International Journal of Multiphysics
Volume 18, No. 3, 2024
ISSN: 1750-9548

Detection Method for Abnormal Flow Data in Power Business Based on Deep Learning

Jinkai Sun*, Guangxu Jiao, Junwei Zhang, Junfeng Yao, Chun Xiao
Marketing Service Center of State Grid Shanxi Electric Power Company, Taiyuan, 030012, China
*Corresponding Author.

Abstract

As the scale and complexity of China's power information network continue to expand, various application services based on this network are becoming increasingly widespread, leading to a gradual increase in the volume of generated data. Consequently, the detection of anomalous traffic data is crucial for ensuring the stability and security of power operations. Traditional anomaly detection methods primarily rely on shallow machine learning algorithms, which have significant limitations in feature extraction and pattern recognition, particularly when dealing with high-dimensional, nonlinear, and complex temporal data. This paper introduces a deep learning approach for anomaly identification in traffic data, combining Bidirectional Long Short-Term Memory (BiLSTM) networks using the Whale Optimization Algorithm (WOA) to achieve superior detection precision and robustness. Experimental findings reveal that the proposed method markedly enhances both accuracy and efficiency in comparison to conventional techniques, offering a more effective solution for anomaly detection in power business operations.

Keywords: Power system, anomaly detection in network traffic, deep learning.

1. Introduction

Anomaly detection in network traffic data for power operations presents unique challenges and is critically important. First, the traffic data in power systems is characterized by high dimensionality, nonlinearity, and complex temporal sequences, making detection tasks particularly challenging. Additionally, the security and stability of power systems are essential for the normal functioning of socio-economic activities[1]; any potential network threats or faults can lead to severe consequences. Therefore, promptly identifying and addressing anomalous traffic is crucial for ensuring the continuity and reliability of power systems[2,3]. Furthermore, with the continuous expansion of power information networks, the volume of data is growing exponentially, rendering traditional shallow machine learning methods inadequate for these challenges. As a result, researching and applying advanced deep learning methods to improve the accuracy and efficiency of anomaly detection has become a key direction in managing power system traffic data[4].

Traditional machine learning models such as LightGBM[5], active entropy[6], and K-means[7] typically address temporal issues and perform anomaly detection through feature engineering and supervised learning. These methods have achieved some success in detecting anomalous traffic. For instance, Wang et al. provided an anomaly traffic detection model using LightGBM to address the high miss and false detection rates in traditional attack detection[8]. This model employs KPCA for dimensionality reduction and then uses the LightGBM model for detection. Verification results indicate that this method performs well in dynamically detecting anomalous traffic in industrial control systems, but its generalization capability is insufficient for detecting unknown anomalous attack traffic. Additionally, a method based on active entropy for detecting network anomaly traffic has been proposed, utilizing the power integrated data network system as the research data source. Active entropy serves as the evaluation standard for traffic anomaly analysis. However, this method generally has a high false positive rate and lacks adaptability, requiring cautious application[9]. K-means is a classical clustering algorithm

in the field of intrusion detection, which iteratively updates centroids and partitions data until optimal clustering is achieved. Niu et al. designed a feature selection method based on information entropy for the IEC61850 intelligent substation protocol, utilizing the K-means clustering algorithm for anomaly traffic detection and analysis[10]. Despite its simplicity, K-means has slow convergence speed, high time complexity, and sensitivity to noise and outliers.

Deep learning algorithms can thoroughly explore and extract latent features from data, enabling the automatic extraction of high-level features without the need for manual feature engineering, thus reducing human bias. These models are particularly suited for processing high-dimensional, nonlinear, and complex time-series data, and have shown good performance and accuracy in traffic anomaly detection and power load prediction[11]. For instance, Vaswani et al. established a convolutional neural network (CNN) model for network intrusion detection, which improves classification accuracy by extracting local feature correlations through convolution operations[12]. Fei et al. proposed a traffic anomaly detection model based on Transformer, effectively addressing the issues of remote dependency and data sample imbalance in network traffic[13]. Although this model demonstrates excellent accuracy and detection time, its complexity results in high time complexity and long training times.

