WEP Warwick Evans Publishing

International Journal of Computer Science and Information Technology

ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 2, Number 3, Year 2024
DOI: https://doi.org/10.62051/ijcsit.v2n3.05
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Cybersecurity Situational Awareness Model Using Improved LSTM-Informer

Xin Zhou*, Bo Li

College of Computer Science and Engineering, Chongqing University of Technology, Chongqing 400054, China

*Corresponding Author: Xin Zhou

ABSTRACT

To address the problem of low prediction accuracy in current network security situational prediction models, this study proposes an improved LSTM-Informer model. The study first employs the Empirical Mode Decomposition (EMD) technique to preprocess network security situational data, extracting stable Intrinsic Mode Functions (IMFs). These IMFs are then used as inputs to the Long Short-Term Memory (LSTM) model, which effectively enhances the accuracy and stability of LSTM in handling time series data. Secondly, this paper introduces a coding-decoding mechanism that is based on Lightweight Gradient Boosting Machine (LGB) to efficiently encode time series data. The encoded data is then fed into the Informer model, which optimizes the flow of information between LSTM and Informer, significantly improving the model's ability to predict future network security situational trends. Finally, the Particle Swarm Optimization (PSO) algorithm is utilized to optimize model parameters, enhancing prediction accuracy. The experimental results indicate that the proposed model outperforms in terms of evaluation metrics such as Mean Square Error and Mean Absolute Error, with a best-fit rate of 99.36%. When applied to real network situational data from the China National Vulnerability Database (CNVD), the model demonstrates good predictive performance, effectively addressing the issue of low accuracy in network security situational prediction.

KEYWORDS

Cybersecurity, Situation Prediction, Long Short-Term Memory, Informer, Empirical Mode Decomposition, Lightweight Gradient Boosting Machine

1. INTRODUCTION

The increasing dependence on the Internet has made network security a significant concern. However, traditional defence techniques have limitations. To tackle this issue, network security situational awareness has emerged. It facilitates the understanding of data patterns, prediction of trends, and timely detection of threats for stable network operations.

Ke et al. [1] integrated the Artificial Bee Colony (ABC) algorithm and Support Vector Machine (SVM) but faced high training costs for large-scale datasets. Wang [2] compared various methods and found that the combination of the PSO algorithm and Radial Basis Function (RBF) neural network performed well. Yuan [3] improved the PSO algorithm for optimizing RBF neural networks. Tang et al. [4] enhanced the PSO algorithm with a particle stagnation disturbance strategy. Chen et al. [5] proposed a method for time series analysis based on LSTM networks. Li et al. [6] constructed a sparse three-layer prediction model using LSTM. Lin et al. [7] combined Convolutional Neural Networks (CNN) with Bidirectional Gated Recurrent Units (BiGRU). He et al. [8] integrated GRU

with an attention mechanism and used PSO for hyperparameter optimization. Li et al. [9] combined the multi-head self-attention mechanism with Transformer and various models. Despite the promising results of LSTM, GRU, and attention mechanisms, the computational complexity of the Transformer model remains a challenge for large-scale long sequences.

This paper introduces a new model for predicting network security situations. The model is based on the LSTM-Informer architecture and uses the PSO algorithm to fine-tune its parameters. To enhance the accuracy and stability of the model for time series data processing, EMD technology is applied to preprocess network security data. This extracts stable IMF as the input of the LSTM module. Additionally, this model optimises the information transmission process between LSTM and Informer by introducing the encoding and decoding mechanism of LGB. This results in a significant improvement in the accuracy of network security situation prediction. The study integrates traditional statistical methods, machine learning, and deep learning techniques to enhance the model's stability and generalisation ability.

2. CYBERSECURITY SITUATION PREDICTION MODEL BASED ON IMPROVED LSTM-INFORMER

The model for predicting network security situations based on LSTM-Informer consists of three main components: LSTM layer, Informer layer, and fully connected layer. Fig. 1 illustrates the specific structure of the model.

