www.cad-journal.net
Computer-Aided Design AND APPLICATIONS

Anomaly Detection of Power Time-Series Data Based on Multi-Dimensional Transformer Network

Xiongbo Xiao[1] , Zhonglin Yang[2] and Xueping Gao[3]

[1]National Key Laboratory of Electromagnetic Energy, 576285039@qq.com
[2]National Key Laboratory of Electromagnetic Energy, blueduny@sina.com
[3]National Key Laboratory of Electromagnetic Energy, gaoxp1202@163.com

Corresponding author: Xiongbo Xiao, 576285039@qq.com

Abstract. Power time-series anomaly detection has always been one of the important means to mine security threats in the power grid. The traditional detection method based on machine learning has certain limitations. It is difficult to capture the dependence between data and is prone to missed detection and false detection. In view of the above problems, this paper proposes a method for detecting abnormal data of power time-series based on Swin transformer. Firstly, the transformer model can predict the development trend of power data more effectively by using its ability to extract global information. Secondly, considering the deployment of the model in the actual scene, in order to improve the running speed and efficiency of the model, Swin transformer uses its unique patch to cut the data into small windows and reduce the sequence length. In addition, the design of the mobile window enhances the information exchange between the data and achieves the ability of global modeling. Finally, the input lightweight gradient boosting tree of the model is further improved in accuracy. The experimental results show that the proposed method can greatly reduce the calculation time under the premise of effectively identifying abnormal data.

Keywords: time-series; anomaly detection; transformer; lightweight
DOI: https://doi.org/10.14733/cadaps.2024.S7.15-27

1 INTRODUCTION

In recent years, with the continuous development of smart grid, power data analysis has become one of the important means to ensure the security of power grid data. The power data contains key information related to grid security, such as data flow, data integrity, and data security. Data tampering and data leakage will lead to economic and information loss of users [1]. Therefore, effectively identifying the outliers in the power grid data has become a necessary measure to maintain the security of the power grid and the economic interests of users.

With the lack of development of network applications, the data of the power information system is facing the threat of network hackers, viruses, Trojan horse programs, etc. Once the power

information system is attacked, the core data of the power grid operation will face the risk of leakage. The power system will also face the risk of paralysis, which will have a great impact on social stability and the personal life of power grid users [2]. Therefore, the abnormal detection of power grid data will have a crucial impact on the continuous and stable operation of the power grid. Timely and effective detection of problems will greatly improve the production efficiency of the power grid system in daily life, which is conducive to the continuous and efficient operation of the power market.

At present, with the rapid development of anomaly detection of power time-series data, there are many detection methods. The method based on statistical models: autoregressive conditional heteroscedasticity model [3] is not only relatively simple and flexible, but also considers the heteroscedasticity of time-series data. However, if the power time-series data has seasonality or trend, the autoregressive conditional heteroscedasticity model may not capture this information. The differential integrated moving average autoregressive model [4] is also a relatively simple model. The model can capture data trends and seasonal changes, and can model and predict long-term and short-term time patterns. However, the model has high data requirements and is less effective for some complex and variable data processing. Method based on unsupervised clustering: K-Means [5] is a simple and efficient clustering algorithm. It has relatively low computational complexity and is suitable for processing large-scale time-series data. At the same time, it does not require additional manual labeling. Generally, there are many previous studies on power time-series datasets analysis, but it is difficult to obtain stable and accurate results when faced with multiple noises and outliers. The density clustering algorithm [6] is sensitive to outlier detection and has strong tolerance to noise, but it has high computational complexity and poor computational efficiency, and it is difficult to deploy in actual scenarios. Spectral clustering algorithm [7] can effectively find small clusters, that is, clusters with fewer data points, which is very important for anomaly detection, because abnormal data usually form relatively good clusters or outliers. There are some key parameters in spectral clustering algorithm, such as similarity measure and spectral decomposition. Choosing appropriate parameters is very important for the performance and results of the algorithm. Parameters may determine the best configuration through experiments and tuning. Supervised classification method: Support vector machine (SVM) [8] shows strong classification ability when dealing with complex nonlinear data. By selecting appropriate kernel function, SVM can map power time-series data to high dimensional feature space for better separating normal data and abnormal samples. It can also be effective in a small number of samples, but it mainly focuses on the boundary and support vector in the feature space. For power time-series data with long-term dependence, the performance of SVM may be limited. Long short-term memory artificial neural network (LSTM) [9] can capture and process long-term dependencies in power time series data through memory cell and gating mechanism. This makes it advantageous in analyzing power time-series data with complex time patterns and interdependence, but LSTM networks usually have a large model size and complex calculation process, especially when dealing with large-scale power time-series data. This may lead to high computational complexity of training and inference, and the algorithm efficiency and scalability should be fully considered. The isolated forest [10] can construct a tree by selecting features and dividing data randomly, so that it can quickly identify the area where the abnormal sample is located without interfered by the data distribution. However, the isolated forest has a relatively weak modeling ability for time series dependencies. He regards the data as an independent sample point and cannot directly capture the time series features and time series dependencies in power time-series data. The hidden Markov algorithm (HMM) [11] describes the time evolution process of the data by defining the hidden state and the state transition probability, which makes it advantageous in capturing the long-term dependence, periodicity or trend characteristics in the time series data. However, HMM is computationally complex, especially when dealing with large-scale power time-series data, the computational complexity is high.

