APPLICATION OF NEURAL NETWORK ALGORITHMS AND NAIVE BAYES FOR TEXT CLASSIFICATION

[1]VADYM S. YAREMENKO, [2]WALERY S. ROGOZA, [3]VLADYSLAV I. SPITKOVSKYI

[1]PhD Student, NTUU "Igor Sikorsky KPI", Institute for Applied System Analysis, Department of the System Design, Kyiv, Ukraine
[2]Professor, NTUU "Igor Sikorsky KPI", Institute for Applied System Analysis, Department of the System Design, Kyiv, Ukraine
[3]Student, NTUU "Igor Sikorsky KPI", Institute for Applied System Analysis, Department of the System Design, Kyiv, Ukraine

E-mail: [1]yaremenko.vs@gmail.com, [2]rosvetnikk@gmail.com, [3]vladyslavspit@gmail.com

ABSTRACT

Neural network algorithms and probabilistic classifiers applied for text data set processing were analyzed. Results indicated advantages of architecture of recurrent and convolutional neural networks, deep learning neural network algorithms and Naive Bayes classifier considering accuracy of classification and processing speed of large data volumes. Classification algorithms' and appropriate mathematical models' development principles were generalized. Mathematical techniques are based on representing classification accuracy criteria, text processing speed in form of objective functions, and key parameters of neural network algorithms and probabilistic classifiers, as well as features of organization and volume of input data as objective function arguments. Mathematical modelling allowed identifying shortcomings of certain types of neural network and probabilistic classifiers limiting their scope. Algorithms based on the Naive Bayes classifier slowly analyzed large data sets, which limits their use. Working with neural network algorithms, features of the procedure for learning process optimization depending on type of neural network were outlined; approaches to optimization of deep and shallow neural network architecture were developed. Thereby analysis of efficiency of algorithms for machine classification of text blocks is proved to be relevant for solving fundamental problem of building artificial intelligence, and specific problems of real-time processing of large volumes of text data

Keywords: *Text Data, Recurrent Neural Network, Convolutional Neural Network, Deep Neural Network*

1. INTRODUCTION

Modern globalization processes, automation and digitalization of production, as well as the active development and spread of distributed information systems (DIS) in recent decades have led to an exponential growth of network information resources [1-3], with their data being subject to real-time analysis (Figure 1).

This, in turn, updated the research in the field of machine data analysis based on neural network and probability classifiers, which provide an opportunity to transfer the analysis of data [4-9] generated in DIS, to the level of M2M (machine to machine) interaction, as well as effectively classify the information generated at the level of P2P (person-to-person) interaction. At the same time, one of the key problems to be solved is the task of accurate and efficient classification of text data blocks. We should note the high level of the problem, which can be attributed to the field of creating artificial intelligence (AI), and show its significant practical value. Let us note that to date, with the expansion of the range of tools, a universal methodology for building effective algorithms for real-time machine analysis of text data has not been developed, which indicates the topicality of this study.

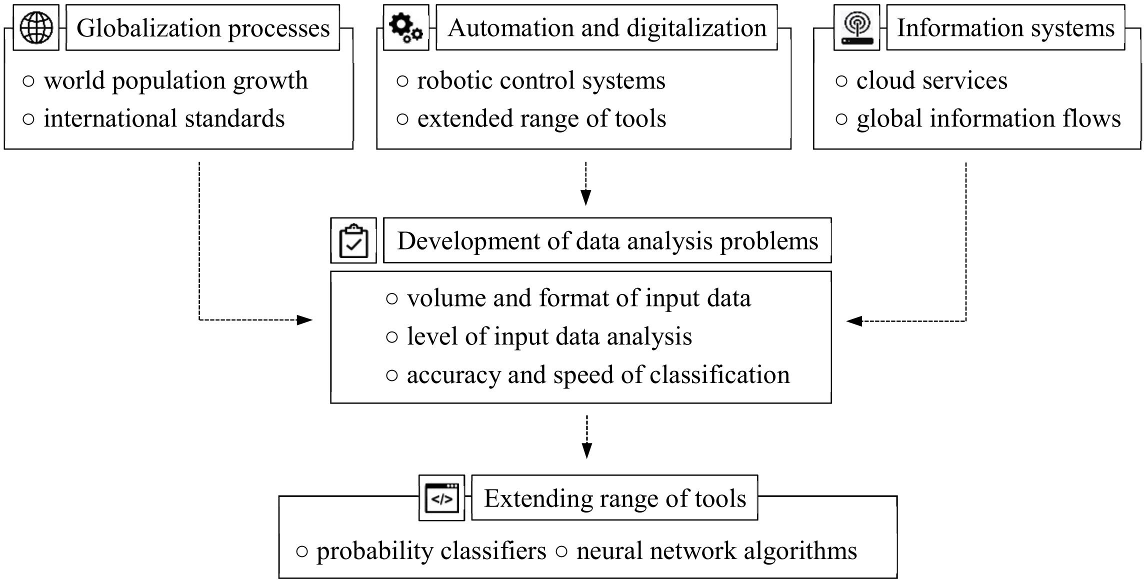
Globalization processes
○ world population growth
○ international standards
Automation and digitalization
○ robotic control systems
○ extended range of tools
Information systems
○ cloud services
○ global information flows
Development of data analysis problems
○ volume and format of input data
○ level of input data analysis
○ accuracy and speed of classification
Extending range of tools
○ probability classifiers ○ neural network algorithms

