ISSN 2522-9818 (print)
Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 3 (25) ISSN 2524-2296 (online)

UDC 004.91 DOI: https://doi.org/10.30837/ITSSI.2023.25.005

O. BARKOVSKA, K. VOROPAIEVA, O. RUSKIKH

# JUSTIFYING THE SELECTION OF A NEURAL NETWORK LINGUISTIC CLASSIFIER

The **subject matter** of this article revolves around the exploration of neural network architectures to enhance the accuracy of text classification, particularly within the realm of natural language processing. The significance of text classification has grown notably in recent years due to its pivotal role in various applications like sentiment analysis, content filtering, and information categorization. Given the escalating demand for precision and efficiency in text classification methods, the evaluation and comparison of diverse neural network models become imperative to determine optimal strategies. The **goal** of this study is to address the challenges and opportunities inherent in text classification while shedding light on the comparative performance of two well-established neural network architectures: Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN). To achieve the goal, the following **tasks** were solved: a comprehensive analysis of these neural network models was performed, considering several key aspects. These aspects include classification accuracy, training and prediction time, model size, data distribution, and overall ease of use. By systematically assessing these attributes, this study aims to provide valuable information about the strengths and weaknesses of each model and enable researchers and practitioners to make informed decisions when selecting a neural network classifier for text classification tasks. The following **methods** used are a comprehensive analysis of neural network models, assessment of classification accuracy, training and prediction time, model size, and data distribution. The following **results** were obtained: The LSTM model demonstrated superior classification accuracy across all three training sample sizes when compared to CNN. This highlights LSTM's ability to effectively adapt to diverse data types and consistently maintain high accuracy, even with substantial data volumes. Furthermore, the study revealed that computing power significantly influences model performance, emphasizing the need to consider available resources when selecting a model. **Conclusions**. Based on the study's findings, the Long Short-Term Memory (LSTM) model emerged as the preferred choice for text data classification. Its adeptness in handling sequential data, recognizing long-term dependencies, and consistently delivering high accuracy positions it as a robust solution for text analysis across various domains. The decision is supported by the model's swift training and prediction speed and its compact size, making it a suitable candidate for practical implementation.

**Keywords**: text classification; neural networks; LSTM; CNN; classification accuracy; model comparison; sequential data.

## Introduction

Text classification is an important task in today's information society, as it allows you to automatically process and classify large amounts of textual information. This is especially true in the digital revolution, when the amount of text data is constantly growing, and with it the need for efficient methods of analysing and disseminating information. Text classification is widely used in many areas of life, such as medicine, finance, marketing, social media, Internet search, and many others, as shown in Figure 1. For example, text classification in medicine can help to automatically determine diagnoses based on patient symptoms or filter out malicious content on social media [1].

Text classification plays an important role in the field of natural language processing (NLP) and is central to other NLP tasks. Text classification helps to determine whether texts belong to certain categories or topics or solve other problems related to the distribution of textual information [2].

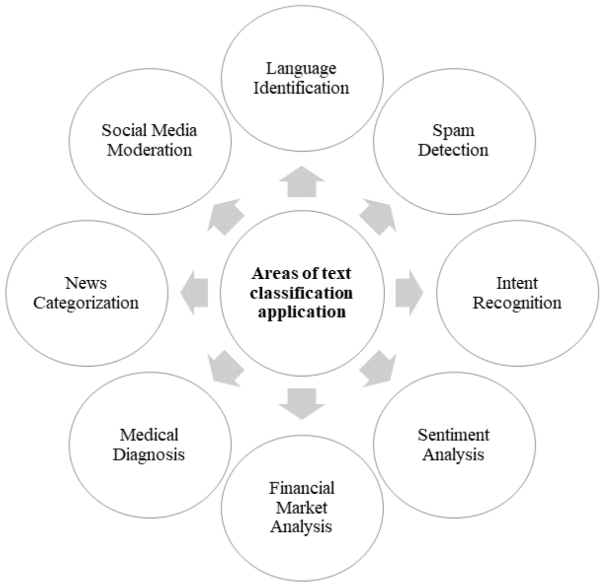


**Fig. 1.** Areas of text classification task application

For text classification, various tools are used to achieve a high level of accuracy in this task,

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2023. No. 3 (25)

often combining existing methods that have proven themselves [3]. First of all, rules and heuristic methods based on predefined rules and expert knowledge are used. These methods are especially effective when the data structure is simple and the relationships between categories are already known. Machine learning methods are also used for classification. With machine learning, you can create models that automatically recognize patterns in text data and perform classification based on a set of training data. In this field, methods such as Naive Bayes, Support Vector Machine (SVM) [4], Decision Trees, and others are widely used.

Neural networks are a special kind of machine learning methods inspired by the structure of the brain's neural network. Neural network models show impressive results and advantages over traditional methods shown in Figure 2, such as Naive Bayes, Support Vector Machine (SVM), Decision Trees, and others, as they allow to automatically detect internal patterns in text data and classify with high accuracy, as shown in [5].