Overall, traditional machine learning methods, being shallow learning algorithms, have limitations in capturing critical information and fully extracting data features and correlations. Their generalization capabilities are weak, and their application range is relatively narrow. Moreover, the complex and dynamic network environment leads to network traffic data becoming increasingly high-dimensional. When faced with high-dimensional feature data, traditional deep learning models struggle to process the data effectively, resulting in reduced model efficiency. Therefore, eliminating redundant features and performing feature selection are crucial strategies to improve model efficiency. This paper provided a novel method which integrates Long Short-Term Memory (LSTM) networks and the WOA. The method employs BiLSTM for forward and backward feature extraction and utilizes a temporal attention mechanism to cope with long-term dependencies and dynamic patterns in time series data. Additionally, WOA is leveraged to optimize the model parameters, further enhancing the detection of anomalous traffic. Experimental results show that the proposed model surpasses traditional methods and other deep learning models on test datasets confirming its practical value and potential in detecting anomalous traffic data in power business operations.

2. Methods

2.1 LSTM

LSTM networks are a distinct variant of Recurrent Neural Networks (RNNs) engineered to handle and forecast long-term dependencies within time series data. Traditional RNNs struggle with the issues of gradient vanishing and gradient explosion when handling long sequences, rendering them ineffective for capturing long-term dependencies. LSTM addresses these challenges by incorporating unique structural elements, which enable the network to maintain and update information over extended periods[14].

An LSTM network comprises a sequence of interconnected LSTM cells, each featuring three crucial gates: an input gate, a forget gate, and an output gate. The input gate identifies the information that requires updating, utilizing a tanh activation function to generate a new candidate value. The forget gate, in particular, takes in the output from the preceding time step and the input from the current time step, producing a value between 0 and 1 via a sigmoid activation function. This value dictates the extent of information retained in the memory cells from the previous time step, where a value nearing 1 signifies substantial retention, and a value nearing 0 indicates significant forgetting. The output of the sigmoid function is combined with the output of the tanh function and added to the forget gate's output to refresh the memory cell. Similarly, the output gate employs a sigmoid activation function to determine which portions of the output are utilized, and a tanh activation function to modulate the output. These gates collectively regulate the flow of information, thereby preserving long-term and short-term memory. The detailed architecture of the LSTM is illustrated in Figure 1.

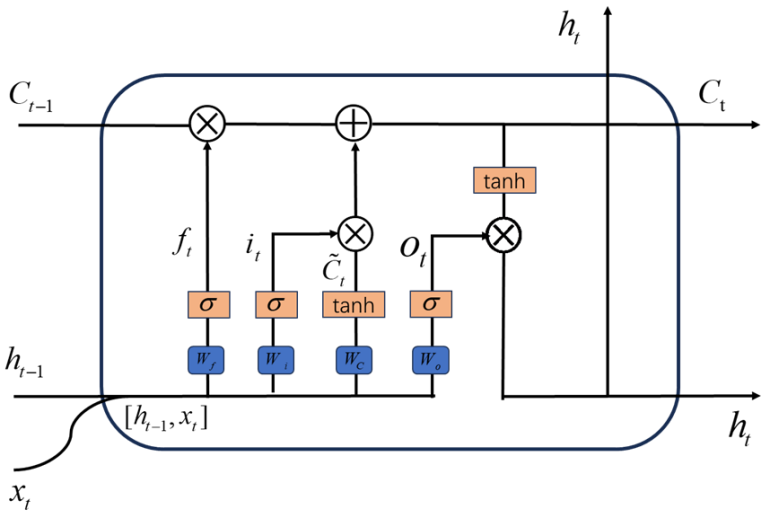

Figure 1 LSTM specific structure diagram

The specific formula for LSTM is as follows

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

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2}$$

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

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

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

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

In the aforementioned series of equations, C_{t-1} represents the model's knowledge state $t-1$, and is the newly acquired knowledge after inputting new observations, which are multiplied by the corresponding weighting parameter and then summed up to get the knowledge state at moment t. The weighting parameter determines the knowledge retained at these moments. This weighting parameter determines the knowledge retained at these moments. Among them, $[h_{t-1}, x_t]$ is the new vector formed by splicing the inputs of the previous moment and the current moment; W_f is the weight of the forget gate, b_f is the bias value of the forget gate; W_C is the weight of the memory unit, b_C is the bias value of the memory unit; W_i is the weight of the input gate, b_i is the bias value of the input gate; W_o is the weight of the output gate, b_o is the bias value of the output gate.