With the rapid rise of deep learning technologies, some of them are already applied to anomaly detection of time series data. Reference [12] introduced an anomaly detection method of power monitoring system based on residual fully connected neural network. The combination of semi-supervised learning mechanism and deep learning method can identify abnormal data behavior in power information system quickly and accurately. In order to realize the fusion of heterogeneous

data in [13], the deep restricted Boltzmann machine mapped the heterogeneous data with different formats to a unified embedded vector space, and the recurrent neural network established a portrait of the obtained embedded vector data for detecting the outliers in the data. Reference [14] proposed an automatic acquisition method for automatic acquisition of abnormal data in power system based on deep learning. Under the premise of short time consumption, it can deal with the number of abnormal electronic transmissions with large values, which is more in line with the requirements of practical applications. However, these CNN based methods can extract data information in depth, but cannot capture long-term data information, including seasonal changes, periodic fluctuations, etc.

2 METHODOLOGY

At present, with the rise of deep learning methods, many deep learning-based methods are applied to time series detection. In order to extract continuous and interdependent time series information, some networks use CNN-based extractors. Because the size of the receptive field is subject to the size of the convolution kernel, the feature extraction based on the CNN method cannot consider the continuous information of the entire sequence. In view of the above problems, this paper proposes a feature extractor based on the Swin Transformer method, with an improved Swin Transformer as the backbone network. Combined with the lightweight gradient boosting tree (lightGBM) to detect outliers, the network structure is shown in Figure 1.

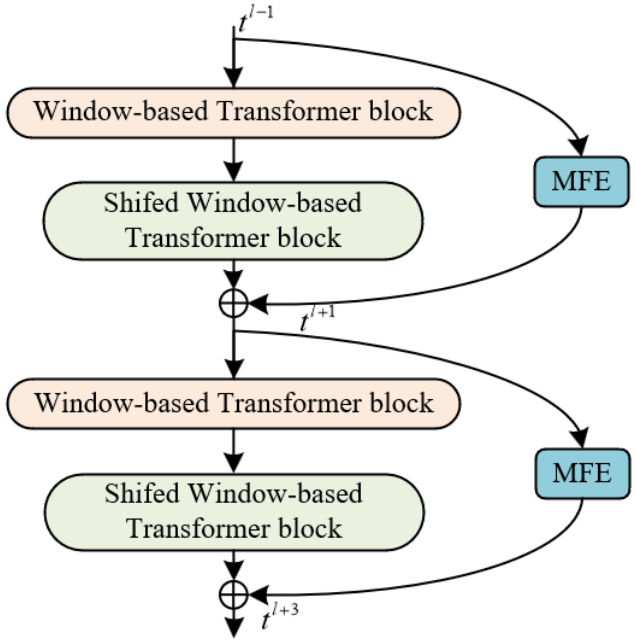

Figure 1: Network structural diagram.

2.1 Network Architecture

Timing information feature extraction has always been an important part of timing anomaly data detection. How to extract more perfect information is very important. In recent years, deep learning-based networks have been widely used as feature extractors in visual tasks such as images, and have good performance in classification, segmentation, and detection tasks. VGGNet [15] adopts a deep network structure, most of which are convolutional layers. This deep structure can make Vano learn more complex feature representations. The convolutional layer in VGGNet uses the

characteristics of parameter sharing, which reduces the number of network parameters and reduces the risk of overfitting. ResNet [16] introduced a residual connection, allowing information to be transmitted directly across layers in the network. This connection method adds the output of the previous layer to the input of the subsequent layer, so that the network can learn the residual mapping, which can effectively reduce the gradient disappearance problem in deep learning. In this paper, Swin transformer is applied as the main trunk, and an auxiliary multi-scale feature enhancement module is added to provide more information that is abundant for Swin transformer. The main trunk diagram of the Swin transformer is shown in Figure 2.