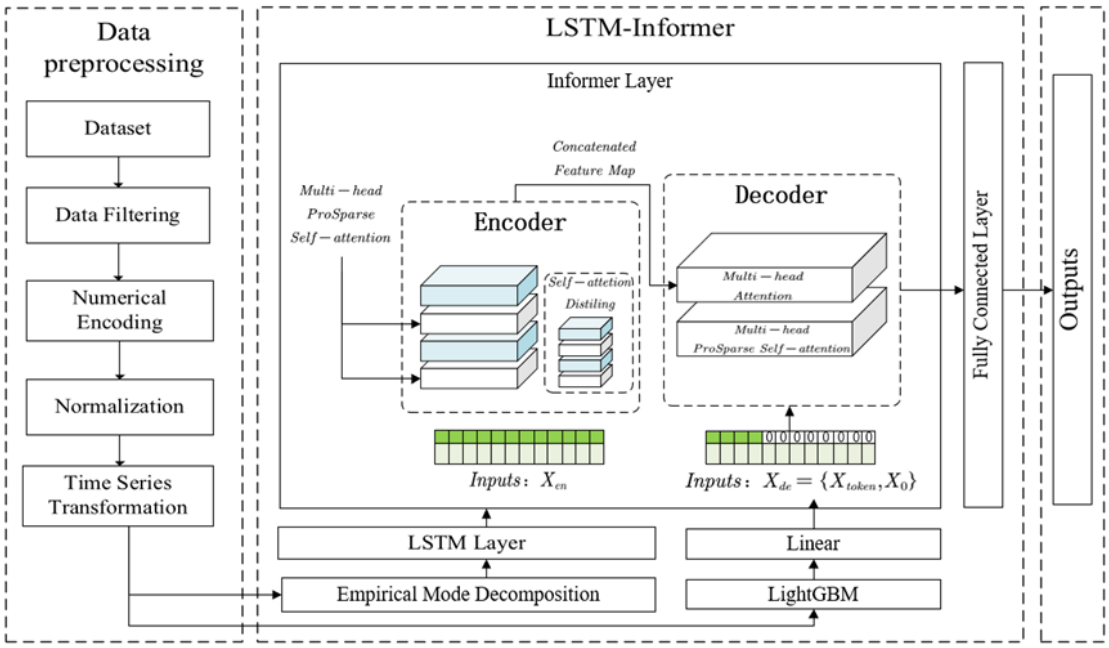

Figure 1. Network model based on Improved LSTM-Informer prediction method

The model is based on the LSTM-Informer architecture. The architecture first employs the EMD technique to preprocess network security-related data. This technique extracts stable IMFs as inputs for the LSTM module. Additionally, the model optimizes the information transmission process between LSTM and Informer by introducing the encoding-decoding mechanism of LGB. This optimization significantly improves the accuracy of network security situation prediction.

2.1. LONG SHORT-TERM MEMORY NETWORK

In 1997, Sepp Hochreiter et al. [10] introduced LSTM, a specialized type of RNN that tackles the issues of gradient vanishing and exploding in long sequence training. LSTM consists of memory cells,

each equipped with three gating units: the forget gate, input gate, and output gate. These gates control the flow of information within the cells.

The forgetting gate f_t is crucial for controlling information transfer from the previous memory unit C_{t-1} to the current memory unit C_t. It uses a sigmoid activation function to limit its output range to [0, 1]. Equation (1) provides the calculation formula for the forgetting gate, and Equation (2) provides the calculation formula for the memory unit.

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{1}$$

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \tag{2}$$

The input gate, represented by i_t, is responsible for controlling the amount of input information x_t at the current time step that is added to the memory unit. Equation (3) and Equation (4) are its calculation formulas.

$$i_t = \sigma(W_t \cdot [h_{t-1}, x_t] + b_i) \tag{3}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{4}$$

The output gate, represented by o_t, plays a crucial role in determining the amount of information output from the current memory unit. Equation (5) and Equation (6) represent its calculation formulas.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t \otimes \tanh(C_t) \tag{6}$$

W and b in the formula represent the weight matrix and bias term, respectively.

2.2. INFORMER

The Informer model was proposed in a paper titled "Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting" [11]. It is an improved version of the Transformer model, incorporating three main enhancements: Prob-Sparse Self-Attention, Self-Attention Distilling, and Generative Decoder.

2.2.1. ProbSparse Self-Attention

ProbSparse Self-Attention is an innovative self-attention mechanism in the In-former model that addresses the computational inefficiency of traditional self-attention. It introduces sparsity to reduce the time complexity from $O(L^2)$ to $O(L \log L)$, resulting in improved computational efficiency. The primary formula for ProbSparse Self-Attention is Equation (7).

$$A(Q,K,V) = Soft\max\left(\frac{\bar{Q}K^T}{\sqrt{d}}\right)V \tag{7}$$

The matrices and have the same size and are sparse, with only the elements corresponding to the top u dominant queries being retained, while the rest of the elements are set to zero. The definition of the approximate query sparsity evaluation is given by Equation (8).

$$\bar{M}(q_i,K) = \max_j\left\{\frac{q_i k_j^T}{\sqrt{d}}\right\} - \frac{1}{L_K}\sum_{j=1}^{L_k}\frac{q_i k_j^T}{\sqrt{d}} \tag{8}$$

ProbSparse Self-Attention enables each head to utilize a unique optimization strategy, thereby enhancing the flexibility and precision of attention allocation.

2.2.2. Self-Attention Distilling

Self-Attention Distilling optimizes self-attention by prioritizing dominant features, reducing redundancy. Equation (9) represents this process.

$$X_{j+1}^t = MaxPoll\left(ELU\left(Conv1d\left([X_j^t]_{AB}\right)\right)\right) \tag{9}$$

By incorporating Self-Attention Distilling, the Informer model not only achieves a commendable performance level but also effectively reduces computational overhead, resulting in improved model efficiency.

2.2.3. Generative Decoder

The Generative Decoder utilizes multi-head self-attention modules to capture complex dependencies and contextual cues, facilitating accurate generation of the target sequence. Equation (10) represents the mathematical expression for the Generative Decoder.

$$X^{t}_{feed\_de} = Concat(X^{t}_{token}, X^{t}_{0}) \in R^{(L_{token}+L_{y})\times d_{model}} \tag{10}$$

The Generative Decoder combines self-attention mechanisms and feed-forward neural network layers to integrate contextual information and previously generated target sequences. This integration enables accurate and distinctive predictions for the current time step by capturing dependencies and incorporating con-textual cues.