LSTM effectively addresses the gradient vanishing problem encountered by traditional Recurrent Neural Networks (RNNs) when processing long time series data through the introduction of cell states. By controlling the input, forget, and output gates, LSTM selectively retains and discards information, enhancing its capability to process and predict time series data. Due to its robust modeling ability and flexibility, LSTM finds applications in speech recognition, text generation, time series prediction, and anomaly detection.

2.2 BILSTM

BiLSTM network represents an advanced iteration of the Long Short-Term Memory network, tailored to manage and encapsulate bidirectional dependencies within time series data. By leveraging LSTM networks in both

forward and reverse directions concurrently, BiLSTM significantly improves the understanding and modeling capabilities for time series data[15].

The core structure of BiLSTM consists of two parallel LSTM networks: one processes information propagating from past to future (forward), while the other processes information from future to past (backward). In the same layer, the forward LSTM network receives the original sequential data, processing the input sequence's forward information. Conversely, the backward LSTM network receives the reversed sequential data, processing the input sequence's backward information. This bidirectional processing allows BiLSTM to capture both past and future information within the time series.

In particular, within a BiLSTM framework, the output at any given time step is influenced not only by all preceding inputs but also by all subsequent inputs. Each time step in a BiLSTM yields two hidden states: one derived from the forward LSTM network and the other from the backward LSTM network. These hidden states are subsequently integrated to produce the final output for that time step. This combination is typically achieved through simple concatenation or averaging. Since the forward and backward LSTM networks are trained simultaneously, BiLSTM can leverage both past and future information at each time step, thus providing richer contextual information[16]. The overall structure of BiLSTM is illustrated in Figure 2, where the operations of the forward and backward LSTM networks are depicted in parallel.

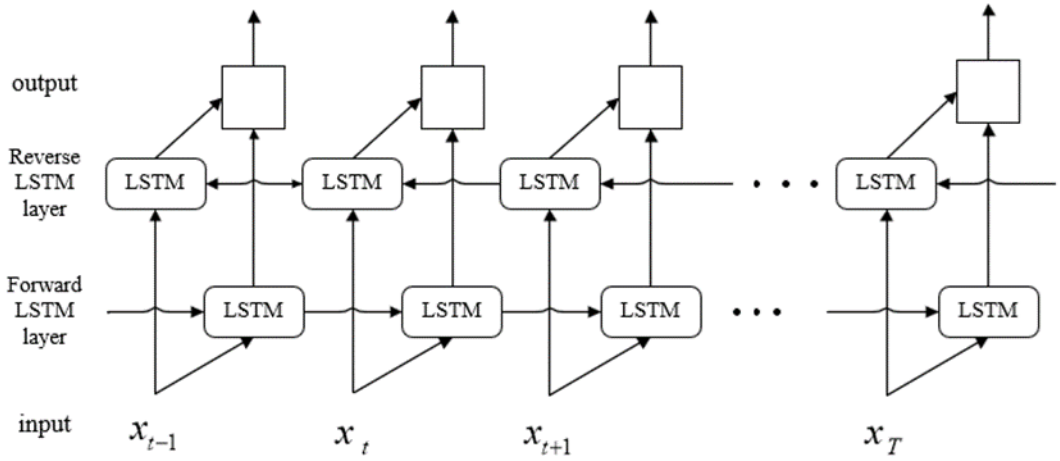

Figure 2 Structure diagram of BILSTM

Table 1 The forward and backward LSTM calculations in BILSTM.

Forward LSTM	Backward LSTM
$C_t = f_t \circ C_{t-1} + i_t \circ \tilde{C}_t$	$C_t = f_t \circ C_{t+1} + i_t \circ \tilde{C}_t$
$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$	$f_t = \sigma(W_f \cdot [h_{t+1}, x_t] + b_f)$
$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$	$i_t = \sigma(W_i \cdot [h_{t+1}, x_t] + b_i)$
$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$	$\tilde{C}_t = \tanh(W_C \cdot [h_{t+1}, x_t] + b_C)$
$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$	$o_t = \sigma(W_o \cdot [h_{t+1}, x_t] + b_o)$
$h_t = o_t \circ \tanh(C_t)$	$h_t = o_t \circ \tanh(C_t)$

Table 1 provides a detailed description of the forward and backward propagation processes in BiLSTM while handling input sequences. By simultaneously processing both forward and backward information, BiLSTM captures global dependencies within time series data. This dual processing enhances the understanding of contextual information, making the output at each time step dependent on both past and future information. The unique bidirectional structure of BiLSTM addresses the limitations of unidirectional LSTM in capturing global information in time series, offering a more comprehensive and efficient modeling approach. This capability

endows BiLSTM with significant advantages and broad application prospects in handling complex time series tasks.

