Research Article

Performance Comparison Analysis on Weather Prediction using LSTM and TKAN

[1*]Ajie Kusuma Wardhana, [1]Yudha Riwanto, [2]Budi Wijaya Rauf

[1] Amikom University, Yogyakarta, Indonesia
[2] Universitas Halu Oleo, Kendari City, South East Sulawesi, Indonesia

* Corresponding Author: ajiekusuma@amikom.ac.id

Abstract: The development of machine learning methods in the last few decades has shown great potential in various predictive applications, including in domains such as financial prediction, medical diagnosis, and big data analysis. One of the most widely used methods in prediction tasks is Long Short-Term Memory (LSTM). LSTM has become popular because of its ability to handle time series data by retaining relevant information in the long term and the ability to forget irrelevant information through the forget-gate mechanism. However, along with the development of technology and the need to improve accuracy and efficiency, new methods such as the Kolmogorov Arnold Network (KAN) have emerged. KAN was then developed into the Temporal Kolmogorov Arnold Network (TKAN), which was designed to match or even surpass the performance of LSTM. The TKAN architecture has produced significant improvements in the management of both new and historical information. Because of this capability, TKAN can excel in multi-step predictions, demonstrating a clear advantage over conventional models such as LSTM and GRU, particularly in the context of long-term forecasting. This research goal is to give insight into the comparison of both the TKAN and LSTM models for weather prediction using model loss and mean absolute error evaluation (MAE). The model for both LSTM and TKAN achieved 0.09 and 0.11 for model loss and 0.08 and 0.96 for MAE.

Keywords: LSTM; Temporal Kolmogorov Arnold Network; Prediction; Performance comparison; TKAN

check for updates

Citation: A.K.Wardhana, Y.Riwanto, B.W.Rauf, "Performance Comparison Analysis on Weather Prediction using LSTM and TKAN", *Iota*, **2024**, ISSN 2774-4353, Vol.04, 03. https://doi.org/10.31763/iota.v4i3.808

Academic Editor : Adi, P.D.P

Received : July, 22 2024

Accepted : July, 25 2024

Published : August, 16 2024

1. Introduction

The development of machine learning methods in the last few decades has shown great potential in various predictive applications, including in domains such as financial prediction, medical diagnosis, and big data analysis. One of the most widely used methods in prediction tasks is Long Short-Term Memory (LSTM). LSTM has become popular because of its ability to handle time series data by retaining relevant information in the long term and the ability to forget irrelevant information through the forget-gate mechanism [1], [2]. These advantages make LSTM more effective compared to other methods, especially in overcoming the vanishing gradient problem often faced by traditional recurrent neural networks.

However, along with the development of technology and the need to improve accuracy and efficiency, new methods such as the Kolmogorov Arnold Network (KAN) [3, 17, 18, 19, 20] have emerged. KAN was then developed into the Temporal Kolmogorov Arnold Network (TKAN), which was designed to match or even surpass the performance of LSTM. TKAN is made with a similar architecture to LSTM, including the forget-gate mechanism, but with the integration of kernel attention that can capture temporal information more effectively [4].

In addition, recent approaches such as the use of Transformer-based models for time series forecasting have also yielded promising results, indicating that the integration of attention techniques can improve prediction accuracy, especially in highly variable and dynamic data [5]. This study aims to review the performance of these two algorithms, namely LSTM and TKAN, both in terms of computational load and evaluation of prediction errors. Thus, it is expected to obtain deeper insights into the advantages and limitations of each method, as well as recommendations for the use of the most appropriate algorithm according to the context and needs of the application.

2. Literature Review

In the era of deep learning, the development of advanced algorithms for temporal data modeling has gained significant momentum. Temporal Knowledge Attention Networks (TKAN) and Long Short-Term Memory (LSTM) networks are among the most prominent architectures employed for handling complex sequence prediction tasks. This review focuses on the evolution, applications, and comparative analysis of TKAN and LSTM algorithms.

2.1 Long-Short Term Memory

Long Short-Term Memory (LSTM) networks have maintained their position as a cornerstone in temporal data analysis. The core design of LSTM, which includes forgetting, input, and output gates, has been refined to improve the handling of long-term dependencies and mitigate issues such as the vanishing gradient problem [1].