2.3. EMPIRICAL MODE DECOMPOSITION

A new adaptive signal time-frequency processing method called EMD, creatively proposed by N. E. Huang and others at NASA in 1988 [12]. EMD is an adaptive time series data analysis method for decomposing complex data into a series of IMF. Each IMF represents a different frequency component of the data, allowing the characteristics of the original signal to be represented at different scales. This decomposition can help the model to better understand and capture the essential features of the time series, while removing noise and reducing the interference of random fluctuations in the data on the prediction results. The adaptive nature of EMD makes it particularly suitable for the analysis of nonlinear and non-stationary time series.

The consideration of including EMD in the LSTM-Informer network posture prediction model is based on the advantages of EMD in dealing with nonlinear and non-stationary time series. EMD can adaptively decompose complex data into a series of IMFs, each of which captures a different frequency component of the original data, thus revealing the intrinsic dynamic characteristics of the data.

The network security posture signal components after EMD decomposition are shown in Fig. 2, forming five network security posture value components, including four network security IMF posture value components and one network security residual signal component. The decomposition process of the cybersecurity posture values is that the original cybersecurity posture signals are continuously differed from the mean value of the envelope and the original cybersecurity posture values are replaced by the residual cybersecurity posture values, and it can be seen that the frequency of the IMF signals formed as a result decreases with the decrease of the frequency of the envelope.

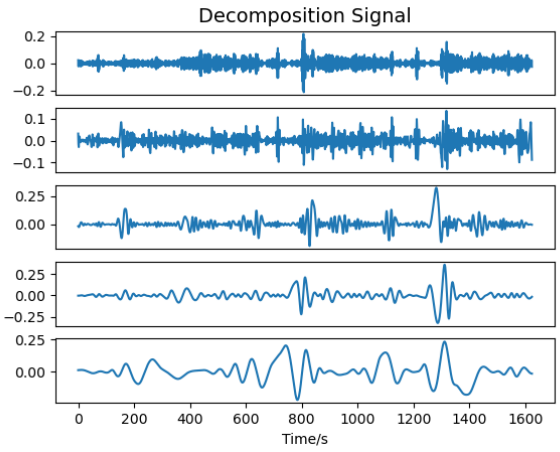

Figure 2. Situational Signal Component Images.

This decomposition can provide more accurate and stable inputs to the LSTM model, enhance the model's ability to capture the changing trends of time series data, and improve the prediction accuracy of the network security posture. By analysis of the essential components of the data, EMD enables the model to learn and predict more effectively, thus improving the overall model performance.

2.4. LIGHTWEIGHT GRADIENT BOOSTING MACHINE

LGB is a boosting framework proposed by Microsoft Research Asia (MSRA) in 2017 with a paper titled A Highly Efficient Gradient Boosting Decision Tree [13]. LGB is a gradient boosting framework for tree-based learning algorithms, which can effectively process and extract features in nonlinear relationships and complex patterns, which is particularly important for the analysis of time series data. Incorporating LGB enhances the model's ability to capture the intrinsic patterns of the time series, thus improving the accuracy of the prediction. It is especially optimal for speed and efficiency, and is suitable for processing large-scale data. The introduction of LGB into the LSTM-Informer network posture prediction model is mainly aimed at exploiting the efficiency of LGB in time series coding, thus improving the quality of model inputs. By using the LGB model to encode the time series and using this encoding as the input to the Informer decoder, it not only enhances the model's ability to capture the complex patterns of the data, but also improves the model's performance and efficiency in dealing with large dimensional data, and ultimately significantly improves the accuracy and genericity of the cybersecurity situational prediction.

The aim of this approach is to improve the performance of the LSTM-Informer model in the task of cybersecurity situational prediction by exploiting the efficiency and predictive power of the LGB.

2.5. PARTICLE SWARM OPTIMIZATION

The PSO algorithm, proposed by James Kennedy et al. in 1995 [14], is a method that achieves global optimization and fast convergence by simulating the movement and information exchange of particles in the solution space. The specific steps for parameter optimization of the PSO algorithm in this experiment are as follows:

Step 1: PSO Parameter Configuration. Configure the parameters of the PSO algorithm as follows: set the inertia weight to 1.2, the individual learning factor to 0.8, the social learning factor to 0.7, the number of iterations to 20, and the number of particles to 5.

Step 2: Model Parameter Search Range Setting. Define the search range for the model parameters.

Step 3: Generating Population Particles. Generate particles representing the population, with each particle representing a combination of parameters (d_model, l_layers, e_layers, d_layers, seq_len, factor, n_heads).

Step 4: Setting Particle Fitness Function. Define the fitness function for each particle based on the model's loss function.

Step 5: Initializing Particle Positions and Fitness Values. Set the initial fitness values as the local best solutions for each particle. Compute and store the best position for each particle at each iteration.

Step 6: Updating Particle Velocity and Position. Update the velocity and position of each particle using Equation (11) and Equation (12).

$$v_{id}^{k+1} = wv_{id}^{k} + c_1 r_1 (p_{id,pbest}^{k} - x_{id}^{k}) + c_2 r_2 (p_{d,gbest}^{k} - x_{id}^{k}) \tag{11}$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \tag{12}$$

Step 7: Updating Best Fitness Value and Termination. Continue iterating and updating the best fitness value until reaching the end of the specified number of iterations. Finally, return the best parameter combination.