Figure 1: Trends in Actualization of the Problem of Real-Time Machine Analysis of Large Data Sets

Review of scientific publications on this topic included works on the modelling of neural network classification algorithms [4-6, 9] and probability classifiers [7, 8, 10]. In particular, considerable attention was paid to research on the use in the analysis of text blocks neural by network algorithms for deep learning [11-13], convolutional neural network algorithms [12], as well as algorithms based on Naive Bayes classifier [14]. Moreover, we reviewed works which deal with the comparison of the specified methods according to classification accuracy and speed of processing of various formats of the text data [15-18], which is in line with the topic of this study. Thereby the review indicated an **unsolved part of the overall problem** in the field of building a comprehensive methodology for developing machine analysis algorithms for processing of social networks text data sets. Consequently, the **objective of this study** was to develop a methodological framework for optimizing neural network and probable classifiers of social networks text blocks that work with large data sets in real time provided limited hardware resources.

2. RESEARCH METHODOLOGY

To build a comprehensive methodology for determining the effectiveness of the classifier of text blocks based on neural network algorithms, it is necessary to develop a generalized neural network model (artificial neural networks, ANN), appropriate mathematical techniques for determining objective functions and their arguments, and a model for presenting text data. According to this approach, in solving the optimization problem, the objective functions will be indicators of classification accuracy and speed of text data processing, and the arguments of the objective functions are ANN architecture parameters, algorithms of work and learning, as well as features of organization and volume of input data (Figure 2).

At the basic representation level, ANN is a set of neurons and a set of connections between them that are characterized by weight coefficients. The set of neurons is divided into subsets of neurons of the input, hidden and output layers. In this study, we propose to formalize it through the introduction of the corresponding matrices A, B and C. The matrix of neurons of the ANN input layer is one-dimensional, and includes I of a_i variables, where $i \in [1; I]$. The matrix of neurons of the ANN output layer is also one-dimensional, and includes K of c_k variables, where $k \in [1; K]$. The matrix of neurons of the ANN intermediate layer consists of a set of B_h matrices, where $h \in [1; H]$, each being one-dimensional and consisting of a limited number $J(h)$ of variables b_j^h, where $j(h) \in [1; J_1]$. In a general case, the ANN architecture may include bias neurons used to adjust the values coming to the input of the layer neurons. Bias neurons are located on the input and intermediate layers, so in the mathematical model it is proposed to represent as a one-dimensional matrix of variables $D: \{d_h\}$, where $h \in [0; H]$, consequently, d_0 is a variable corresponding to the input layer bias neuron.