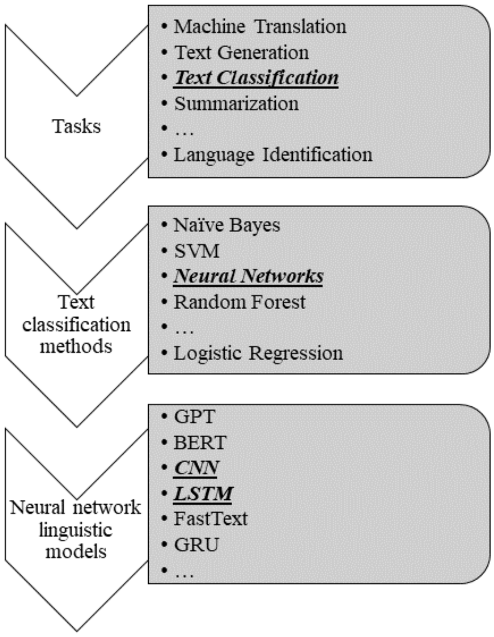


**Fig. 2.** Generalized analysis of the problem area

When solving text classification tasks, it is important to choose a neural network model that best suits the specifics of the task and the needs of the research. Choosing the best model requires a careful comparison of different architectures and their characteristics to ensure optimal performance and accuracy. To select the best model, a comparative analysis of various model characteristics is necessary. For this analysis, we have selected the following key characteristics: classification accuracy, learning rate, prediction rate, model size, data dissemination, and overall ease of use. In the following, we report on the comparison of these models based on these characteristics to find out their effectiveness and suitability for use in different text classification scenarios.

## Analysis of last achievements and publications

Work [6] shows that an important role in improving classification efficiency is played by the use of Word Embedding, which allows words to be converted into vectors of numbers with small sizes while preserving the semantic connections between them. This enables neural networks to work with both text and numeric data, allowing for higher classification accuracy and reduced computing costs. In addition, in recent years, new and more powerful methods have emerged, such as Contextual Embedding, which allow for more accurate capture of the semantic context of words and sentences. As an extension of Word Embedding, Contextual Embedding takes into account the context of each word in the text and thus allows for a deeper understanding of textual information using neural networks. These trends in the use of Word Embedding and Contextual Embedding in neural network models have become an important means of improving text classification results (Table 1).

In addition, some studies emphasize the importance of achieving state-of-the-art results in text classification tasks. A high level of accuracy in text classification can be achieved by using pre-training and fine-tuning models. Such approaches are becoming more and more relevant, which increases the variety of applications and the development of text classification for various tasks.

In our study, we will focus on analyzing and comparing different approaches using neural networks for text classification, in particular, we will compare the performance of models using traditional Word Embedding [7] and modern Contextual Embedding [8–9]. Such an analysis will allow us to better understand current trends in this field and identify the most promising areas of research in text classification using neural networks.

Comparison of neural network models for text classification is a critical task that allows us to determine the most efficient and accurate approaches to specific tasks. It is especially interesting to compare models based on Contextual Embedding and static Word Embedding.

ISSN 2522-9818 (print)
Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 3 (25) ISSN 2524-2296 (online)

Such a comparison will show the pros and cons of different approaches to text vectorization and emphasize the importance of taking context into account when analyzing text data. A comparative study of neural network models for text classification has several advantages. It allows us to determine which models are more effective for different types of text data, which can improve the quality of classification and the accuracy of the results. Knowing the advantages and limitations of different models also helps to choose the best classifier for a particular text classification task. However, performing a neural network comparison can cause certain problems. For example, it may require significant computing resources and time, as such models usually have a large number of parameters. It is also important to choose the right metrics to evaluate the results and avoid training the models repeatedly on the same dataset. Taking all these factors into account, studying the comparison of neural network models for text classification remains an interesting and important task that contributes to the development and improvement of this field.

**Table 1.** *Ways to improve text classification accuracy*

| Strategy | Description |
|---|---|
| Contextual vectors | Using contextual word vectors such as BERT, GPT, or ELMO to capture context |
| Model selection | Selecting a suitable classification model, such as Naive Bayes, SVM, LSTM, CNN, etc. |
| Tuning of hyperparameters | Adjusting hyperparameters to optimize the performance of the selected model |
| Data augmentation | Generating synthetic data to expand the training set and improve model generalization |
| Ensemble approach | Combining forecasts from several models to improve overall accuracy |
| Transfer of learning | Use of pre-trained models and their fine-tuning for a specific classification task |
| Cross-validation | Evaluating model performance using methods that take into account multiple trials for accuracy |
| Text preprocessing | Text cleaning and normalization, redundant word removal, stemming, and special character processing |

One of the first known successful applications of neural networks for text classification was based on the Convolutional Neural Network (CNN) architecture. The CNN model [10-11] has been successfully applied to image analysis, but has also proven to be effective in processing text data. Using CNNs and Word Embedding, which transform words into numerical vectors [12–14], impressive results have been achieved in text classification tasks, in particular in determining the tone of the text. However, recent trends indicate that there are more powerful and flexible approaches to text classification, including Contextual Embedding. Models such as Transformer, BERT, or GPT are very popular in the world of applied machine learning because they can analyze texts in context, taking into account the semantic relationships between words and sentences.