3. EXPERIMENTS AND ANALYSIS

3.1. DATASET

Using the UNSW-NB15 dataset [15], managed by the Network Security research team at the University of New South Wales, the experiment focused on the sub-sets UNSW-NB15_1.csv and UNSW-NB15_3.csv, totaling 1.4 million instances.

The situational value for each time period was calculated based on the situational index system from Reference [16], considering the attack count factor N and the attack threat factor X_i. The resulting situational value is represented by the variable S, was determined using the formula in Equation (13).

$$S(t) = f(N, X_i) = \sum_{i=1}^{N} X_i \tag{13}$$

Distinguishing attack types exhibit distinct values for the attack threat factor, as indicated in Table 1 for reference.

Table 1. Attack threat factor.

Attack Types	Attack Threat Factor
Normal	0
Analysis	1
Reconnaissance	2
DoS	3
Fuzzers	4
Generic	5
Shellcode	6
Worms	7
Exploits	8
Backdoor	9

Upon calculating the situation value, it undergoes a normalization process to map the values to a consistent range, enabling easier analysis and modeling. Normalization ensures that the data is scaled appropriately, and facilitates comparison between different instances. Equation (14) represents the normalization formula applied to the situation value.

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{14}$$

The normalized situation value set is partitioned into a training set and a test set, adhering to a 9:1 ratio. The training set encompasses 1260 time periods, while the test set comprises 140 time periods. Furthermore, a sliding window technique is employed to process the situation value set, wherein the preceding 10 situation values are utilized as input sequences for predicting the subsequent 11th situation value. The dataset structure is succinctly depicted in Table 2.

Table 2. Sliding window settings.

Input Sequences	Output Results
$x_1, x_2, ..., x_{10}$	x_{11}
$x_2, x_3, ..., x_{11}$	x_{12}
...	...
$x_{n+1}, x_{n+2}, ..., x_{n+10}$	x_{n+11}

3.2. EXPERIMENTAL ENVIRONMENT SETUP

The improved LSTM-Informer network model in this study was implemented using the PyTorch framework. The experimental environment was configured as follows: the operating system was

Windows 10, the CPU was an Intel(R) Core(TM) i5-7300HQ @ 2.50GHz, the internal memory was 8GB, and the hard disk was a 1TB mechanical hard disk. The development framework used was PyTorch 1.12.0, and the development language was Python 3.9.16.

3.3. PARAMETER SETTINGS

The PSO algorithm was utilized to optimize the experiment's parameters and im-prove prediction results. The optimization process followed these steps: configuring PSO algorithm parameters, defining the search range for model parameters, generating particles in the population, setting the fitness function, initializing fitness values, updating velocities and positions, and computing the best parameter combination. The optimized model parameters are listed in Table 3.

Table 3. Optimized model parameter configuration.

Parameter Names	Optimization Values
d_model	60
l_layers	2
e_layers	2
d_layers	1
seq_len	8
factor	7
n_heads	10

Additionally, the learning rate was set to 0.001, the batch size was set to 32, and the model underwent training for 100 epochs.

3.4. EVALUATION CRITERIA

When evaluating network security situation prediction models, various metrics are used to assess their predictive accuracy and precision. In this study, we employed Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-Square (R^2) as evaluation criteria [17]. The formulas for these metrics are given by Equation (15).

$$\begin{cases} MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \\ MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \\ RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \\ R^2 = \frac{\left[\sum_{i=1}^{N}(y_i - \overline{y})\left(y_i - \overline{\hat{y}}_i\right)\right]^2}{\left[\sum_{i=1}^{N}(y_i - \overline{y})^2\right]\left[\sum_{i=1}^{N}\left(\hat{y}_i - \overline{\hat{y}}_i\right)^2\right]} \end{cases} \quad (05)$$

In these formulas, y_i represents the true situation value, $\hat{y}_i$ represents the predicted situation value, N represents the sample size, $\overline{y}_i$ represents the mean of the true situation values, and $\overline{\hat{y}}_i$ represents the mean of the predicted situation values.

3.5. COMPARISON EXPERIMENT AND ANALYSIS

3.5.1. Comparison of Regular LSTM-Informer and Improved LSTM-Informer for Image Prediction

To validate the advantages of the LSTM-Informer prediction model developed in this paper, which combines LGB and EMD, it is compared with the conventional LSTM-Informer model for image prediction, as depicted in Fig. 3 to 5.

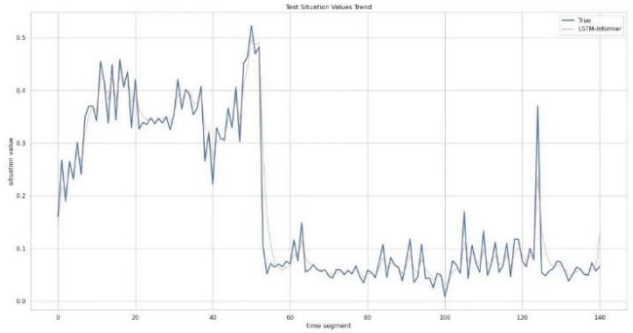

Figure 3. Regular LSTM-Informer Model Prediction.