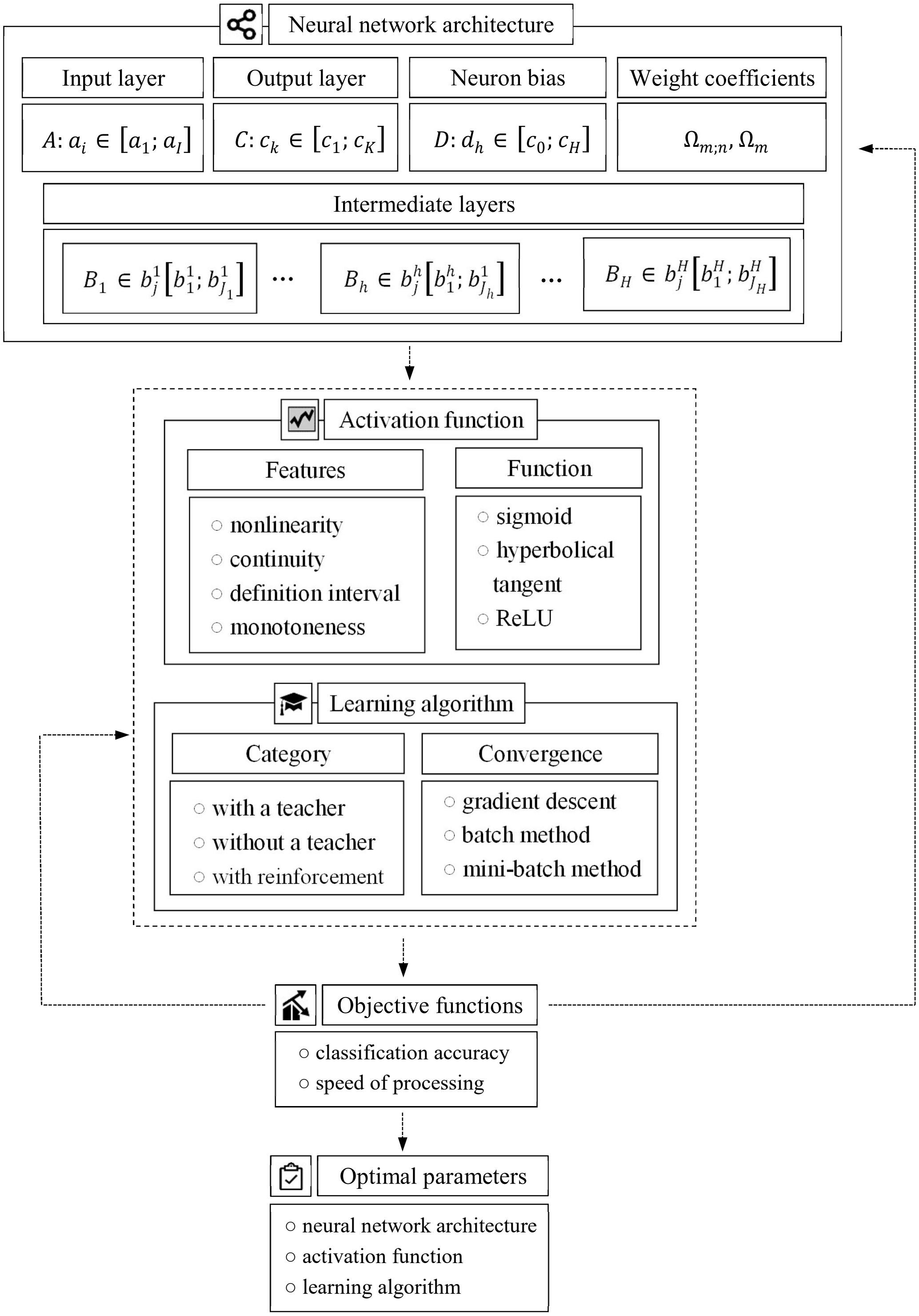

Figure 2: Basic Scheme of Calculation of Optimal Parameters of Neural Network Classifier on the Basis of Objective Functions

The values of the components of the matrices A, B and C are calculated on the basis of the normalized activation function from the sum of the initial values of the variables corresponding to the neurons of the previous layer multiplied by the weight coefficients. Weight coefficients should also be represented as a set of matrices. The basic model considers the connections of neurons of each layer with the following one, so this model will include the following sets of one-dimensional matrices of weight coefficients: (i) a set of matrices showing the connections between neurons of the input and the first hidden layer of the ANN:

$$\Omega_{0;1}(A, B_1): \left\{\Omega_{0;1}^{1}(a_1, B_1): \{\omega(a_1, b_1^1) \ldots \omega\left(a_1, b_{J_1}^1\right)\} \ldots \Omega_{0;1}^{I}(a_I, B_1): \{\omega(a_1, b_1^1) \ldots \omega\left(a_1, b_{J_1}^1\right)\}\right\} \qquad (1);$$