LSTM has continued to play a pivotal role in NLP tasks, particularly in sentiment analysis, text generation, and machine translation. In 2020, Huang et al. demonstrated that LSTM models combined with attention mechanisms could significantly enhance the performance of sentiment classification models by emphasizing contextually relevant information within textual data[6, 11, 12, 13, 14, 15, 16]. Furthermore, Liu et al. (2021) utilized LSTM networks for automatic text summarization, leveraging their ability to understand and generate coherent summaries from large text corpora[7].

LSTM's application in financial forecasting has also expanded, particularly through hybrid models that combine LSTM with other techniques to improve prediction accuracy. Zhang et al. (2020) introduced a hybrid LSTM-ARIMA model for stock price prediction, which outperformed traditional models by capturing both linear and nonlinear patterns in financial time series data [8]. A further improved this approach by incorporating a Bayesian optimization framework, which dynamically adjusted the LSTM hyperparameters to optimize performance across different market conditions[9].

2.2 Temporal Kolmogorov-Arnold Network

The architecture of the Kolmogorov-Arnold Network (KAN) for time series forecasting incorporates both recurring and gating mechanisms to improve stability and performance. The authors developed temporal Kolmogorov-Arnold Networks (TKANs), which combine the strengths of recurrent neural networks (RNNs) with those of Kolmogorov-Arnold Networks, and effectively address the long-term dependency problems that traditional RNN models often face. Using Recurrent Kolmogorov-Arnold Networks (RKANs), the TKAN architecture has produced significant improvements in the management of both new and historical information. Because of this capability, TKAN can excel in multi-step predictions, demonstrating a clear advantage over conventional models such as LSTM and GRU, particularly in the context of long-term forecasting.

When applied to real historical market data, TKAN is more stable and performs better than GRU and LSTM. However, TKAN may not be as effective for short-term predictions, but it significantly outperforms existing models in multi-step forecasting. These results confirm the effectiveness of the KAN framework in practical time series applications and suggest that TKAN provides valuable advancements in the accuracy and robustness of long-term forecasting.

Finally, the results demonstrate the ability of TKAN to solve complex problems in temporal prediction. These findings also pave the way for the improvement and application of this architecture in various time series analysis scenarios [10].

3. Methodology

In this research, to achieve the result, several methods are required. this research follows an experimental design where both models are trained and evaluated using the same dataset.

3.1 Data Collection

The dataset used for this research is acquired from the Kaggle website. It consists of weather data from 1980 to 2024. The data features that are used for training are temperature, relative humidity, and precipitation. Hence, the goal is to predict the rainfall based on the features we have selected.

3.2 Data Preprocessing

Before using the data to be trained, pre-processing data is done.

1. Data Filtering
   The dataset is filtered which only involves entries up to 2023-12-31 in y-m-d format.
2. Normalization
   Min-max scaler normalization is applied to the selected features which scale the data to a range of [0,1].
3. Sequence Creation
   The features are converted into a sequence of lengths of 20
4. Splitting Data
   The dataset split ratio for train and testing with 80% and 20% of the data.

3.3 Model Implementation

In this research, the Model Implementation process that we use uses Keras TensorFlow for both the LSTM and TKAN model. The description is as follows:

3.3.1 TKAN (Temporal Kolmogorov Arnold Network)

The Kolmogorov Arnold Networks model can described as following equation 1.

$$f(x) = \sum_{i=1}^{n} w_i \phi(xi) \tag{1}$$

Where w_i are weights, x_i are inputs, and $\phi(x_i)$ are activation functions.

Furthermore, to incorporate temporal dependencies, the KAN is extended by adding a temporal component. This involves using recurrent connections to capture long-term dependencies. The equation is following equation 2.

$$h_t = f(h_{t-1}, x_t) \tag{2}$$

Where h_t is the hidden state at time t, x_t is the input at time t., and the final output y_t is calculated based on the hidden state h_t. y_t can be written as equation 3. Where g is an output activation function.

$$y_t = g(h_t) \tag{3}$$

3.3.2 Long-Short Term Memory (LSTM)

The LSTM model can described as following equation 4

$$c_t = f_t \odot c_{t-1} + i_t \odot \breve{c}_t \tag{4}$$

Where c_t is the cell state at time t, f_t is the forget gate, i_t is the input gate, and $\tilde{c}_t$ is the candidate cell state. h_t can be written in equation 5.

$$h_t = o_t \odot \sigma(c_t) \tag{5}$$

Where h_t is the hidden state at time t, and o_t is the output gate. Then y_t can be written as equation 6.

$$y_t = g(h_t) \tag{6}$$

Where g is an output activation function. Finally, In this research, the performance matrices to be used are model loss and Mean Absolute Error.