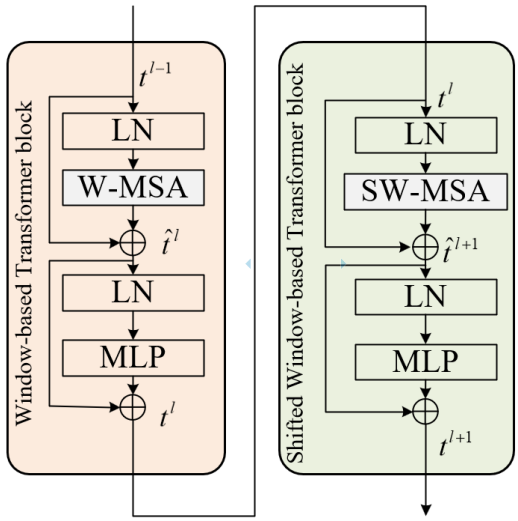

Figure 2: Swin transformer backbone diagram.

SW-MSA is a self-attention mechanism introduced in the Swin Transformer model. SWMSA works by dividing the input data into non-overlapping data blocks and performing self-attention operations on each data block. Different from the traditional self-attention mechanism, SW-MSA performs self-attention operations inside each data block rather than on the entire data. In addition, it also introduces a translation operation to establish connections between data blocks. This translation operation shifts the features of the data block and uses the shifted features for self-attention operation. This can help the model better capture the correlation between data blocks and improve the representation ability of features.

The expression of the Swin transformer trunk is as follows:

$$\hat{t}^{l} = WMSA\left(LN\left(\hat{t}^{l-1}\right)\right) + \hat{t}^{l-1} \tag{2.1}$$

$$t^{l} = MLP\left(LN(\hat{t}^{l})\right) + \hat{t}^{l} \tag{2.2}$$

$$\hat{t}^{l+1} = SWMSA\left(LN\left(\hat{t}^{l}\right)\right) + \hat{t}^{l} \tag{2.3}$$

$$t^{l+1} = MLP\left(LN\left(\hat{t}^{l+1}\right)\right) + \hat{t}^{l+1} \tag{2.4}$$

Among them, WMSA represents Window-based Transformer block, SWMSA represents Shifted Window based Transformer block, MLP represents multi-layer perceptron operation, and LN represents layer normalization.

The Swin Transformer module is included in each stage of the Swin transformer network. Each module has two structures, one is composed of a multi-headed self-attention (MSA) based window (WMSA) with a two-layer multi-layer perceptron (MLP), and the other is a sliding window (SWMSA). Because WMSA is applied first, and then SWMSA is used, the number of times the Swin Transformer module is stacked is even. A layer normalization unit (LN) is placed between each MSA and MLP to

accelerate the convergence speed of the model. Residual connections are used between LN and MSA and between MLP and LN to improve gradient dissipation.

In order to reduce the amount of calculation, the Swin Transformer network divides the data to be processed into non-overlapping windows, and performs self-attention calculation in different windows. Assuming that the height, width and dimension of an image are H, W and K respectively, then the calculation amount of the image is as follows:

$$CQ(MSA) = 4H \cdot W \cdot K^2 + 2(H \cdot K)^2 \cdot K \tag{2.5}$$

Assuming that the height and width of each window in the data are M, the data contains (H/M) × (W/M) windows. The calculation amount of each window is as follows:

$$CQ(Win) = 4(M \cdot K)^2 + 2(W)^2 \cdot K \tag{2.6}$$

Since there are (H/M) × (W/M) windows, the calculation of the WMSA is as follows:

$$CQ(WMSA) = 4H \cdot W \cdot K^2 + 2M^2 \cdot H \cdot W \cdot K \tag{2.7}$$

In the formulas above, CQ represents the amount of calculation. By comparing formula, it can be seen that the calculation amount of WMSA is lower than that of the whole image MSA. The computational complexity of WMSA is linearly related to the size of the image, which greatly reduces the computational complexity of the Swin Transformer network and improves the efficiency of the network. When using the WMSA module, due to the self-attention calculation only in each window, there is a lack of information exchange between the non-overlapping windows. In order to solve this problem, the Swin Transformer network introduces a sliding window segmentation, that is, the SWMSA module, as shown in Figure 3.