(ii) sets of one-dimensional matrices showing connections between neurons of hidden layers of the ANN:

$$\Omega_{h;h+1}(B_h, B_{h+1}): \left\{\Omega_{h;h+1}^{1}(b_1^h, B_{h+1}): \{\omega(b_1^h, b_1^{h+1}) \ldots \omega\left(b_1^h, b_{J_{h+1}}^{h+1}\right)\} \ldots \Omega_{0;1}^{J_h}\left(b_{J_1}^h, B_{H1}\right): \{\omega\left(b_{J_h}^h, b_1^{h+1}\right) \ldots \omega\left(b_{J_h}^h, b_{J_{h+1}}^{h+1}\right)\}\right\} \; (2);$$

(iii) a set of matrices showing the connections between the neurons of the last hidden and input layer of the ANN:

$$\Omega_{H;H+1}(B_h, C): \left\{\Omega_{H;H+1}^{1}(b_1^H, C): \{\omega(b_1^H, c_1) \ldots \omega(b_1^H, c_K)\} \ldots \Omega_{H;H+1}^{J_H}(b_1^H, C): \{\omega\left(b_{J_h}^H, c_1\right) \ldots \omega\left(b_{J_h}^H, c_K\right)\}\right\} \qquad (3).$$

However, the development of a more complex ANN architecture, such as recurrent networks, leads to increased capabilities through the establishment of connections between neurons of non-adjacent layers and neurons of one layer, which can be formalized through the introduction of sets of matrices $\mathbf{\Omega}_{m;n}$ and $\mathbf{\Omega}_m$, respectively, where $m \in [0; H+1]$ and $n \in [0; H+1]$, while $n \nsim m$.

Thus, matrices and sets of matrices A, B, C, D, $\mathbf{\Omega}_{m;n}$ and $\mathbf{\Omega}_m$ can be considered as functional parameters that determine the ANN architecture and the arguments of the objective functions of the efficiency of the neural network classifier. Other arguments of the objective functions can be determined through the introduction of the activation function F_A, which normalizes the signal transmitted between the neurons of the ANN. A typical set of features of the activation function includes nonlinearity, which is necessary for ANN multilayer structures, continuity of differentiation, which is relevant when using optimization methods based on gradient descent, definition interval, and monotoneness. This study used a set of the most typical activation functions applied when working with this class of problems such as the sigmoid $F_A^S(x) = 1/(1 + e^{-ax})$, the hyperbolic tangent $F_A^T(x) = (e^{2ax} - 1)/(e^{2ax} + 1)$, ReLU $F_A^R(x) = log(1 + e^x)$, and so on.

Finally, to form a set of arguments for the objective functions of the ANN performance appraisal system, it is necessary to determine the type of learning, which guide further development of learning dataset. Standard sets are divided into categories such as learning with a teacher, learning without a teacher, and reinforcement learning. At this stage, it is also necessary to form sets of actual algorithms of convergence of the ANN system, which can be similarly divided into such groups as stochastic gradient descent method, batch gradient descent method, and mini-batch gradient descent method, characterized by different levels of vulnerability to local extremum convergence and maximum load on computing resources.

Similarly, the scheme for calculating the optimal parameters can be determined for the probability classifier, as an algorithm that, based on a given set of input data, determines the probability distribution according to the set of classes. But we propose to consider the Bayes classifier in this study. The probabilistic classification approach based on the Bayes classifier is considered optimal due to the fact that the result of its work is an unambiguous answer in cases where an unambiguous answer is possible, and a quantitative measure of ambiguity in the opposite case, and therefore cannot be improved. The disadvantage of this classifier is the exponential growth of the training sample according to the growth of the number of variables, so later, in the experimental study, this algorithm was used only to compare the efficiency of neural network and probabilistic algorithms only for bounded sequences. It should be noted that this problem can be partially eliminated (reduce exponential to linear) through the use of Naive Bayes classifier, based on the assumption of independence of variables — this approach was used in the work. This allows increasing the speed of the algorithm while reducing its accuracy for a number of problems.

3. RESULTS

The analysis carried out in the previous section allowed building effective schemes of neural network classifiers, analysing the accuracy of their work and time spent on processing text blocks of fixed format and size. The text materials of the online publication British Broadcasting Corporation posted on an open network resource were used as input data [19]. All models of classifiers were learnt on 1,725 samples of the specified database, and the next 500 samples were used to assess the accuracy of determining one of the five classes, which corresponds to the text block. The computing power of the workstation can be described by the following indicators: (i) Intel Core i7-8700 3.20 GHz CPU, 16 GB/3200 MHz RAM, Sata-M2 SSD with a total capacity of 1 TB.