4. Result and Discussion

For both models, the same configuration is applied to see the differences. The configuration is shown in Table 1.

Table 1. Training Setting

Hyperparameter	LSTM	TKAN
Optimizer	Adam	Adam
Epoch	50	50
Batch Size	64	64
Callback	Early stopping	Early stopping

Based on Table 1, the hyperparameter used is a model optimizer using Adam Optimizer, the epoch is 50, and the batch size is 64. While training the model will be stopped if it reaches the optimal point using the early stopping method.

4.1 LSTM Model Loss and TKAN Model Loss

From the test results, For the LSTM model, the result of model loss is in Figure 1. Moreover, the mean absolute error results are shown in Figure 2.

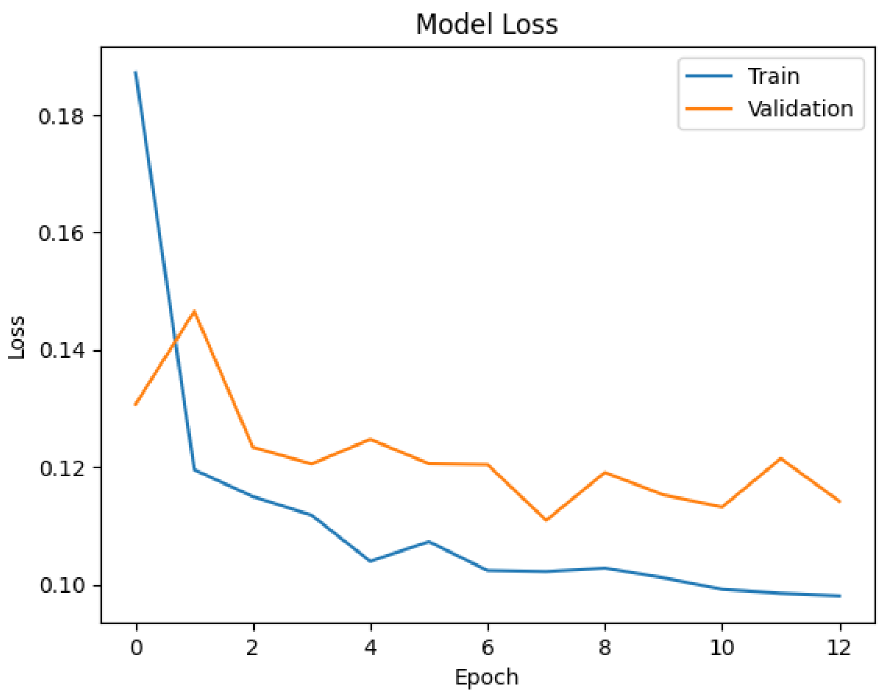

Figure 1. LSTM Model Loss

Based on Figure 1, the initial loss is around 0.18, also showing a sharp decline within the first few epochs. The model also shows fast stabilization, reaching a consistent loss value after epoch 4 and maintaining sTable performance throughout the training process.