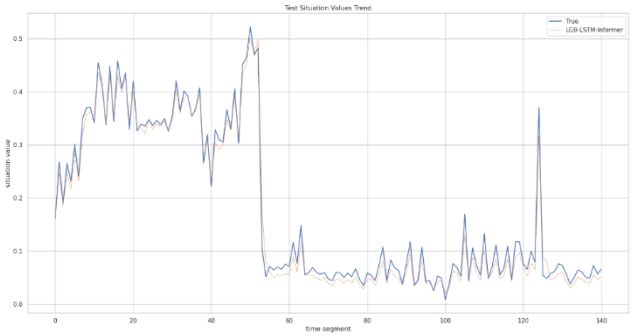

Figure 4. LGB-LSTM-Informer Model Prediction

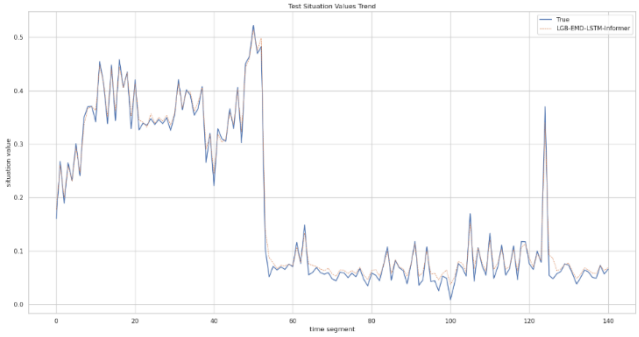

Figure 5. LGB-EMD-LSTM-Informer Model Prediction

As shown in these figures above, after comparing the ordinary LSTM-Informer model and the LGB-LSTM-Informer model, the improved LSTM-Informer model in this study demonstrates significantly superior prediction results. The experimental results fully demonstrate that the fusion of LGB and EMD significantly enhances the robustness of the model. Specifically, LGB significantly improves the model's ability to identify and adapt to outliers and complex patterns in the data through its efficient feature processing capability, thus enhancing the stability and accuracy of prediction. Meanwhile, EMD, through its excellent data decomposition capability, effectively removes the noise

components in the input data, ensuring the purity and quality of the data, which in turn provides the model with more accurate basic data. The combination of these two techniques not only makes the model more robust and significantly improves its ability to cope with data fluctuations, but more importantly, they work together to significantly improve the model's prediction accuracy, especially when dealing with complex and nonlinear time series data. Therefore, by integrating LGB and EMD techniques, the LSTM-Informer model significantly outperforms traditional methods in terms of prediction accuracy and robustness, setting a new performance standard for time series analysis.

3.5.2. Convergence Comparison between Regular LSTM-Informer and Improved LSTM-Informer

Similarly, to further validate the effectiveness of the improved LSTM-Informer model proposed in this paper, a comparison of its convergence with the ordinary LSTM-Informer model is shown in Fig. 6. By observing the descending curve of the loss function, we can evaluate the convergence of the model during the training process.

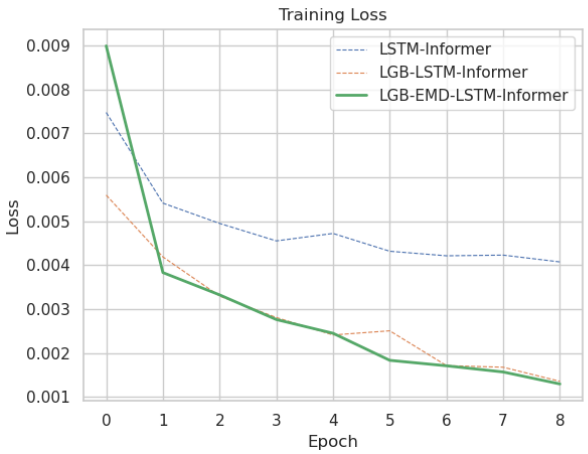

Figure 6. Comparison of convergence of the models

As shown in the above figure, the improved LSTM-Informer model performs better in terms of convergence speed and stability than the normal LSTM-Informer model and the LGB-LSTM-Informer model. Experiments show that the improved LSTM-Informer model proposed in this paper has better convergence.

3.5.3. Comparison of images predicted by different models

To verify the effectiveness of the improved LSTM-Informer model proposed in this paper, it is compared with LSTM, GRU, BiGRU and CNN models for predicting images as shown in Fig. 7.

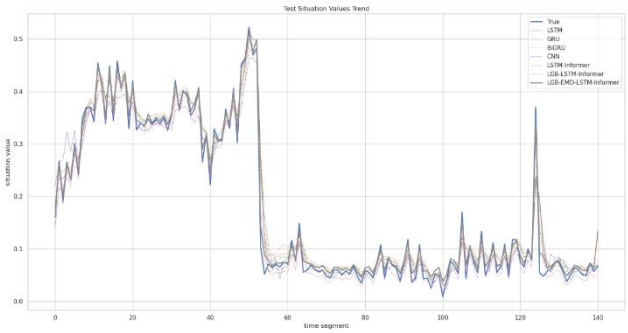

Figure 7. Comparison of images predicted by different models.