The model of the deep neural network (DNN), which was built during the experimental study, consists of the following components:

- a set of input data, which is determined by the number of classes, training samples and test samples;
- tokenizer for pre-segmentation of input data according to the average sentence length and the total volume of the dictionary;
- model for distributed representation of GloVe words;
- sequence alignment unit (embedding);
- a neural network model characterized by the architecture, the number of neurons in each of the hidden layers, the activation function used on each of the hidden layers, the optimizer, and the loss function;
- a unit of analysis of accuracy and speed of data classification.

Table 1 and 2 provide the results of experimental studies. The study indicated the priority of using the sigmoid activation function for most types of DNN architecture; a combination of sigmoid and ReLU in this case. Among the optimization algorithms, optimal results were obtained for the Adam algorithm; this is typical for DNN, while for shallow neural networks (SNT) the stochastic gradient descent (SGD) method show the optimal results.

Table 1: Optimization of the Classifier of Text Blocks on the Basis of a Deep Neural Network with One and Two Fully Connected Layers

Item No.	Activation function	Number of epochs	Optimization algorithm	Fully connected layers	Number of neurons in the layer	Time of passing of one epoch	Classification accuracy
D-1.1	sigmoid	20	Adam	1	5	3 s	94.2%
D-1.2	sigmoid	20	Adam	2	32	4 s	95.8%
	sigmoid				5		
D-1.3	ReLU	20	Adam	2	32	4 s	93.8%
	sigmoid				5		

Table 2. Optimization of the Classifier of Text Blocks on the Basis of a Deep Neural Network with Three Fully Connected Layers

Item No.	Activation function	Number of epochs	Optimization algorithm	Fully connected layers	Number of neurons in the layer	Time of passing of one epoch	Classification accuracy
D-2.1	ReLU	20	Adam	3	32	4 s	93.4%
	ReLU				16		
	sigmoid				5		
D-2.2	sigmoid	20	Adam	3	32	4 s	94.8%
	sigmoid				16		
	sigmoid				5		
D-2.3	linear	20	Adam	3	32	4 s	25.2%
	linear				16		

	linear				5		
D-2.4	sigmoid	20	Adam	3	16	3 s	96%
	sigmoid				16		
	sigmoid				5		
D-2.5	sigmoid	20	Adam	3	16	3 s	95.2%
	sigmoid				10		
	sigmoid				5		

Such mechanisms as gated recurrent unit (GRU) and long short-term memory (LSTM) were used to build the classifier based on recurrent neural networks (RNN). A sufficient accuracy level was obtained only for LSTM algorithms (Tables 3 and 4), and the experimental study showed the optimal kernel size as a value which increase leads to increased data processing time, but does not affect the classification accuracy.

Table 3. Optimization of Text Block Classifier Based on Recurrent Neural Network with GRU

Item No.	Activation function	Number of epochs	Optimization algorithm	GRU input data	Simple RNN	Time of passing of one epoch	Classification accuracy
R-1.1	ReLU	20	Adam	32	16	12 c.	18.6%
R-1.2	sigmoid	20	Adam	32	16	12 c.	16.2%
R-1.3	linear	20	Adam	32	16	12 c.	14.2%
R-1.4	ReLU	20	Adam	16	16	13 c.	23.2%
R-1.5	ReLU	10	Adam	32	16	8 c.	19.2%
R-1.6	ReLU	10	Adam	10	16	10 c.	16.4%
R-1.7	sigmoid	20	Adam	16	16	12 c.	14.8%

Table 4. Optimization of Text Block Classifier Based on Recurrent Neural Network with GRU

Item No.	Activation function	Number of epochs	Optimization algorithm	LSTM size	Time of passing of one epoch	Classification accuracy
R-2.1	Softmax	20	Adam	300	120..160 s	95%
R-2.2	Softmax	20	Adam	64	40-50 s	93.2%
R-2.3	ReLU	20	Adam	64	10 s	22%
R-2.4	sigmoid	20	Adam	64	21 s	96%

The efficiency of classification algorithms based on convolutional neural networks (CNN) was also experimentally evaluated. At the level of the analytical solution of the problem of optimization of the process of classification of text blocks this type of classifiers was considered effective. According to the general scheme, the result of each of the convolutions was to be transferred to the next to obtain a template, which, depending on the size of the convolution kernel could be groups of related words (phrases and individual expressions). This mechanism can be used to determine the emotional colour of the text block, identify elements of spam and categorize the topic. However, a practical study showed a relatively low accuracy of CNN classifiers (Table 5).