2.3 Attention mechanisms

The Attention Mechanism is an approach extensively employed in deep learning, especially within natural language processing (NLP) domains. Its primary purpose is to emulate the human attention mechanism by assigning different weights to various input information, thereby highlighting key information, suppressing irrelevant information, and enhancing the model's performance and efficiency. The working process of the attention mechanism can be divided into the following steps[17]:

1) Given a query vector, calculate its similarity with all key vectors. The similarity calculation typically employs dot product, cosine similarity, or other similarity measures.

2) Transform the calculated similarities into a weight distribution using a SoftMax function. These weights reflect the importance of each key-value pair relative to the query vector.

3) Perform a weighted sum of all value vectors to obtain the final attention vector. Specifically, the attention vector is formed by the weighted sum of the query vector and the value vectors. The overall structure of the attention mechanism is illustrated in Figure 3.

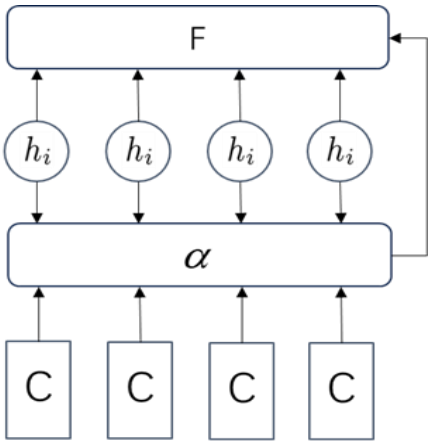

Figure 3 Schematic diagram of the attention mechanism

2.4 WOA algorithm

The Whale Optimization Algorithm (WOA), introduced by Mirjalili and Lewis in 2016, is an optimization algorithm based on swarm intelligence. The algorithm emulates the humpback whale's "bubble net feeding" behavior in nature, and solves complex optimization problems by simulating the humpback whale's behaviors of Encircling Prey, Spiral Updating Position, Search for Prey[18].

Whales are usually able to recognize the position of their prey and encircle them. This behavior is modeled using the following mathematical approach: as the precise position of each whale cannot be determined in advance, the position of the whale must be updated at this stage based on the current position of the prey in order to lock the encirclement.

$$D = | C \cdot X^*(t) - X(t) | \tag{7}$$

In Equation 7 $X^*(t)$ is the optimal individual position; $X(t)$ is the current individual position, t is the current number of iterations; D is the update step size when surrounded, C is the perturbation to the prey. The following equations show the update of each unique point and the definition of coefficient vectors A,C:

$$X(t+1) = X^*(t) - A \cdot D \tag{8}$$

$$A = 2\boldsymbol{a} \cdot \boldsymbol{r} - \boldsymbol{a} \tag{9}$$

$$C = 2r \tag{10}$$

In this context, *a*reduces linearly from 2 to 0, and *r*represents a random vector within the range [0,1]. The hunting behavior of whales can be summarized as spiral updating, encircling, and contracting, with the following updating formula for the model.

$$X(t+1) = D^{*} \cdot e^{bl} \cdot cos(2\pi l) + X^{*}(t) \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{11}$$

Where: D^{*} denotes the distance between the whale and its prey, which is defined as $D^{*} = | X^{*}(t) - X(t) |$, b is a constant defining the shape of the spiral and l is a random number between [-1, 1].

To increase the global search capability of the algorithm, whales search for prey randomly. When |A|>1, the whale explores the search space randomly for prey. The mathematical representation is shown as follows:

$$D = | C \cdot X_{rand} - X(t) | \tag{12}$$

$$X(t+1) = X_{rand} - A \cdot D \tag{13}$$

The WOA process is as follows:

1) Initialization: Generate the initial positions of individual whales randomly, calculate each whale's fitness value, and determine the initial optimal solution.