The experimental results demonstrate a notable improvement in the cyber security posture prediction accuracy achieved by the LSTM-Informer model. This model, incorporating both LGB and EMD techniques, surpasses the prediction accuracy of several other prominent models such as LSTM, GRU, BiGRU, and CNN. This enhancement underscores the efficacy of integrating LGB and EMD

methodologies into the LSTM-Informer framework, offering a more robust and reliable approach for cyber security posture prediction tasks.

3.5.4. Comparison of evaluation indicators for different models

In order to compare the performance of each model, this paper uses several evaluation metrics, including MAE, MSE and RMSE. The comparison results of the evaluation metrics for each model are shown in Table 4 and Fig. 8.

Table 4. Comparison of evaluation indexes of each model.

	MSE	RMSE	MAE
LSTM	0.0011	0.0325	0.0225
GRU	0.0010	0.0316	0.0210
BiGRU	0.0010	0.0317	0.0218
CNN	0.0012	0.0353	0.0241
LSTM-Informer	0.0008	0.0286	0.0185
LGB-LSTM-Informer	0.0003	0.0166	0.0134
LGB-EMD-LSTM-Informer	0.0001	0.0119	0.0090

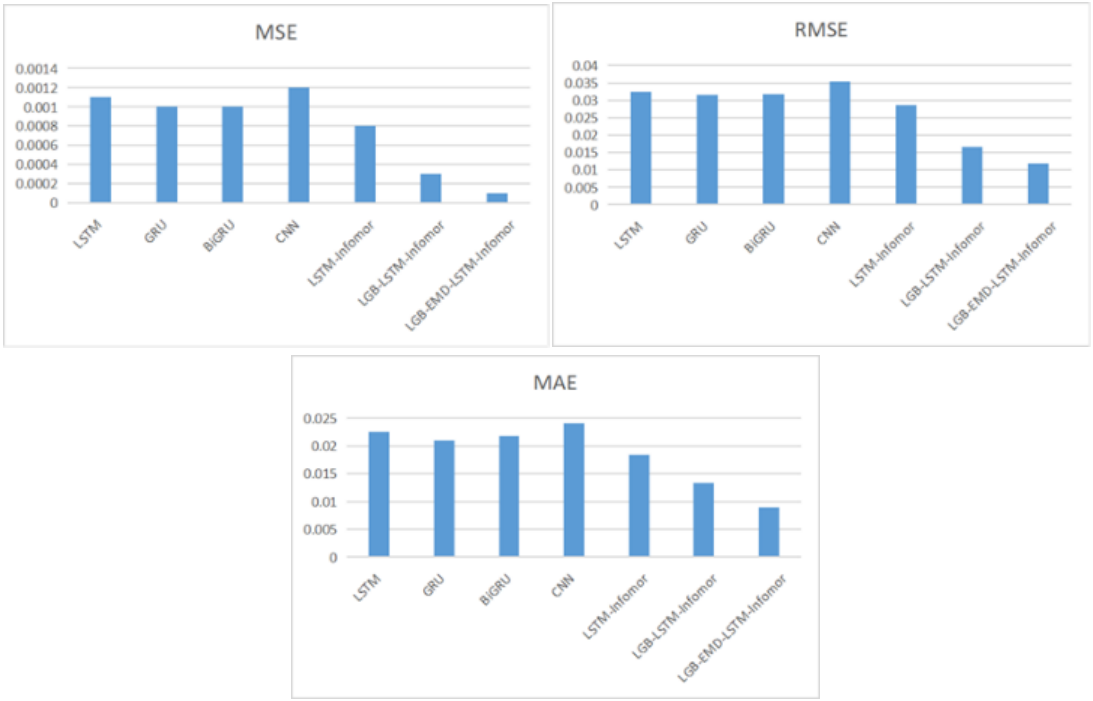

Figure 8. Comparison of prediction error of each model.

From Table 4 and Fig. 8, it can be seen that the LSTM-Informer model combining LGB and EMD exhibits superior performance in terms of error values compared to other models. It achieves significant reduction in MAE, MSE and RMSE compared to LSTM, GRU, BiGRU and CNN models. The experimental results demonstrate the excellent predictive accuracy of the improved LSTM-Informer model in predicting the network security situation values.

3.5.5. Comparison of different model fits

The effectiveness of the improved LSTM-Informer model proposed in this paper is verified by observing the differences in the fit of the different models. The comparison of the fit of each model is shown in Table 5 and Fig. 9.

Table 5. Comparison of fitting degree of each model.

	R2
LSTM	0.9519
GRU	0.9547
BiGRU	0.9544
CNN	0.9433
LSTM-Informer	0.9629
LGB-LSTM-Informer	0.9875
LGB-EMD-LSTM-Informer	0.9936

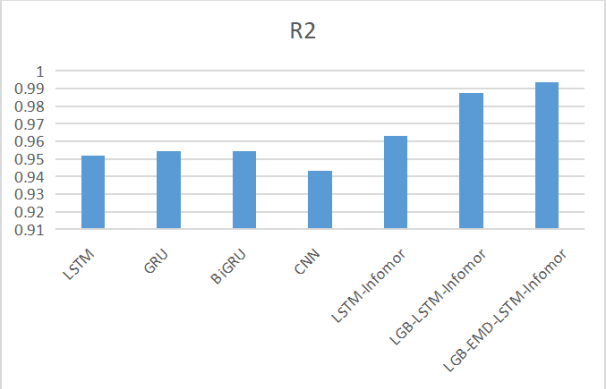

Figure 9. Comparison of the fit of the models.