Table 5. Optimization of Text Block Classifier Based on Convolutional Neural Network

Item No.	Activation function	Number of epochs	Optimization algorithm	Fully connected layers	Convolution layers	Convolution kernel	Training samples	Integration time	Classification accuracy
C-1	ReLU	20	Adam	1	1	5	5	2 s	19.2%
C-2	sigmoid	20	Adam	1	1	5	5	2 s	19%
C-3	linear	20	Adam	1	1	5	5	2 s	18.2%
C-4	ReLU	20	Adam	1	1	300	5	26 s	19%
C-5	ReLU	20	Adam	2	1	32	5	12 s	24.6%
C-6	ReLU	20	Adam	2	2	32	5	13 s	23%

The specified training sample (BBC, n.d.) was also used for the Naive Bayes classifier (NBC) model. High classification accuracy was obtained (98%), but the speed of text processing dropped significantly when working with datasets of more than 500 characters, which limits the functionality of this algorithm.

The next series of studies was conducted for the Reuters training kit, which consisted of 7,000 text blocks with an average size of 10,000 characters. The studies were conducted for the previously selected DNN, RNN, and NBC architectures. At the same time, the vector size was reduced from 3,000 to 50 for embedding, which was compensated by an 80-fold increase in training time. As a result, the accuracy of the DNN-classifier decreased to 78%, due to the larger size of sentences and text fragments, which leads to gradient degradation. Also, the NBC accuracy decreased to 64%, which was partially levelled (increased to 81%) by using the Complement Naïve Bayes algorithm. For the RNN classifier, in addition to reducing the accuracy to 60%, the training time significantly increased (up to 40 minutes per epoch).

4. RESULTS ANALYSIS

The study provided an opportunity to form an optimization model for a text data classifiers based on DNN and RNN algorithms. The generalization of this model is presented in Figure 3. Methodological recommendations for the use in real-time text data classification in the context of hardware resource constraints for these classifiers, as well as CNN and NBC models were also made:

1. It is noted that the DNN classifier is optimal for the classification of text blocks, where the set of input data is assigned a class or label. This type of neural network algorithms can also be used to predict regression in the form of a real value defined by the appropriate set. The DNN classifier can be recommended for use in classification problems for medium-sized text blocks (about 375 characters) and longer than average (about 1,000 characters).
2. The key stage of work with the RNN classifier, which determines the relevance of its further application is the training stage. For most problems, this classifier is proposed to be used in conjunction with LSTM, which solves the problem of learning the periodic network. The RNN-LSTM model can be used most effectively when working with word sequences and elementary text blocks in a natural-language processing (NLP) program.
3. CNN classifier has advantages when working with data that have a spatial representation, such as through a one-dimensional input matrix. However, research has shown that presenting input data to CNN classifier as a one-dimensional matrix allows using this algorithm to classify text blocks and track relevant connections.
4. NBC is an effective regressive method of classification of small text data, the classification accuracy of which decreases with increasing size of the dataset, starting from the threshold value to be determined in case of application of this algorithm and compared with the input data format considered in the task.