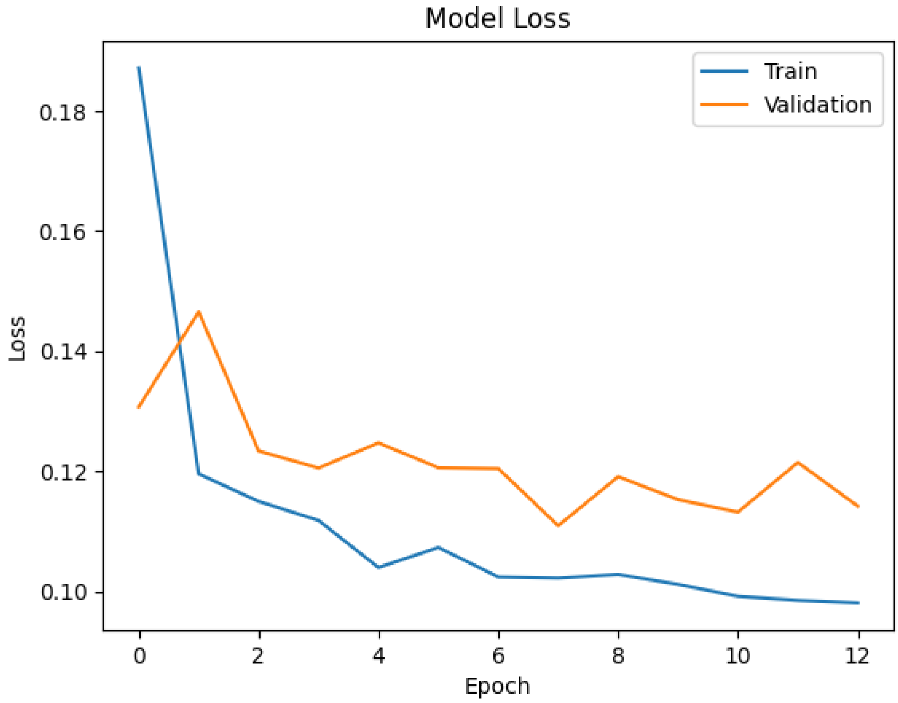

Figure 2. TKAN Model Loss

Based on Figure 2, the initial loss starts higher around 0.2 but rapidly decreases as training progresses. The model also showed improvement in loss reduction in the first few epochs, reaching a more stable loss curve around epoch 5. However, fluctuations are observed in the validation loss after epoch 10. The final training loss stabilizes at around 0.12, with some oscillations in the validation loss.

Table 2. Performance Comparison for Model Loss

Model	Model Loss Train	Model Loss Validation
LSTM	0.09	0.11
TKAN	0.11	0.17

Based on Figure 3, the model loss comparison for both models using model loss evaluation, the LSTM outperforms the TKAN model in terms of overall model loss. Hence, LSTM shows a faster convergence and better generalization. While TKAN shows potential for loss reduction.

4.2 LSTM MAE and TKAN MAE

The final comparison is to compare the Mean Absolute Error for both LSTM and TKAN models. The results are shown in Figure 3 and Figure 4.

Table 3. MAE Comparison

Model	MAE Training	MAE Validation
LSTM	0.08	0.07
TKAN	0.96	0.10

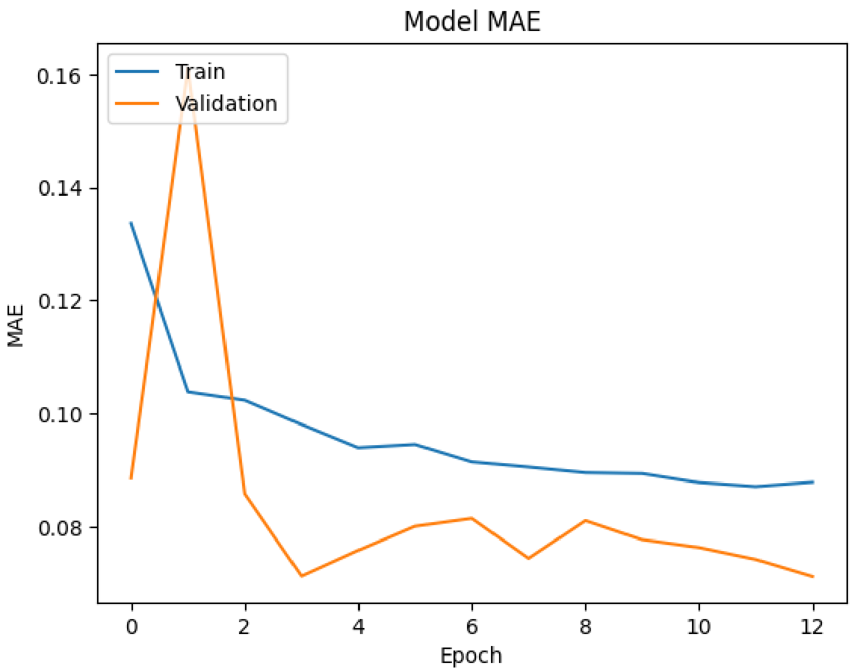

Figure 3. LSTM MAE

Based on Figure 3, the initial value from MAE is around 0.13 for training and 0.09 for validation. The model shows a peak for validation MAE of around 0.16 for the model. It is also showing a decrement for both training and validation while training. Although the model shows a significant decrease in MAE value, it shows an overfitting for the model.