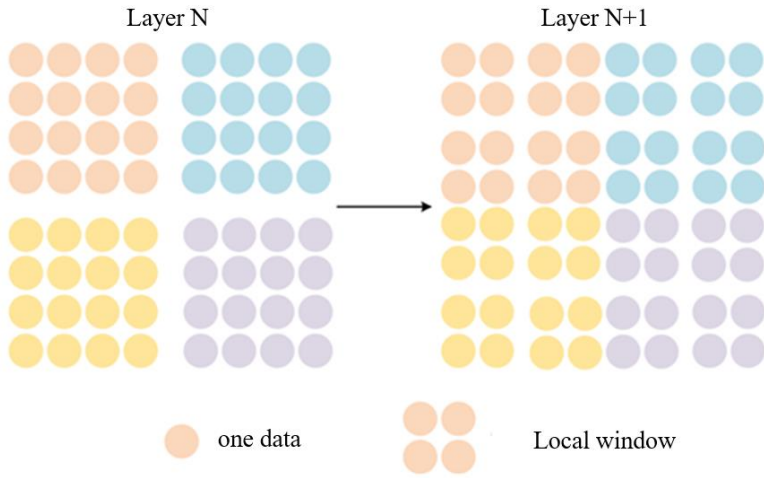

Figure 3: WMSA to SWMSA Transform.

According to Figure 3, on the left side is the WMSA module in the N layer, because the W-MSA module and the SWMSA module are used in pairs, then the N+1 layer is the SWMSA module. It can be seen that from the N layer to the N+1 layer, the window has shifted, such as the window of the first row and the second column in the N+1 layer can make the two windows of the first row in the N layer exchange information, the window of the second row and the second column in the N+1 layer can make the four windows in the N layer exchange information, and the window of the second row and the first column in the N+1 layer can make the two windows of the first column in the N layer exchange information, and the other is the same. This way of shifting the window increases the connection between the adjacent and non-overlapping windows of the previous layer, 188 increases the receptive field, and solves the problem of lack of information exchange between non-overlapping windows. Since the window offset after four windows into nine windows, making the amount of calculation increased, so the use of circular displacement method to deal with the window, as shown in Figure 4.

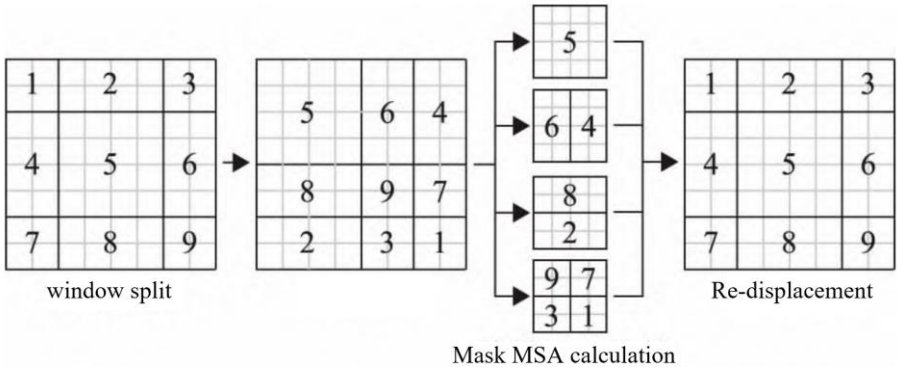

Figure 4: End-around shift.

2.2 Multi-scale Feature Enhancement Module Design

DeepLabv3 [17] can make full use of the context information in the image by introducing mechanisms such as dilated convolution and global pooling. It adopts a multi-scale fusion method to fuse features from different scales, so that it can handle targets at different scales. PSPNet [18] captures context information at different scales through Pyramid Pooling module, which enables the model to understand the relationship between the target and the surrounding environment and better handle targets at different scales. The Inception [19] network extracts multi-level features in images by using multi-scale convolution kernels and receptive fields of different sizes. This structure can help the network capture both low-level features and high-level semantic features, thereby improving the model's ability to understand images. In this paper, a multi-scale feature enhancement module is proposed. Features of different scales can provide different perspectives and context information, which helps to understand and describe images or data more comprehensively. By introducing multi-scale processing in the trunk, features of different scales can be captured, thereby improving the expression ability of features. The model structure is shown in Figure 5.