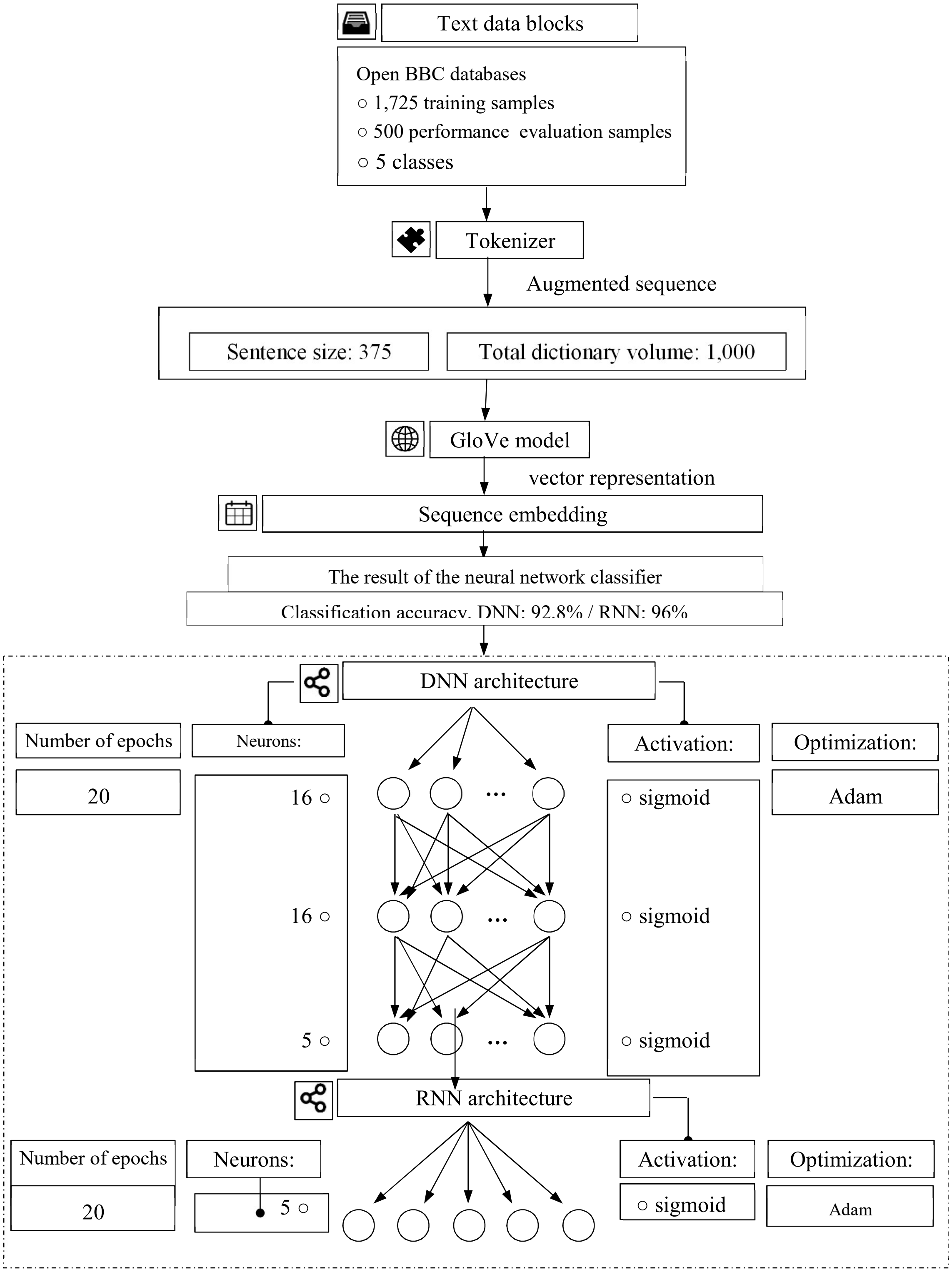

Figure 3: Text Data Classifier Optimization Model Based on DNN and RNN Algorithms

5. CONCLUSIONS

The analysis of modern neural network algorithms and probabilistic classifiers that can be effectively used when working with text data sets indicated the advantages of the architecture of recurrent and deep neural networks, as well as Naive Bayes classifier in solving the problem of high classification accuracy and the speed of real-time processing of large data volumes provided limited computing resource. In accordance with the task, the principles of building neural network classification algorithms were generalized, and mathematical techniques were developed based on the presentation of indicators of classification accuracy and speed of text data processing as objective functions. Similarly, the parameters of the neural network architecture and probability classifiers, as well as the features of the input data arrangement were presented as arguments of the objective functions. According to the developed mathematical techniques, the solution of the problem of optimization of classification of constituent elements of a text data set was shown as part of the analytical solution, which allowed generalizing algorithms for working with typical problems. Moreover, a method of accurate numerical calculation of optimal parameters of classifiers was built based on software algorithms. Mathematical modelling also allowed identifying the shortcomings of neural network and probability classifiers that limit their scope. Algorithms based on Naive Bayes classifier showed low speed in the analysis of large data sets. When working with neural network algorithms, the peculiarities of the procedure of optimization of the learning process were indicated depending on the type of neural network and developed general approaches to optimize the deep and neural network architecture.

Thereby it was developed comprehensive methodology of the machine analysis of text data sets based on neural networks' and probable classifiers optimizing. Proceeding of the research must include calculation of optimal parameters of specialized neural network classifier on the basis of objective functions.

REFERENCES:

[1] A. Fandango and W. Rivera, "High performance storage for big data analytics and visualization", in R.S. Segall and J.S. Cook (Eds), *Handbook of research on big data storage and visualization techniques*, Hershey, PA: IGI Global, 2018, pp. 254-275.