As a first step, we empirically compare the performance of contextual embeddings with classical embeddings such as word2vec [15] and GloVe [16]. Contextual Embedding is a type of Word Embedding in which the vector values take into account the context in which the words appear in the text. This gives you a more accurate representation of the word depending on the context. Context-aware neural network models, such as BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer), can achieve high results in various classification tasks. On the other hand, static Word Embeddings [2], such as Word2Vec and GloVe, assign a fixed vector to each word that is independent of context. These models work well for many tasks, but do not take into account semantic dependencies between words in a sentence, which can lead to less accurate text classification results.

For this reason, we analyze the impact of using Contextual Embedding on text classification accuracy. Using this analysis, we can evaluate how effective Contextual Embeddings are compared to traditional static Word Embeddings in various classification scenarios. They check whether contextual vector representations provide better results and more accurate classification, especially in complex and diverse text data analysis scenarios. This approach allows us to understand which word vectorization methods are more effective for different types of textual information and helps to draw conclusions about the advantages and limitations of each approach in text classification tasks.

**The purpose of this study** is to conduct a comparative analysis of two important neural network architectures – Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN) – to provide recommendations for the selection of a neural network

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2023. No. 3 (25)

linguistic classifier. The study includes an analysis of various aspects of both models, including classification accuracy, training and prediction time, and model size, taking into account three different training sample sizes. For applications in areas such as news and content classification, reviews and testimonials classification, social media analysis, and others.

To achieve this goal, the following key tasks must be solved, which are discussed in detail in this study:

– analysis of the problem area and justification of the relevance of the topic;

– comparative analysis of existing text corpora;

– formulation of criteria and requirements for neural network classifiers;

– performing experimental studies on samples of different sizes to compare the training and prediction time, prediction accuracy, and model size when using LSTM and CNN neural network models as a classifier;

– analysis of the obtained results;

– justification for choosing the most effective model.

## Materials and methods

There are many different corpora for training text classification models that have become an important resource for research and applications in the field of natural language processing [17, 18]. These corpora represent different types of text data from different sources and cover a wide range of topics. To ensure a successful comparison of neural network models for text classification, it is important to carefully select an appropriate corpus that meets the research goals and task characteristics. The most popular cases include the ones listed in Table 2: IMDB corpus, Reuters news corpus, PubMed scientific article corpus, Twitter sentiment analysis corpus.

**Table 2.** *Text bodies for text classification in NLP tasks*

| № | Corpus | Description | Classification task |
|---|---|---|---|
| 1 | IMDB corpus | Contains movie reviews from IMDB, divided into positive and negative categories | Binary text classification (positive/negative reviews) |
| 2 | Reuters news corpus | Contains news articles organized by topic | Categorical classification of texts |
| 3 | PubMed scientific articles corpus | Contains scientific articles from medical sources categorized by topic or importance | Categorical text classification |
| 4 | Twitter sentiment analysis corpus | Contains Twitter messages categorized by sentiment (positive, negative, or neutral) | Sentiment classification in short texts |

Given the complexity of the text classification task and the desire to perform a qualitative comparison of neural network models, it was important to choose a suitable corpus for training the models. The most convenient and suitable corpus for our research purposes was the IMDB corpus.

The reasons for choosing the IMDB corpus are its diversity, accessibility, and representativeness. Due to the large amount of data, IMDB can provide a sufficient number of examples for training and testing models, which is important for reliable comparison of their performance. In addition, IMDB contains textual reviews with emotional coloring, which is an important feature for solving the binary classification task. Another important advantage of IMDB is its accessibility, which allows researchers from all over the world to use this corpus for their studies. This contributes to the wide applicability of the results and the possibility of comparison with other scientific studies. In addition, the variety of text lengths in the IMDB corpus allows testing the ability of models to work with sequences of varying complexity and length, which is crucial for realistic analysis of neural network performance on a variety of input data.

The results of the comparison of neural network models for text classification find application in many promising areas and tasks, such as:

1. Sentiment analysis of product and service reviews – this will help companies understand customer satisfaction, identify problem areas, and improve their products.

2. Content classification in web services – this will help organizations and platforms automatically filter content to ensure safety and a positive user experience.

3. Emotion analysis in social media – this will help to understand the reaction to news, events and publications, which is important for advertisers and marketers.

4. Monitoring brands and companies to track and analyze public opinion, helping managers respond to changes in the perception of goods and services.

Analyzing the accuracy and effectiveness of different approaches is an important step to better

Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 3 (25)
ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

understand the potential advantages and limitations of contextual embedding compared to traditional methods. This data can serve as an important guide in choosing the optimal model for specific text classification tasks. The goal of our research is to improve the quality and accuracy of classification results and to contribute to the development and implementation of new methods in this field.

Choosing the right neural network architecture is an important step in solving the text classification problem. The appropriate architecture can affect the efficiency and accuracy of the model in the text classification task from different perspectives. The architecture of a neural network determines how it solves the problem of analyzing text data. Different architectures may have different approaches to pattern recognition, word dependency detection, and interpretation of textual information. The right architecture can help solve specific problems in the text classification task and provide more accurate and reliable results. In this study, we chose two popular models for comparative analysis: Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM), because these networks represent two different approaches to analyzing text data and have their own features that can be useful for different types of texts and classification tasks (Table 3).