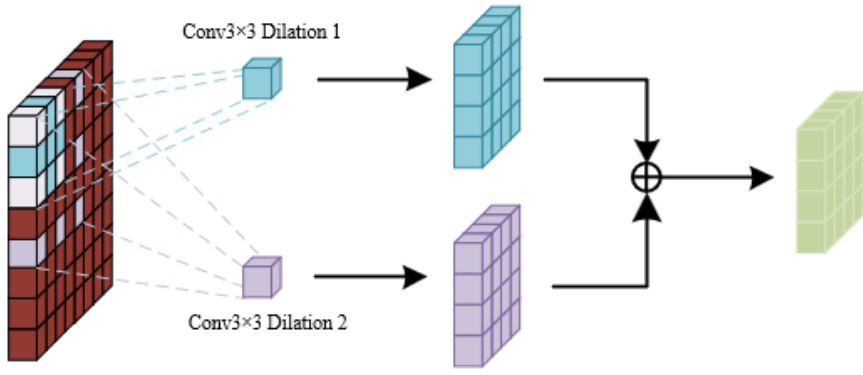

Figure 5: Multi-scale feature enhancement module diagram.

The expression of multi-scale feature enhancement module is as follows:

$$f_{out} = conv3 \times 3(f_{in}) + conv3 \times 3\_D2(f_{in}) \tag{2.8}$$

When dealing with power data, data of different scales can provide information at different levels and angles. By considering data of multiple scales at the same time, more comprehensive and richer information can be obtained, so as to better understand the characteristics and internal structure of data. This is very important for complex data analysis and problem solving.

2.3 Accuracy Improvement of Lightweight Gradient Boosting Tree

Decision tree is a common machine learning algorithm, which is widely used in power load forecasting, credit risk evaluation, highway weaving area merging location and other fields. Decision tree is a multi-classification model using tree model for decision-making. It has the characteristics of low training time complexity, fast prediction process and easy display of the model. The core of the decision tree is to select the optimal attributes for classification. By continuously dividing nodes, the data contained in a branch node is divided into the same category as much as possible.

The general deep learning model uses a simple neural network connection to perform a simple classification of the information extracted by the convolutional network at the end of the network to identify the data. This method will not only increase the number of network parameters, but also cannot make full use of the feature information obtained by convolution. Therefore, it is necessary to replace the last fully connected layer in the deep convolutional network with a lightweight gradient boosting tree [20] for recognition. The lightweight gradient boosting tree uses two solutions: Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB).

For a large number of instances, GOSS retains all instances with large gradients and uses random sampling on instances with small gradients. In order to offset the impact on the data distribution, when calculating the information gain, GOSS introduces a constant multiplier for small gradient data. GOSS first selects Top A instances according to the absolute value of the gradient of the data, and then randomly samples B in the remaining data, and then calculates the information gain by multiplying the sampled small gradient data by (1-A)/B. In this way, the algorithm will pay more attention to the instances with insufficient training, and will not change the distribution of the original data set too much. Therefore, LightGBM adopts a decision tree algorithm based on Leaf-wise, which is a growth strategy based on leaf growth with depth constraints. Most gradient boosting models use Level-wise decision tree algorithm, which is a layer-by-layer growth strategy, as shown in Figure 6. Leaf-wise is a better growth strategy. It finds the leaf node with the largest splitting gain from all the current leaf node nodes for splitting each time. Therefore, compared with the Level-wise growth strategy, Leaf-wise can get better results and faster training speed under the same conditions such as splitting times.

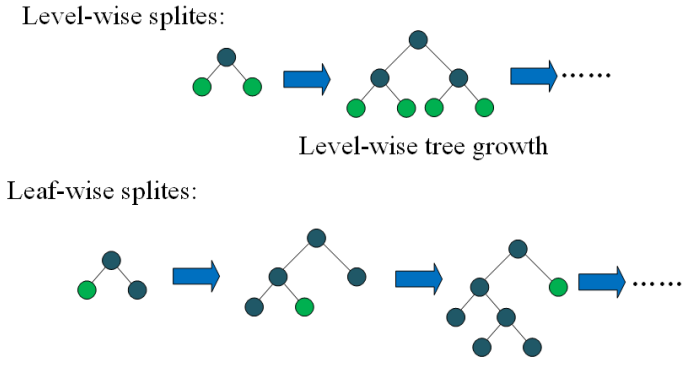

Figure 6: Level-wise and Leaf-wise growth strategies.