As can be seen from Table 5 and Fig. 9, the improved LSTM-Informer model has a better fit than the other models, its R^2 is relatively high, and there is still a certain degree of room for improvement in this model. The experimental results show that the improved LSTM-Informer model proposed in this paper has certain advantages.

4. VALIDATION AND APPLICATION

The LSTM-Informer model was evaluated on real-world situational datasets obtained from the CNVD [18]. This dataset covers a period of 376 weeks, from the 10th issue of 2016 to the 21st issue of 2023. Utilizing the CNVD data provides extensive coverage and authenticity, as it is based on actual network security incidents, ensuring reliability and representativeness. By conducting experiments in this real-world environment, we can accurately assess the predictive capabilities and effectiveness of the LSTM-Informer model, as shown in Fig. 10.

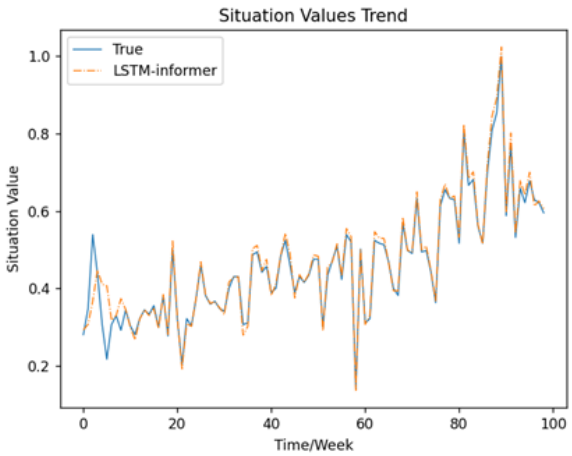

Figure 10. Forecasts of real environmental posture values.

The experimental findings suggest that the LSTM-Informer model displays robust fitting capabilities when applied to real situational data. This performance underscores its potential and reliability for practical applications in various domains. The model's ability to effectively capture and analyze real-world situational data highlights its adaptability and suitability for addressing complex scenarios in real-time environments.

5. CONCLUSION

In this paper, an improved LSTM-Informer cybersecurity posture prediction model is proposed. By preprocessing the cybersecurity posture data through EMD technology, the stable IMF is extracted as the input to the LSTM model, which significantly improves the accuracy and stability when dealing with time-series data, and enhances the performance of the whole LSTM-Informer model in cybersecurity posture prediction. Second, an LGB-based encoding and decoding mechanism is introduced. This mechanism is able to efficiently encode the features of the time-series data of cybersecurity posture, thus improving the information flow between the LSTM layer and the Informer layer. This not only enhances the model's capability in processing and parsing the data, but also demonstrates a significant performance improvement in predicting future cybersecurity postures. Meanwhile, a parameter tuning method for the PSO optimal algorithm is proposed to enhance the model stability and the ability of generating. We conducted experimental comparisons using the UNSW-NB15 dataset, and also evaluated real-world data from the China National Vulnerability Database. The experimental results show that the proposed improved LSTM-Informer model has good performance in fitting real cybersecurity scenarios, affirming its potential and reliability in practical applications. This approach, which combines deep learning and traditional machine learning techniques, not only improves the prediction accuracy, but also proves the great potential of the combination of these techniques to be applied in the field of cybersecurity.

In future research, efforts should be made to refine and systematically improve the data acquisition and processing methods, while focusing on the tuning of model parameters and the improvement of algorithm performance. Such a comprehensive approach not only helps to validate the robustness and the ability to panoply of the model in a variety of real-world application scenarios, but also ensures the comprehensiveness and scientific validity of the research methodology. Through this multifaceted improvement and validation, the effectiveness of the model can be more deeply understood and enhanced, laying a solid foundation for future applications and research.

REFERENCES

[1] G. Ke, R.-S. Chen, Y.-C. Chen, and J.-h. Yeh, "Network security situation prediction method based on support vector machine optimized by artificial Bee colony algorithms," Journal of Computers, vol. 32, no. 1, pp. 144-153, 2021.

[2] G. Wang, "Comparative study on different neural networks for network security situation prediction," Security and Privacy, vol. 4, no. 1, p. e138, 2021.

[3] L. Yuan, "Prediction of network security situation awareness based on an improved model combined with neural network," Security and Privacy, vol. 4, no. 6, p. e181, 2021.

[4] Y. Tang, C. Li, and Y. Song, "Network security situation prediction based on improved particle swarm optimization and extreme learning machine," Journal of Computer Applications, vol. 41, no. 3, p. 768, 2021.

[5] L. Chen, G. Fan, K. Guo, and J. Zhao, "Security situation prediction of network based on lstm neural network," in IFIP international conference on network and parallel computing, 2020: Springer, pp. 140-144.

[6] S. Li, D. Zhao, and Q. Li, "A framework for predicting network security situation based on the improved LSTM," EAI Endorsed Transactions on Collaborative Computing, vol. 4, no. 13, 2020.

[7] J. Lin and M. Wei, "Network security situation prediction based on combining 3D-CNNs and Bi-GRUs," International Journal of Performability Engineering, vol. 16, no. 12, p. 1875, 2020.