**Table 3.** *Comparative analysis of LSTM and CNN models for text classification*

| Characteristics | LSTM Model | CNN Model |
|---|---|---|
| Basic architecture | Recurrent neural network with LSTM layer | Convolutional neural network with Conv1D layers |
| Typical tasks | Analysis of sequential data, text data | Analysis of sequential data, images |
| Features | Takes into account the context of word dependencies | Detects local patterns in data |
| Learning algorithms | Backward error propagation, Adam optimizer | Back propagation of error, Adam optimizer |
| Activation functions | Tanh, Sigmoid | ReLU |
| Memory usage | Uses short-term and long-term memory | Does not use memory |
| Application | Sequential text analysis, language translation | Image, video analysis |
| Implementation libraries | TensorFlow, Keras | TensorFlow, Keras |

The CNN model is unique and specialized for recognizing patterns in images, but it can also be successfully used to process text data, where it recognizes local dependencies and important features of the textual context. On the other hand, the LSTM model is a part of recurrent neural networks and has the ability to store and use information from previous steps, which allows it to work efficiently with sequential data, especially text sequences, and take into account the context in texts.

In order to make the right choice of a neural network linguistic model for classifying input texts in future studies, this section presents the results of a comparative analysis of the selected models according to the following criteria: classification accuracy, training speed, prediction speed, model size, data distribution, and overall usability.

Accuracy is an important indicator for determining the performance of classification models. This characteristic measures the proportion of correct predictions made by the model out of the total number of predictions. In our experiment, accuracy allows us to understand how accurately the selected models identify positive and negative reviews.

Learning and prediction times are important aspects when implementing neural models. The learning rate indicates the time it takes for the model to adapt to the data during training. It can affect the overall training time of the model. Prediction speed indicates the time it takes for the model to predict new input data. In our study, we measure these parameters for each architecture and sample size to understand which model can be more efficient in terms of computational complexity.

Model size reflects the number of parameters used to hold information in the model. In our experiment, we can use this metric to understand which architecture has more parameters to store information. A large model size can affect memory and computation requirements.

By comparing two models based on data distribution characteristics, we evaluate how well each model adapts to different types of data. Possible changes in the distribution can affect the training results, as certain models may be more sensitive to changes in the data distribution. In our experiment, this metric is used to understand how well each model is able to generalize the knowledge gained during training to new and unknown data. It is important to keep in mind that real-world data that a model encounters in solving practical problems may contain variations and diversity. Thus, the importance of adapting to different data

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2023. No. 3 (25)

distributions emphasizes the need to choose a model that demonstrates stable and reliable performance even under variable input conditions.

The importance of evaluation parameters and metrics lies in their ability to provide an objective and complete assessment of model performance. The accuracy rating helps to understand which of the architectures is better at solving the classification task. Measuring training and prediction time gives us an idea of how fast the models work in real time. Model size is important in practical applications where computing resources may be limited.

In addition, the experiment investigated the effect of the training set size on the time to obtain the result, namely

- Small Training Set. In this case, lower-level computing resources were used, which are characterized by limited capacity and processing of a limited amount of data;
- Medium Training Set. For this category, medium computing resources were used, which allowed working with a larger amount of data and provided higher computing power;
- Large Training Set. To ensure the efficiency of calculations, powerful computing resources were used in this case, which allowed us to process large amounts of data quickly and efficiently.

By analyzing the experimental results and considering the impact of the computing base on the performance of models for different training set sizes, we can understand the importance of the role of computing resources in determining the efficiency of neural networks under different operating conditions. The results allow us to better understand how computing resources affect key model characteristics, such as training time, prediction time, and classification accuracy. This analysis helps to select the most efficient models for specific work scenarios. For example, for tasks with limited computational resources, it may be important to favor models that perform best under constrained conditions. On the other hand, if there is a lot of computing power available, it may make sense to use more sophisticated models with higher accuracy and the ability to adapt to different types of data.

In the Results section, we present detailed results obtained during the experiments with LSTM and CNN text classification models. The experiments were conducted with three different sizes of training samples: small (1000 samples), medium (10000 samples), and large (50.000 samples). Table 4 shows the results of the comparative analysis of two models – Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN) – for different training set sizes.

## Research results

All models presented in Table 4 were trained on the same hardware computing platform with the same characteristics. This ensured an adequate comparison of model performance, as the possibility of hardware differences affecting the results was excluded. In addition, the same input data was used for all experiments. The dataset was divided into training and test sets, ensuring an even distribution of classes in each set. This helped to avoid bias in the results due to class imbalance.

**Table 4.** *Results of comparative analysis of LSTM and CNN models on different sizes of training samples*

| Model/Sample size | Training time (sec) | Prediction time per sample (sec) | Training accuracy | Accuracy on test data | Model size (parameters) |
|---|---|---|---|---|---|
| LSTM Small Training Set | 144.22 | 0.00173 | 0.8860 | 0.7288 | 689.473 |
| CNN Small Training Set | 143.05 | 0.00140 | 0.7500 | 0.5002 | 804.225 |
| LSTM Medium Training Set | 272.19 | 0.00171 | 0.9346 | 0.8377 | 689.473 |
| CNN Medium Training Set | 188.91 | 0.00139 | 0.9877 | 0.8422 | 804.225 |
| LSTM Big Training Set | 483.21 | 0.00173 | 0.9489 | 0.8705 | 689.473 |
| CNN Big Training Set | 443.11 | 0.00099 | 0.9848 | 0.8652 | 804.225 |

The data preprocessing methods shown in Figure 3 were also identical for both models. This allowed us to create a common initial context for all models and compare their performance under the same conditions. This approach to conducting experiments helped to avoid biases when comparing different models and ensured objectivity in choosing the most effective neural network classifier.