[2] N. Mukherjee, S. Neogy and S. Chattopadhyay, "Big data storage", in N. Mukherjee, S. Neogy and S. Chattopadhyay, *Big data in eHealthcare. Challenges and perspectives*, New York, NY: Chapman and Hall, 2019, pp. 163-190.

[3] Z. Zou and Q. Kong, "Secure provable data possession for big data storage", in X. Liu, R. Anand, G. Xiong, X. Shang and X. Liu, *Big data and smart service systems,* Cambridge, MA: Academic Press, 2017, pp. 27-41.

[4] S.P. Kim, and D.H. Reddy, "Text mining: classification of text documents using granular hybrid classification technique", *International Journal of Research in Advent Technology,* Vol. 7, No. 6, 2019, pp. 1-8, available from: https://doi.org/10.32622/ijrat.76201910

[5] R. Ma, S. Teragawa and Z. Fu, "Text sentiment classification based on improved BILSTM-CNN", in J. He et al., *2020 Asia-Pacific conference on image processing, electronics and computers (IPEC),* Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2020, pp. 1-4.

[6] V. Makarenkov, B. Shapira and L. Rokach, *Language models with pre-trained (GloVe) word embeddings*, Retrieved from: https://arxiv.org/abs/1610.03759 (October 10, 2020).

[7] F. Nafa, S. Othman and J. Khan, "Automatic concepts classification based on bloom's taxonomy using text analysis and the Naive Bayes classifier method", in J. Uhomoibhi, G. Costagliola, S. Zvacek and B. M. McLaren (Eds), *Proceedings of the 8th international conference on computer supported education,* Setúbal, Portugal: SciTerPress, Vol. 2, 2016, pp. 391-396.

[8] I.U. Ogul, C. Ozcan and O. Hakdagli, "Fast text classification with Naive Bayes method on Apache Spark", in G. Ince et al., *25th Signal processing and communications applications conference (SIU),* Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2017, pp. 1-4.

[9] X. Wang, J. Li and Y. Liu, "Application of convolutional neural network (CNN) in microblog text classification*",* in J. Li et al., *2018 15th international computer conference on wavelet active media technology and information processing (ICCWAMTIP),* Piscataway, NJ: Institute of Electrical and Electronics Engineers, 2018, pp. 127-130.

[10] J. Ababneh, "Application of Naïve Bayes, decision tree, and k-nearest neighbors for automated text classification", *Modern Applied Science,* Vol. 13, No. 11, 2019, pp. 31, Available from: https://doi.org/10.5539/mas.v13n11p31

[11] I. Goodfellow, Y. Bengio and A. Courville, *Deep learning,* Cambridge, MA: The MIT Press, 2017.

[12] S. Kostadinov, *Understanding GRU networks*, Retrieved from: https://towardsdatascience.com/understanding-gru-networks-2ef37df6c9be (October 10, 2020).

[13] C. Nwankpa, W. Ijomah, A. Gachagan and S. Marshall, "Activation functions: comparison of trends in practice and research for deep learning", *ArXiv,* No. 1811.03378, 2018, pp. 1-20.

[14] M. Mandt, M. Hoffman and M. Blei, "Stochastic gradient descent as approximate Bayesian inference", *Journal of Machine Learning Research,* Vol 18, 2017, pp. 1-35.

[15] A.Z. Amin, *Convolutional neural network: text classification model for open domain question answering system,* Retrieved from https://arxiv.org/abs/1809.02479 (October 10, 2020).

[16] J. Brownlee, *When to use MLP, CNN, and RNN neural networks,* Retrieved from: https://machinelearningmastery.com/when-to-use-mlp-cnn-and-rnn-neural-networks (October 10, 2020).

[17] M.S. Islam, S.M. Khaled, K. Farhan, M.A. Rahman and J. Rahman, "Modeling spammer behavior: Naïve Bayes vs. artificial neural networks", in H. Lee et al., *2009 international conference on information and multimedia technology,* Washington, DC: IEEE Computer Society, 2009, pp. 52-55.

[18] V.S. Yaremenko and M.V. Tarasenko, "Comparative analysis of software libraries for the classification of text data using artificial neural networks", *Scientific Notes of TNU Named after V. I. Vernadsky, Series: Technical Sciences,* Vol. 30/69, No. 3, 2019, pp. 214-218.

[19] British Broadcasting Corporation, *BBC databases,* Retrieved from http://mlg.ucd.ie/datasets/bbc.html (October 10, 2020).