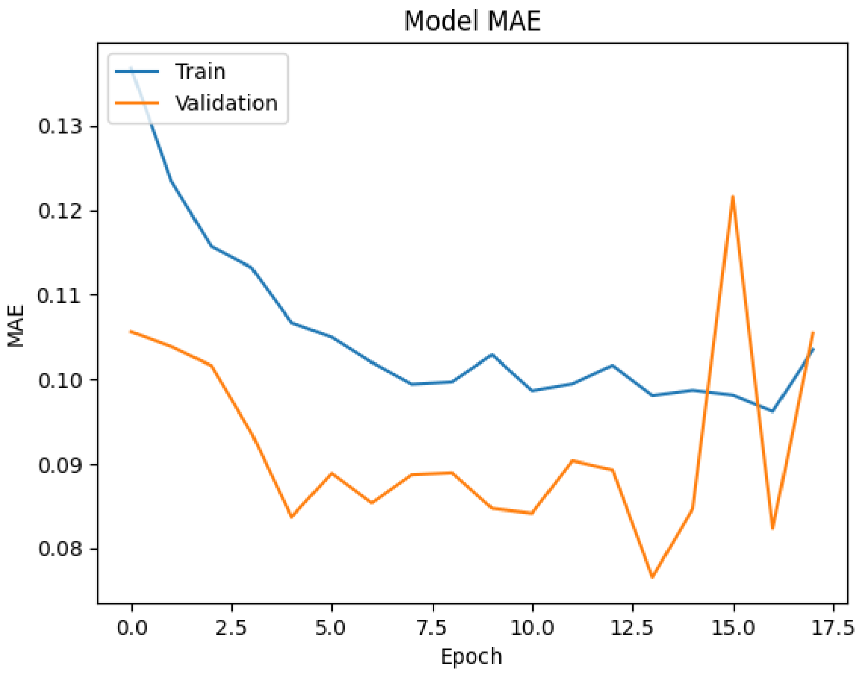

Figure 4. TKAN MAE

Based on Figure 4, the initial state for both training and validation values shows a higher initial state for validation compared to the LSTM model. While the training process occurred, the model showed a peak at epoch 15. Moreover, Based on Table 3, the LSTM still outperforms the TKAN model. It also shows a better optimal result in MAE since the epoch for the LSTM is less than the TKAN model.

5. Conclusion

In conclusion, the result shows that LSTM is better to use for prediction tasks rather than the TKAN model. This result occurred based on the architecture of LSTM having a forget gate while the TKAN model has only recurrent architecture at the final node of the model before output nodes. Hence, the LSTM model can be more suitable for the weather prediction. For further implementation, a hyperparameter tuning for the research needs to be implied to achieve a better performance for both models. Advanced pre-processing and data analysis also need to be implemented to avoid model overfitting.

Acknowledgments: Thanks to the entire research team and lecturers at Amikom University, Yogyakarta, Indonesia and also Halu Oleo University, Kendari City, South East Sulawesi, Indonesia who have helped in the completion of this article, hopefully this article can be a good tool in the development of science, computerization, especially the field of Artificial Intelligence (AI) development in Indonesia.

Author contributions: All authors are responsible for building Conceptualization, Methodology, analysis, investigation, data curation, writing—original draft preparation, writing—review and editing, visualization, supervision of project administration, funding acquisition, and have read and agreed to the published version of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yu Y, Si X, Hu C, Zhang J. A review of recurrent neural networks: LSTM cells and network architectures. Neural computation. 2019 Jul 1;31(7):1235-70.
2. Staudemeyer RC, Morris ER. Understanding LSTM--a tutorial into long short-term memory recurrent neural networks. arXiv preprint arXiv:1909.09586. 2019 Sep 12.M. Shell. (2002) IEEEtran homepage on CTAN. [Online]. Available: http://www.ctan.org/tex-archive/macros/latex/contrib/supported/IEEEtran/
3. Liu Z, Wang Y, Vaidya S, Ruehle F, Halverson J, Soljačić M, Hou TY, Tegmark M. Kan: Kolmogorov-arnold networks. arXiv preprint arXiv:2404.19756. 2024 Apr 30."PDCA12-70 datasheet," Opto Speed SA, Mezzovico, Switzerland
4. Genet R, Inzirillo H. Tkan: Temporal kolmogorov-arnold networks. arXiv preprint arXiv:2405.07344. 2024 May 12.
5. Wang C, Chen Y, Zhang S, Zhang Q. Stock market index prediction using deep Transformer model. Expert Systems with Applications. 2022 Dec 1;208:118128.
6. H. Heidari, R. Abdar, and S. P. Hariri, "Attention-Based Residual LSTM for Time Series Forecasting," IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 12, pp. 5959-5969, 2021.
7. Y. Liu, M. Ott, N. Goyal, et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," arXiv preprint arXiv:1907.11692, 2019.
8. Y. Zhang, X. Xue, and Y. Liu, "Hybrid LSTM-ARIMA Model for Stock Price Prediction," Journal of Forecasting, vol. 39, no. 8, pp. 1209-1220, 2020.
9. Q. Sun, L. Duan, and J. Gao, "Bayesian Optimization for Hyperparameter Tuning of LSTM Models in Financial Market Prediction," IEEE Access, vol. 9, pp. 116623-116634, 2021.
10. Genet R, Inzirillo H. Tkan: Temporal kolmogorov-arnold networks. arXiv preprint arXiv:2405.07344. 2024 May 12.
11. S. Prakash, A. S. Jalal and P. Pathak, "Forecasting COVID-19 Pandemic using Prophet, LSTM, hybrid GRU-LSTM, CNN-LSTM, Bi-LSTM and Stacked-LSTM for India," 2023 6th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 2023, pp. 1-6, doi: 10.1109/ISCON57294.2023.10112065.