The table shows the metrics such as training time, inference time on a single sample, training accuracy, testing accuracy, and model size for each of the training sample sizes. These results help us

*ISSN 2522-9818 (print)*
*Сучасний стан наукових досліджень та технологій в промисловості. 2023. № 3 (25)* *ISSN 2524-2296 (online)*

to better understand how different models respond to different conditions and data sizes, which may indicate their effectiveness and suitability for specific text classification tasks.

To better understand the results of the study and comparative analysis of LSTM and CNN models in the context of text classification, we present Figure 4, which illustrates the key indicators of each model.

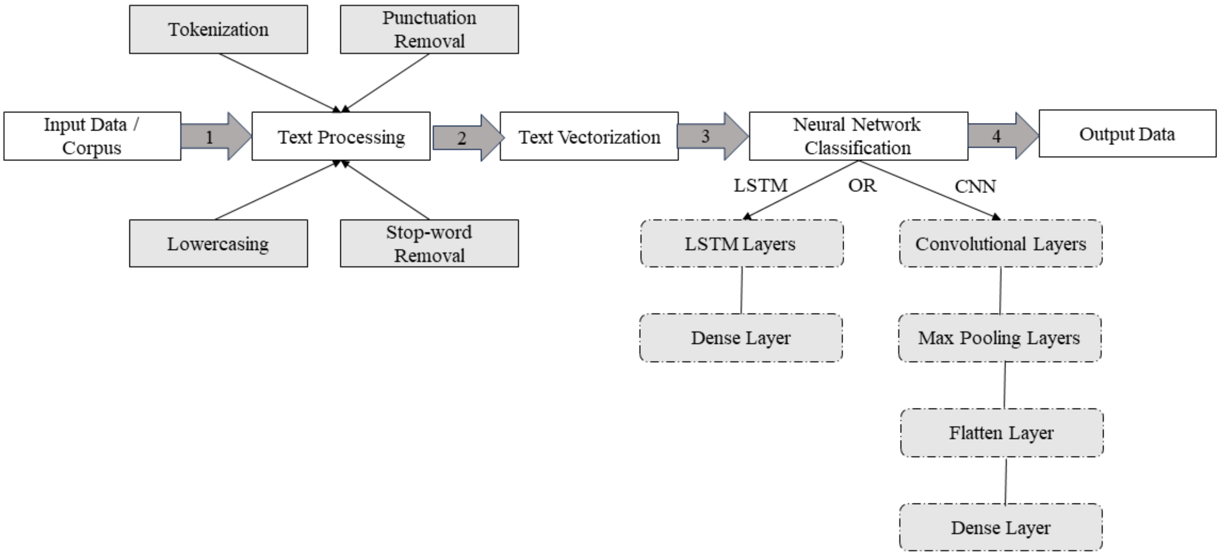


**Fig. 3.** Generalized requirements for the experiment

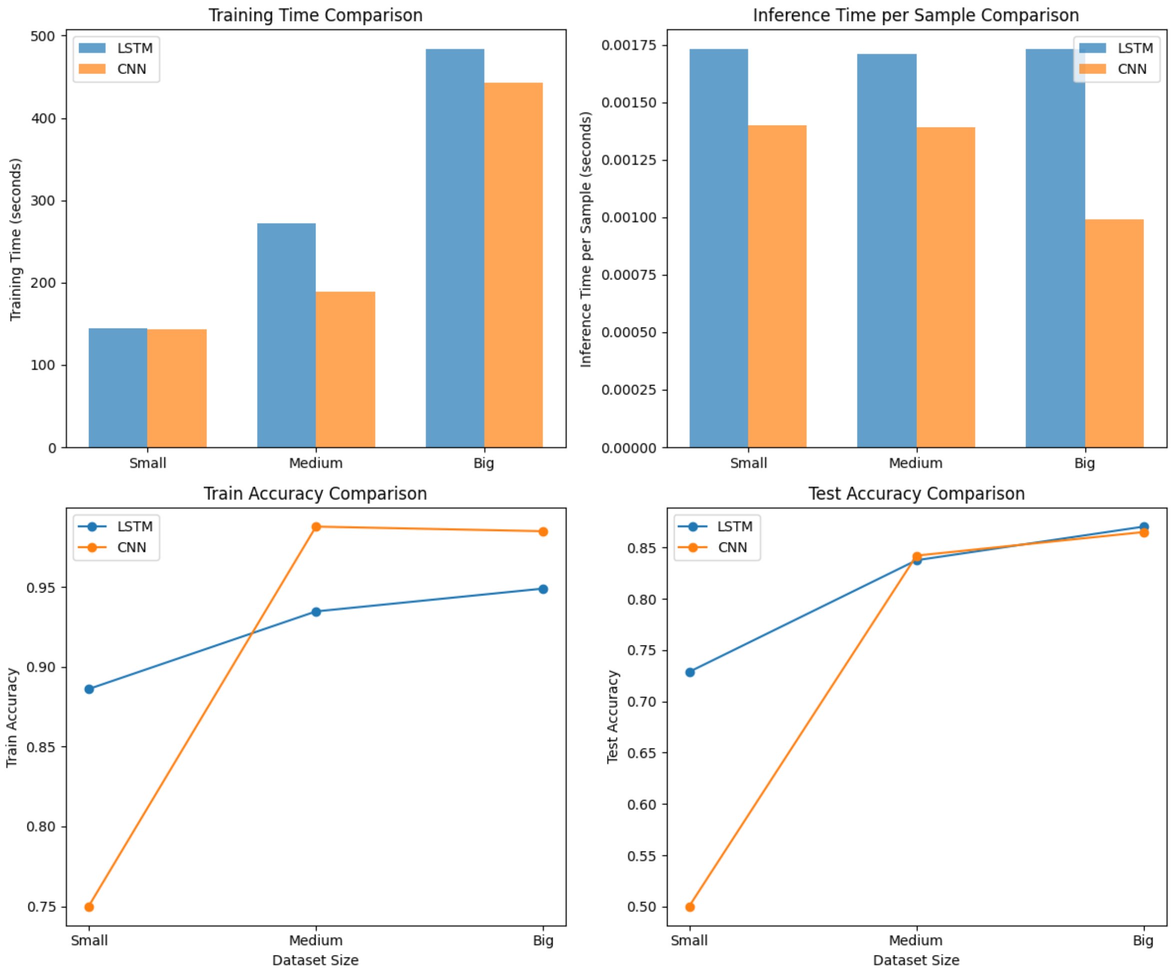


**Fig. 4.** Comparison of LSTM and CNN models on different sizes of training samples

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2023. No. 3 (25)

The graphs allow you to quickly evaluate the performance of each model based on various aspects such as training time, prediction time, and classification accuracy. The graphs show results for three training sample sizes: small, medium, and large. Each graph shows the training time and classification accuracy for LSTM and CNN models on their respective training sets. This approach allows you to compare the performance of both models depending on the size of the training set and the computing base.

The analysis of experimental results provides an understanding of the performance and properties of LSTM and CNN models in the context of text classification. It is important to note that the LSTM model demonstrates competitive accuracy at different training set sizes, which indicates its ability to analyze long-term dependencies in text data. On the other hand, the CNN model shows improved performance as the training set size increases, indicating its effectiveness in recognizing local features and patterns. In addition, different training and prediction times provide insight into the computational efficiency of each model. In general, the CNN model shows faster training times, especially with large training samples. On the other hand, the LSTM model shows stable performance with different training sample sizes. In terms of model size, both LSTM and CNN have the same number of parameters regardless of the training sample size. This aspect emphasizes their scalability for different amounts of data.

