickly complete the classification task of texts corpus that do not overlap thematically.

The algorithm based on a recurrent neural network with the LSTM layer has the best accuracy. Convolutional neural networks also demonstrate a fairly high level of classification accuracy. The difference between these models and bag-of-words models is that they are able to take into account the semantic relationship between words. Models using bigrams and trigrams demonstrate something similar, but their results turned out to be worse than other models results. It is worth noticing the high potential of the convolutional neural network, since it has a large set of adjustable parameters and with the correct selection of these parameters, it is possible to achieve higher classification accuracy.

This study shows the difference between some common models of text data analysis and demonstrates their performance indicators based on solving the problem of text classification in the Russian-language data corpus. Further research can be aimed at improving the existing algorithms of text data analysis using the results obtained for solving various problems of natural language processing.

References
[1] Kowsari K, Jafari Meimandi K, Heidarysafa M, Mendu S, Barnes L and Brown D 2019 Text Classification Algorithms: A Survey Information 10 150
[2] Bin Wang, Angela Wang, Fenxiao Chen, Yuncheng Wang and C-C Jay Kuo 2019 Evaluating Word Embedding Models: Methods and Experimental Results. APSIPA Transactions on Signal and Information Processing 8
[3] Tao Chen, Ruifeng Xu, Yulan He and Xuan Wang 2017 Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN Expert Systems with Applications 72221-230