[8] C. He and J. Zhu, "Security situation prediction method of GRU neural network based on attention mechanism," Systems Engineering and Electronic Technology, vol. 43, no. 1, pp. 258-266, 2021.

[9] Z. Dongmei and L. Zhijian, "Network security situation prediction based on Transformer," J. Journal of Huazhong University of Science and Technology (Natural Science Edition), vol. 50, no. 05, pp. 46-52, 2022.

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol. 9, no. 8, pp. 1735-1780, 1997.

[11] H. Zhou et al., "Informer: Beyond efficient transformer for long sequence time-series forecasting," in Proceedings of the AAAI conference on artificial intelligence, 2021, vol. 35, no. 12, pp. 11106-11115.

[12] N. E. Huang et al., "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences, vol. 454, no. 1971, pp. 903-995, 1998.

[13] G. Ke et al., "Lightgbm: A highly efficient gradient boosting decision tree," Advances in neural information processing systems, vol. 30, 2017.

[14] R. Eberhart and J. Kennedy, "Particle swarm optimization," in Proceedings of the IEEE international conference on neural networks, 1995, vol. 4: Citeseer, pp. 1942-1948.

[15] S. Choudhary and N. Kesswani, "Analysis of KDD-Cup'99, NSL-KDD and UNSW-NB15 datasets using deep learning in IoT," Procedia Computer Science, vol. 167, pp. 1561-1573, 2020.

[16] Luo Z, "Research on neural network based cyber security posture assessment and prediction techniques," Master, 2018. [Online]. Available: https://kns.cnki.net/kcms2/article/abstract?v=w1Je9LIFm5BHIu9LTLzVC_YbmKGVCMPz4-UGKUgJrCGt7xXcWj5yFFr0WiKMf6wkFbVAjfZAfxj2jlRDsSWI2C9q_c6SQDtuhr375r-e_oV-0QkhbzqzqituvGbKg5P64yZkmJNcS7lA5UWgXcUDBQ==&uniplatform=NZKPT&language=CHS

[17] Z. Guo, J. Zhou, D. Wang, Z. Lv, and W. Yang, "Network intrusion detection method based on transformer neural network model," J Chongqing Univ, vol. 44, no. 11, pp. 81-88, 2021.

[18] CNVD. "Weekly report on network security information from 2016 to 2023 [Weekly report/Online]." https://www.cnvd.org.cn/webinfo/show/8876 (accessed.

[19] Z. Chen, "Research on the Application of Intelligent Learning Algorithms in Network Security Situation Awareness and Prediction Methods," in 2021 5th Asian Conference on Artificial Intelligence Technology (ACAIT), 2021: IEEE, pp. 309-311.

[20] J. Hu, D. Ma, C. Liu, Z. Shi, H. Yan, and C. Hu, "Network security situation prediction based on MR-SVM," IEEE Access, vol. 7, pp. 130937-130945, 2019.

[21] Z. Hu, S. Chen, and H. Chen, "Convolutional Neural Network Based Power Information Network Security Situational Awareness Model," in 2022 4th International Academic Exchange Conference on Science and Technology Innovation (IAECST), 2022: IEEE, pp. 243-247.

[22] T. Pooja and P. Shrinivasacharya, "Evaluating neural networks using Bi-Directional LSTM for network IDS (intrusion detection systems) in cyber security," Global Transitions Proceedings, vol. 2, no. 2, pp. 448-454, 2021.

[23] A. Sahu et al., "Multi-source multi-domain data fusion for cyberattack detection in power systems," IEEE Access, vol. 9, pp. 119118-119138, 2021.

[24] Y. Sun, L. Hou, Z. Lv, and D. Peng, "Informer-Based Intrusion Detection Method for Network Attack of Integrated Energy System," IEEE Journal of Radio Frequency Identification, vol. 6, pp. 748-752, 2022.

[25] Z. Xiong, Y. Li, J. Chen, and D. Chen, "FusedCNN-LSTM-AttNet: A Neural Network Model for Cyber Security Situation Prediction," in Proceedings of the 2023 International Conference on Communication Network and Machine Learning, 2023, pp. 207-210.

[26] C. Yao, Y. Yang, J. Yang, and K. Yin, "A Network Security Situation Prediction Method through the Use of Improved TCN and BiDLSTM," Mathematical Problems in Engineering, vol. 2022, 2022.

[27] K. Yin, Y. Yang, C. Yao, and J. Yang, "Long-Term Prediction of Network Security Situation Through the Use of the Transformer-Based Model," IEEE Access, vol. 10, pp. 56145-56157, 2022.

[28] H. Zhang, C. Kang, and Y. Xiao, "Research on network security situation awareness based on the LSTM-DT model," Sensors, vol. 21, no. 14, p. 4788, 2021.

[29] S. Zhang, Q. Fu, and D. An, "Network Security Situation Prediction Model Based on VMD Decomposition and DWOA Optimized BiGRU-ATTN Neural Network," IEEE Access, vol. 11, pp. 129507-129535, 2023.

[30] Z. Zhang et al., "Artificial intelligence in cyber security: research advances, challenges, and opportunities," Artificial Intelligence Review, pp. 1-25, 2022.