## Conclusions

Given the results of our study, we chose the Long Short-Term Memory (LSTM) model for further experiments in text classification. This decision is based on several important factors that confirm the advantages of LSTM in this context:

1. LSTM is a recurrent neural network specially designed to work with sequential data such as text. This allows it to recognize complex relationships and dependencies between words in a text, which is crucial for accurate classification.

2. One of the key advantages of LSTM is its ability to identify long-term dependencies in sequential data. This can be an important factor for text analysis, where the relationships between words can be very scattered.

3. Our experimental results show that LSTM has high classification accuracy for different training sample sizes. This demonstrates its ability to effectively learn and generalize patterns in the data.

4. The LSTM showed a stable size regardless of the size of the training samples. This can be an important aspect for further research, as it avoids unnecessary computational effort when expanding the dataset.

All these justifications confirm that LSTM is an excellent choice for solving the problem of text classification. Its ability to work with sequential data, detect long-term dependencies, and demonstrate high accuracy makes it an important tool for analyzing text data in various fields.

Taking into account the results of the experiments and the analysis of the data obtained, we can conclude that the best model for text data classification can be chosen. Our study included a comparative analysis of two popular neural network architectures – LSTM (Long Short-Term Memory) and CNN (Convolutional Neural Network) – to determine their effectiveness in solving the text classification task. Comparing the models by various characteristics, we obtained convincing results that demonstrate their advantages and disadvantages. Analyzing the classification accuracy, we found that the LSTM model performs better than CNN for all three training sample sizes. It is able to effectively adapt to different types of data and demonstrates consistently high accuracy even with large amounts of data. In addition, analyzing the impact of the computing base on the results, we found that computing power can significantly affect the performance of models. The choice of model should be justified not only by the results but also by the available resources.

So, taking into account all the data and analysis results, we can conclude that the LSTM model is more suitable for our text data classification task. Its high accuracy, training and prediction speed, and compact size make it the best choice for further research and implementation in practical applications.