2) Iterative optimization: Adjust the coefficient vectors A and C, then decide to either encircle the prey, execute a bubble net attack, or randomly search for the prey, depending on the probability p. Update the position of each whale and recompute the fitness value. If a new optimal solution is found, replace the current optimal solution.

3) Termination condition: Termination criteria include reaching the maximum number of iterations or meeting the convergence condition, at which point the optimal solution is produced.

The WOA algorithm enhances global search capabilities by emulating the feeding behavior of humpback whales, particularly their bubble net feeding strategy, which helps the method to find globally optimal solutions rather than falling into local optima when solving complex, multi-peak optimization problems, and the algorithm also excels in convergence speed[19,20].

2.5 Detailed model

The illustrated model architecture in Figure 4 consists of three main components: the WOA section, and the BiLSTM-Attention section. In the BiLSTM-Attention section, the initial parameter decoding is influenced by the WOA algorithm, which includes factors like the number of nodes per hidden layer, learning rate, and iteration count. The network undergoes training with the specified training dataset. Subsequently, predictions from the test dataset are analyzed using the mean square error between actual and target outputs. This error metric is then utilized by the WOA algorithm as a fitness value. The WOA algorithm employs discoverer, follower, and vigilant strategies based on these fitness values to update both individual and collective optimal solutions within the population. This process results in a set of refined network hyperparameters, thereby improving the performance of the BiLSTM model. The operational flow of the WOA-BiLSTM-Attention model is depicted in Figure 4.

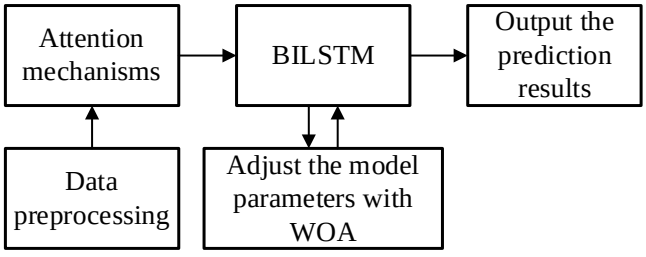

Figure 4 The specific flow chart of the model in this article

3. Results

3.1 Data pre-processing

The power business anomalous traffic data dataset collected in this experiment has a total of 42575 pieces of data after cleaning, among which there are 31634 pieces of normal data, of which denial of service attack (DoS) accounts for 76.25%, the largest proportion, followed by Unauthorised Remote Control Attack(URCA), Field Combination Attack(FCA) and False Data Injection Attack(FDIA), respectively. The data preprocessing steps for this experiment are as follows.

1) Remove specific feature columns. As this paper needs to train the model based on grouping characteristics, useless features such as source and target IPs, source and target ports need to be removed.

2) Integration of data and elimination of dirty data. As the whole dataset is scattered, it must be synthesised into one dataset if it is to be trained, in addition the dataset has dirty data such as Nan, Infiniti, first row feature names and duplicates that need to be eliminated[21].

3) The proportion of some attack traffic in the data file of the balanced data set is too small, resulting in an unbalanced proportion. Therefore, this paper chooses 4 kinds of traffic with more data as the experimental object, and uses the conditional generative adversarial network to resample the original unbalanced data to get a new dataset, and normalises the obtained feature data, and the percentage of different traffic before and after resampling is shown in Table 2.

4) Data set partitioning. The pre-processed dataset is split into training and test sets at a ratio of 7:3, with a fixed random seed applied to ensure consistency across all classification iterations.

Table 2 Comparison of attack ratios before and after data preprocessing.

Comparison items	Type of attack	Number of attacks	Percentage
Before data preprocessing	DOS	8342	76.25%
	URCA	1226	11.21%
	FCA	978	8.94%
	FDIA	395	3.61%
After data preprocessing	DOS	8342	48.14%
	URCA	3225	18.61%
	FCA	2987	17.24%
	FDIA	2774	16.01%

3.2 Parameter setting and related experiments

Parameter settings significantly impact the convergence efficiency of the training model and the experimental outcomes. During training, the loss function employed is the commonly used cross-entropy loss, in conjunction with the Adam optimizer. The Adam optimizer is preferred for its ability to accommodate sparse gradients and mitigate gradient oscillation issues. Although the Adam optimizer can autonomously adjust the learning rate, the initial learning rate must be experimentally determined. In this study, three learning rates 0.1, 0.01, and 0.001 were tested. The loss values for both the training and testing sets were recorded and plotted, as shown in Figure 5. The results indicate suboptimal fitting on the test set at learning rates of 0.1 and 0.001. Conversely, a learning rate of 0.01 yielded optimal performance on the training set, thus establishing 0.01 as the initial learning rate.