12. A. Diaaeldin and M. Zaher, "Enhancing Road Safety: Leveraging CNN-LSTM and Bi-LSTM Models for Advanced Driver Behavior Detection," 2024 Intelligent Methods, Systems, and Applications (IMSA), Giza, Egypt, 2024, pp. 416-422, doi: 10.1109/IMSA61967.2024.10652785.
13. S. Chakraborty, J. Banik, S. Addhya and D. Chatterjee, "Study of Dependency on number of LSTM units for Character based Text Generation models," 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA), Gunupur, India, 2020, pp. 1-5, doi: 10.1109/ICCSEA49143.2020.9132839.
14. A. Joshi, P. K. Deshmukh and J. Lohokare, "Comparative analysis of Vanilla LSTM and Peephole LSTM for stock market price prediction," 2022 International Conference on Computing, Communication, Security and Intelligent Systems (IC3SIS), Kochi, India, 2022, pp. 1-6, doi: 10.1109/IC3SIS54991.2022.9885528.
15. T. Baihaqi, M. A. Sugiyarto, R. P. Daksa, F. I. Kurniadi, M. Fakhruddin and H. Erandi, "Unveling the Precision of Deep Learning Models for Stock Price Prediction: A Comparative Analysis of Bi-LSTM, LSTM, and GRU," 2023 International Conference on Converging Technology in Electrical and Information Engineering (ICCTEIE), Bandar Lampung, Indonesia, 2023, pp. 61-64, doi: 10.1109/ICCTEIE60099.2023.10366646.
16. K. S. Gill, V. Anand, R. Chauhan, A. Choudhary and R. Gupta, "CNN, LSTM, and Bi-LSTM based Self-Attention Model Classification for User Review Sentiment Analysis," 2023 3rd International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON), Bangalore, India, 2023, pp. 1-6, doi: 10.1109/SMARTGENCON60755.2023.10442498.
17. S. Gui, J. Wang, C. Tang and J. Zhao, "High precision frequency source anomaly monitor using Kolmogorov-Arnold Networks," 2024 22nd International Conference on Optical Communications and Networks (ICOCN), Harbin, China, 2024, pp. 1-2, doi: 10.1109/ICOCN63276.2024.10648537.
18. L. Pang, J. Li, J. Gan and J. Zhang, "Dissolved Gas Analysis of Transformers Based on Kolmogorov-Arnold Network," 2024 6th International Conference on Energy Systems and Electrical Power (ICESEP), Wuhan, China, 2024, pp. 1568-1571, doi: 10.1109/ICESEP62218.2024.10652040.
19. G. Andrei and M. Vyacheslav, "Analysis of approaches to the universal approximation of a continuous function using Kolmogorov's superposition," 2019 International Conference on Engineering and Telecommunication (EnT), Dolgoprudny, Russia, 2019, pp. 1-4, doi: 10.1109/EnT47717.2019.9030591.
20. V. Usatyuk, "Wireless Channels Topology Invariant as Mathematical Foundation of Neural Network Channel Estimation Transfer Learning Properties," 2020 43rd International Conference on Telecommunications and Signal Processing (TSP), Milan, Italy, 2020, pp. 105-111, doi: 10.1109/TSP49548.2020.9163528.