## References

1. Serdechnyi, V., Barkovska, O., Rosinskiy, D., Axak, N., & Korablyov, M. (2020), "Model of the internet traffic filtering system to ensure safe web surfing". *In Lecture Notes in Computational Intelligence and Decision Making: Proceedings*

ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

*of the XV International Scientific Conference "Intellectual Systems of Decision Making and Problems of Computational Intelligence"(ISDMCI'2019),* Ukraine, May 21–25, 2019, 15 P. 133–147. Springer International Publishing. DOI: https://doi.org/10.1007/978-3-030-26474-1_10

2. Barkovska, O., Pyvovarova, D., Kholiev, V., Ivashchenko, H., & Rosinskiy, D. "Information Object Storage Model with Accelerated Text Processing Methods". In COLINS, 2021, P. 286–299, available at: https://csitjournal.khmnu.edu.ua/index.php/csit/article/download/182/112/559

3. Zhang, J. et al. (2018), "LSTM-CNN hybrid model for text classification", *IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). – IEEE.* P. 1675–1680. DOI: https://doi.org/10.1109/IAEAC.2018.8577620

4. WANG, Haitao, HE, Jie, ZHANG, Xiaohong, et al., (2020), "A Short Text Classification Method Based on N-Gram and CNN", *Chinese Journal of Electronics*, Vol. 29, No. 2, P. 248–254. DOI: https://doi.org/10.1049/cje.2020.01.001

5. Mohammad, A.H., Alwada'n, T., Al-Momani, O., (2019), "Arabic Text Categorization Using Support Vector Machine.", *Naïve Bayes and Neural Network*, P. 930–933. DOI: https://doi.org/10.5176/2251-3043_4.4.360

6. Wang, Congcong & Nulty, Paul & Lillis, David. (2020), "A Comparative Study on Word Embeddings in Deep Learning for Text Classification". *Conference on Natural Language Processing and Information Retrieval.* P. 37–46. DOI: https://doi.org/10.1145/3443279.3443304

7. Selva, Birunda S., Kanniga, Devi R. (2021), "A review on word embedding techniques for text classification", *Innovative Data Communication Technologies and Application: Proceedings of ICIDCA*. P. 267–281. DOI: https://doi.org/10.1007/978-981-15-9651-3_23

8. Mars, M. (2022), "From word embeddings to pre-trained language models: A state-of-the-art walkthrough", *Applied Sciences*. – Vol. 12. №. 17. P. 8805. DOI: https://doi.org/10.3390/app12178805

9. Patil, Rajvardhan & Boit, Sorio & Gudivada, Venkat & Nandigam, Jagadeesh. (2023), "A Survey of Text Representation and Embedding Techniques in NLP". *IEEE Access*. P 1–10. DOI: https://doi.org/10.1109/ACCESS.2023.3266377.

10. Krizhevsky, Alex & Sutskever, Ilya & Hinton, Geoffrey. (2012), "ImageNet Classification with Deep Convolutional Neural Networks". *Neural Information Processing Systems*. 25 p. DOI: https://doi.org/10.1145/3065386.

11. Wang, S. et al. (2018), "Densely connected CNN with multi-scale feature attention for text classification" *IJCAI*. Vol. 18. P. 4468–4474. DOI: https://doi.org/10.24963/ijcai.2018/621

12. Jang B. et al. (2020), "Bi-LSTM model to increase accuracy in text classification: Combining Word2vec CNN and attention mechanism". *Applied Sciences*. Vol. 10. No 17. 5841 p. DOI: https://doi.org/10.3390/app10175841

13. Gao, M., Li T., Huang P. (2019), "Text classification research based on improved Word2vec and CNN". *Service-Oriented Computing–ICSOC 2018 Workshops: ADMS, ASOCA, ISYyCC, CloTS, DDBS, and NLS4IoT*, Hangzhou, China, November 12–15, Revised Selected Papers 16. Springer International Publishing. P. 126-135. DOI: https://doi.org/10.1007/978-3-030-17642-6_11

14. Zhou, H. (2022), "Research of text classification based on TF-IDF and CNN-LSTM". *Journal of Physics: Conference Series*. *– IOP Publishing*, Vol. 2171. No 1. 2021 p. DOI: https://doi.org/10.1088/1742-6596/2171/1/012021

15. Mikolov, Tomas & Chen, Kai & Corrado, G.s & Dean, Jeffrey "Efficient Estimation of Word Representations in Vector Space", Proceedings of Workshop at ICLR. 2013. available at: https://www.researchgate.net/publication/234131319_Efficient_Estimation_of_Word_Representations_in_Vector_Space

16. Jeffrey, Pennington, Richard, Socher, and Christopher, Manning (2014), "GloVe: Global Vectors for Word Representation". *In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP),* Doha, Qatar. Association for Computational Linguistics. P. 1532–1543. DOI: https://doi.org/10.3115/v1/D14-1162.