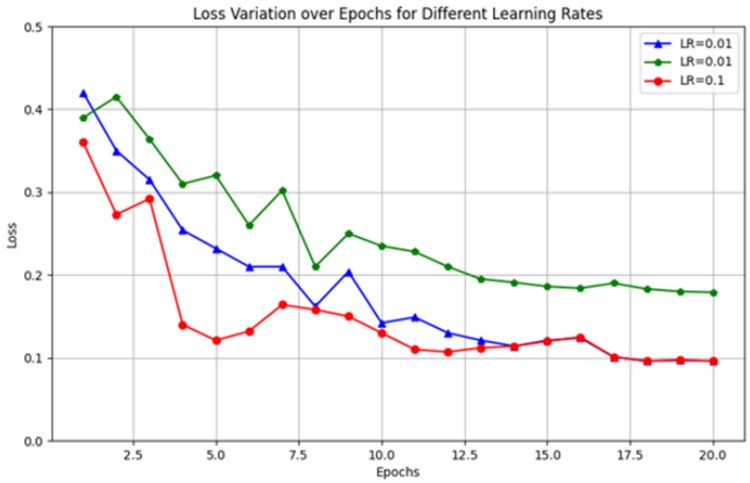

Figure 5 Experimental comparison of different learning rates

3.3 Analysis of results

In this paper, we use accuracy (ACC), recall (RE), precision (PR), and classifier precision score (F1-score, F1) to evaluate the model, calculated as follows.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{14}$$

$$PR = \frac{TP}{TP + FP} \tag{15}$$

$$RE = \frac{TP}{TP + FN} \tag{16}$$

$$F1 = 2 \times \frac{PR \times RE}{PR + RE} \tag{17}$$

Where, TP denotes detecting normal type as normal type, TN denotes detecting abnormal type as abnormal class, FP denotes detecting abnormal type as normal type and FN denotes as detecting normal type as abnormal type. The confusion matrix can be expressed as Table 3.

Table 3 Confusion matrix for attack detection metrics.

real type	Type of forecast	
	Normal	Attack
Normal	TP	FN
Attack	FP	TN

To provide a clearer assessment of the pre-processed data quality and model performance, Table 4 presents the model's classification accuracy on the experimental dataset,and it can be seen that the monitoring model of this experiment can effectively learn the data characteristics of various types of traffic, and the overall accuracy of the test (ACC) is 94.27% after the model is trained with the pre-processed dataset, and the precision rate (PR) and recall rate (RE) of the detection of normal data reach 99.26% and 99.43%, respectively. rate (PR) and recall rate (RE) for normal data reached 99.26% and 99.43%, respectively, indicating that the detection model is able to discriminate between normal and abnormal traffic better; therefore, it has a good performance in ACC, P, R and F1 values, and also indicates that the dataset which has been pre-processed by the experimental data is able to improve the classification accuracy of the detection model effectively.

Table 4 The model's performance on the dataset for classification tests.

Type of attack	ACC	PR	RE	F1
Normal	98.64%	99.26%	99.43%	99.56%
DOS		97.91%	97.22%	97.56%
URCA		97.14%	96.81%	96.96%
FCA		96.37%	96.89%	96.63%
FDIA		97.15%	97.12%	97.64%

Figure 6 illustrates the confusion matrix for the experimental results, further corroborating the aforementioned conclusions. he confusion matrix offers a detailed view of the model's predictive capability across various categories. The detection of normal data and denial-of-service attacks was the most accurate, whereas the detection of unauthorized remote control attacks, field combination attacks, and false data injection attacks was comparatively poor. This discrepancy is likely attributable to the smaller sample sizes of these attack types, leading to data imbalance during model training. Enhancing the dataset and refining the model architecture could improve detection efficacy for these attack types.