17. Panchenko, D. et al. (2021), "Ukrainian news corpus as text classification benchmark". *International Conference on Information and Communication Technologies in Education, Research, and Industrial Applications*. *Cham: Springer International Publishing*, P. 550-559. DOI: https://doi.org/10.1007/978-3-319-76168-8

18. Schwenk, H., & Li, X. "A Corpus for Multilingual Document Classification in Eight Languages". 2018. available at: https://arxiv.org/abs/1805.09821

*Received 14.09.2023*

*Відомості про авторів / About the Authors*

**Барковська Олеся Юріївна –** кандидат технічних наук, доцент, Харківський національний університет радіоелектроніки, доцент кафедри електронних обчислювальних машин, Харків, Україна; e-mail: olesia.barkovska@nure.ua; ORCID ID: https://orcid.org/0000-0001-7496-4353

**Воропаєва Ксенія Андріївна** – Харківський національний університет радіоелектроніки, магістрантка кафедри електронних обчислювальних машин, Харків, Україна; e-mail: kseniia.voropaieva@nure.ua; ORCID ID: https://orcid.org/0009-0003-1270-4857

**Руських Олександр Валерійович** – Харківський національний університет радіоелектроніки, магістрант кафедри електронних обчислювальних машин, Харків, Україна; e-mail: oleksandr.ruskikh@nure.ua; ORCID ID: https://orcid.org/0000-0001-9931-4226

**Barkovska Olessia** – PhD (Engineering Sciences), Associate Professor, Kharkiv National University of Radio Electronics, Associate Professor of the Department of Electronic Computers, Kharkiv, Ukraine.

**Voropaieva Kseniia** – Kharkiv National University of Radio Electronics, student of Department of Electronic Computers, Kharkiv, Ukraine.

**Ruskikh Oleksandr** – Kharkiv National University of Radio Electronics, student of Department of Electronic Computers, Kharkiv, Ukraine.

# ОБҐРУНТУВАННЯ ВИБОРУ НЕЙРОМЕРЕЖНОГО ЛІНГВІСТИЧНОГО КЛАСИФІКАТОРА

**Предметом статті** є дослідження архітектури нейронних мереж для підвищення точності класифікації тексту, зокрема у сфері оброблення природної мови. Значення класифікації тексту помітно зросло в останні роки, що пов’язано з її ключовою роллю в різних програмах, зокрема аналіз налаштувань, фільтрація вмісту та категоризація інформації. З огляду на зростання попиту на точність та ефективність методів класифікації тексту, оцінювання та порівняння різноманітних моделей нейронних мереж стають обов’язковими для визначення оптимальних стратегій. **Метою дослідження** є порівняльний аналіз двох важливих архітектур нейронних мереж – довгострокової короткочасної пам’яті (*LSTM*) та згорткової нейронної мережі (*CNN*) – для формування рекомендацій щодо вибору нейромережного лінгвістичного класифікатора. Для досягнення мети були розв’язані такі **завдання**: проаналізовано проблемні сфери, зокрема обґрунтування актуальності теми, порівняння наявних текстових корпусів; сформовано критерії та вимоги до роботи нейромережних класифікаторів; проведено дослідження на вибірках різних розмірів з метою порівняння часу навчання та передбачення, точності передбачення в процесі використання нейромережних моделей *LSTM* і *CNN* як класифікатора; проаналізовано здобуті результати; обґрунтовано вибір найефективнішої моделі. Оцінювання таких параметрів, як точність класифікації, час навчання та прогнозування, розмір моделі, розподіл інформації та простота використання, надає обґрунтовані показники про переваги й недоліки кожної моделі та дає змогу дослідникам і практикам приймати рішення щодо вибору нейромережного лінгвістичного класифікатора. Застосовано такі **методи**: попереднє оброблення текстової інформації (методи токенізації, вилучення стоп-слів), векторизація тексту, нейромережні методи аналізу вхідних даних, статистичні методи. Здобуто конкретні **результати**. Модель *LSTM* продемонструвала вищу точність класифікації для всіх трьох розмірів навчальної вибірки порівняно з *CNN*. Це доводить здатність *LSTM* ефективно адаптуватися до різних типів інформації та стабільно підтримувати високу точність, навіть із великими обсягами даних. Крім того, дослідження показало, що обчислювальна потужність значно впливає на продуктивність моделі та необхідно враховувати доступні ресурси у виборі моделі. **Висновки**. З огляду на результати дослідження *LSTM*-модель визнана кращим вибором для класифікації текстової інформації. Вправність зазначеної моделі в обробленні послідовних даних, розпізнаванні довгострокових залежностей і стабільному забезпеченні високої точності позиціонує її як надійне рішення для аналізу тексту в різних сферах. Швидке навчання, ефективність прогнозування та компактний розмір моделі роблять її придатною для практичного застосування.

**Ключові слова:** класифікація тексту; нейронні мережі; *LSTM; CNN;* точність класифікації; порівняння моделей; послідовні дані.

*Бібліографічні описи / Bibliographic descriptions*

Барковська О. Ю., Воропаєва К. А., Руських О. В. Обґрунтування вибору нейромережного лінгвістичного класифікатора. *Сучасний стан наукових досліджень та технологій в промисловості*. 2023. № 3 (25). С. 5–14. DOI: https://doi.org/10.30837/ITSSI.2023.25.005

Barkovska, O., Voropaieva, K., Ruskikh, O. (2023), "Justifying the selection of a neural network linguistic classifier", *Innovative Technologies and Scientific Solutions for Industries*, No. 3 (25), P. 5–14. DOI: https://doi.org/10.30837/ITSSI.2023.25.005