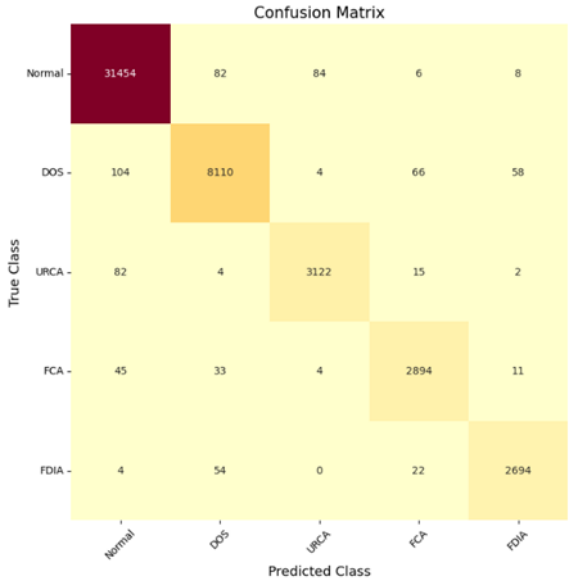

Figure 6 Confusion matrix of model classification detection results

In this experiment, the typical machine learning method SVM and deep learning methods LSTM and BILSTM-attention of the proposed model in this paper are selected for comparison, and the accuracy, precision, F1 score and recall of the model are analysed, Figure 7 demonstrates the comparison of this paper's model with the other models with regard to precision, accuracy, F1 score and recall with the pre-processed dataset, and it can be see that this paper's model DRSN-BiLSTM outperforms other models in all aspects.

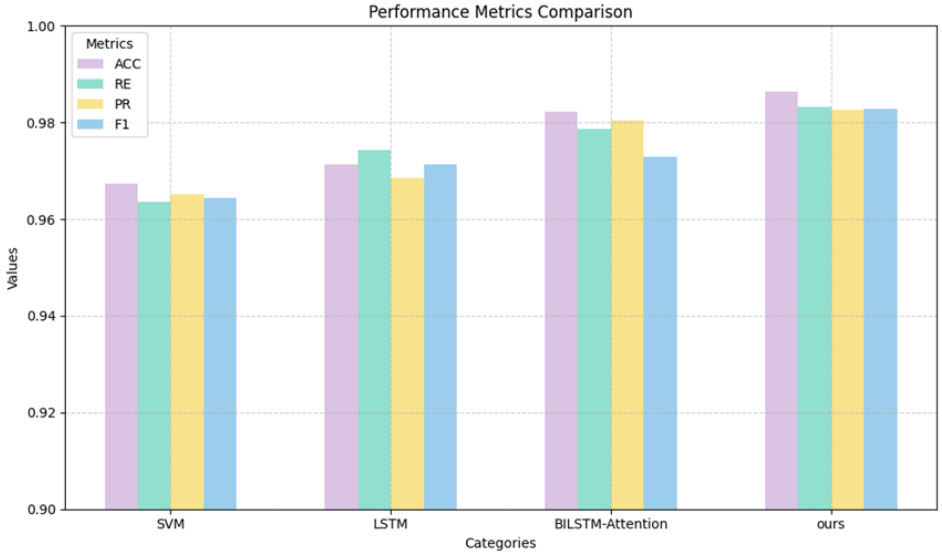

Figure 7 Comparison of accuracy, precision, F1 scores, and recall of different models

4. Discussion

In summary, this paper proposes a deep learning-based anomalous traffic data detection model for power business, which effectively improves the accuracy and robustness of anomaly detection by combining BiLSTM and Whale Optimisation Algorithm to optimise the model parameters, and introduces a temporal attention mechanism. Compared with traditional machine learning methods, the method in this paper is able to better capture potential features and associations in the data, handle high-dimensional nonlinear time-series data, and exhibit higher detection accuracy and generalisation ability. Experimental results show that the approach in this paper has better performance than multiple existing algorithms in all aspects such as accuracy, precision, F1 score, and recall, providing a more effective solution for network traffic anomaly detection in electric power business. This study not only provides strong technical support for the stability and security of the power system, but also provides new ideas and methods for the anomaly detection of other complex time-series data, further verifies the advantages of deep learning in processing complex time-series data, and provides a solid theoretical foundation and practical guidance for the development and application of the smart grid. Future research can further optimise the model structure and algorithm to improve its real-time and adaptability to handle the complex and changing power network situation.

Knowledgment

This paper is supported by the Science and Technology Project of the State Grid Shanxi Electric Power Company under grant, named Research and Application of Full Link Control and Tracking Technology for Sensitive Business Data Based on Fluorescent Labelling, No. 52051L230005.

References

[1] Krishna K, Murty M N. Genetic K-means algorithm. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 1999, 29(3): 433-439.

[2] Shi G, Shen X, Xiao F, et al. DANTD: A deep abnormal network traffic detection model for security of industrial internet of things using high-order features. IEEE Internet of Things Journal, 2023, 10(24): 21143-21153.

[3] Ageyev D, Radivilova T, Mulesa O, et al. Traffic monitoring and abnormality detection methods for decentralized distributed networks. Information security technologies in the decentralized distributed networks. Cham: Springer International Publishing, 2022: 287-305.

[4] Dong S, Xia Y, Peng T. Network abnormal traffic detection model based on semi-supervised deep reinforcement learning. IEEE Transactions on Network and Service Management, 2021, 18(4): 4197-4212.

[5] Lao Z, He D, Wei Z, et al. Intelligent fault diagnosis for rail transit switch machine based on adaptive feature selection and improved LightGBM. Engineering Failure Analysis, 2023, 148: 107219.

[6] DeAlmeida J M, Pontes C F T, DaSilva L A, et al. Abnormal behavior detection based on traffic pattern categorization in mobile networks. IEEE Transactions on Network and Service Management, 2021, 18(4): 4213-4224.
[7] Salman O, Elhajj I H, Chehab A, et al. A machine learning based framework for IoT device identification and abnormal traffic detection. Transactions on Emerging Telecommunications Technologies, 2022, 33(3): e3743.
[8] Salman O, Elhajj I H, Kayssi A, et al. A review on machine learning–based approaches for Internet traffic classification. Annals of Telecommunications, 2020, 75(11): 673-710.
[9] Zhao Yu, Huo Yonghua, Huang Wei, et al. Traffic anomaly detection method based on bidirectional LSTM model. Radio Engineering, 2023, 53(07): 1712-1718.
[10] Zhaoyang Niu, Guoqiang Zhong, Hui Yu, A review on the attention mechanism of deep learning, Neurocomputing, Volume 452, 2021,Pages 48-62, ISSN 0925-2312.
[11] Zhao J, Huang F, Lv J, et al. Do RNN and LSTM have long memory? International Conference on Machine Learning. PMLR, 2020: 11365-11375.
[12] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. Advances in neural information processing systems, 2017, 30.
[13] Fei J, Yao Q, Chen M, et al. The abnormal detection for network traffic of power iot based on device portrait. Scientific Programming, 2020, 2020(1): 8872482.
[14] Zhang J, Ye L, Lai Y. Stock price prediction using CNN-BiLSTM-Attention model. Mathematics, 2023, 11(9): 1985.
[15] Sangeetha J, Kumaran U. A hybrid optimization algorithm using BiLSTM structure for sentiment analysis. Measurement: Sensors, 2023, 25: 100619.
[16] Cahuantzi R, Chen X, Güttel S. A comparison of LSTM and GRU networks for learning symbolic sequences. Science and Information Conference. Cham: Springer Nature Switzerland, 2023: 771-785.
[17] Shiri F M, Perumal T, Mustapha N, et al. A comprehensive overview and comparative analysis on deep learning models: CNN, RNN, LSTM, GRU. arXiv preprint arXiv:2305.17473, 2023.
[18] Mirjalili, S.; Lewis, A. The Whale Optimization Algorithm. Adv. Eng. Softw. 2016, 95, 51–67.
[19] Javaheri D, Gorgin S, Lee J A, et al. Fuzzy logic-based DDoS attacks and network traffic anomaly detection methods: Classification, overview, and future perspectives. Information Sciences, 2023, 626: 315-338.
[20] Arjunan T. Real-Time Detection of Network Traffic Anomalies in Big Data Environments Using Deep Learning Models. International Journal for Research in Applied Science and Engineering Technology, 2024, 12(9): 10.22214.
[21] Wawrowski Ł, Białas A, Kajzer A, et al. Anomaly detection module for network traffic monitoring in public institutions. Sensors, 2023, 23(6): 2974.