They are never non-zero at the same time. The EFB algorithm can transform many mutually exclusive features into low-dimensional dense features, which can effectively avoid the calculation of unnecessary zero-value features, greatly accelerate the training process of GBDT and lose accuracy. In fact, the histogram algorithm is used to mark non-zero elements with tables to ignore zero-value features. By scanning the data in the table, the time complexity of establishing the histogram will be reduced from O (# data) to O (# non _ zero _ data). From the perspective of memory consumption, the histogram algorithm only needs (# data* # features*1Bytes) memory. When finding the segmentation point, the time complexity cost of the histogram algorithm is O (# feature* # data), while in data segmentation, the time complexity cost of the histogram algorithm is only O (# data). In the calculation, the number of split nodes is greatly reduced; when the data

is parallel, the communication cost is greatly reduced. LightGBM also directly supports category features and does not require a one-hot encoding operation, which greatly reduces the data dimension. In addition, Cache hit rate, network communication and parallel computing have a certain degree of optimization, 260 and support GPU acceleration.

3 EXPERIMENTAL SIMULATION

In order to verify the effectiveness of the model proposed in this paper, the performance of the proposed model is tested and compared with the real power data of different substations collected by power companies. The sampling period is to sample the active and reactive values once per minute. The experiment will be compared with autoregressive moving average model (ARIMA), random forest algorithm based on machine learning, SVR algorithm, LightGBM algorithm, XGBoost algorithm, GPR algorithm, DA-RNN algorithm, LSTM algorithm and so on.

3.1 Test Settings

The number of tags for power data is 120,000, of which 96,000 data are used as training sets, 12,000 data are used as validation sets, and the rest are used as test sets. In the network training, the sliding window is used to extract the data. In the detection of outliers, this paper uses three different evaluation indexes of recall rate, precision rate and calculation time to judge the performance of different algorithms. The recall rate is also called the recall rate, which represents the number of positive cases predicted correctly in the sample. The expression is:

$$Recall = \frac{TP}{TP+FN} \tag{3.1}$$

In the formula above, TP indicates that the data is predicted as a positive sample and it is actually the number of positive samples. FN indicates that the data is predicted as a negative sample and it is actually the number of negative samples. The recall rate is relative to the sample, that is, how many positive samples in the sample are predicted correctly, so there are TP, and all positive samples have two destinations, one is judged to be positive, and the other is wrongly judged to be negative, so there are TP + FN in total. The expression of the accuracy of another evaluation index is as follows:

$$Precision = \frac{TP}{TP+FP} \tag{3.2}$$

In the formula above, the meaning of TP is the same as that expressed in the recall rate. FP is the number of positive samples predicted in the data but itself is a negative sample. The accuracy rate is relative to the prediction result, which indicates how many of the samples predicted to be positive are correct; therefore, there are two possible sources for samples with positive predictions. One is to make positive predictions positive, which has TP, and the other is to make negative misjudgments positive, which has FP.

Table 1 gives the index comparison results of this method and several other methods after training on the data set. It can be seen from the table that the algorithm proposed in this paper is superior to other comparison algorithms. ARIMA (Autoregressive Integrated Moving Average) model is a commonly used time series analysis method. It is a linear model and has limited adaptability to nonlinear time series data. There are complex nonlinear patterns in the data, and ARIMA may not be able to capture these features, resulting in inaccurate prediction, so the performance in terms of recall and accuracy is slightly lower than other models. Random Forest performs classification and regression tasks by constructing multiple decision trees and combining their prediction results. It may tend to predict a large number of categories. It can be seen from the table that its accuracy is 91.22%, and the accuracy of SVR is 89.01%. Because the SVR model fits the data through the support vector, and the support vector is the key sample point that determines the model. When there are outliers, they may have a greater impact on the fitting of the model, resulting in a decline in the performance of the model. LightGBM is an efficient gradient boosting framework, which performs well in processing large-scale data sets. As can be seen from the table, its accuracy index is 92.41%. Although XGBoost, GPR, and DA-RNN are relatively fast in training speed, their training time may be longer than other algorithms. Compared with other traditional neural network models,

such as Feedforward Neural Networks and Convolutional Neural Networks, LSTM has higher computational complexity. This is because LSTM performs multiple operations at each time step and involves a large number of gating mechanisms. When processing large-scale data sets or deep networks, the training and reasoning process of LSTM may be slower, but its accuracy is 93.54%. Some deep learning-based methods, such as ExtremeC3Net [21], DANet [22], CGNet [23], PSPNet [18], DeepLabv3 [17], have also achieved good performance in Precision. It can be seen from the table that the model in this paper has a greater advantage than other models in terms of running time in terms of accuracy.

Methods	*Recall (%)*	*Precision (%)*	*Times (s)*
ARIMA	82.31	86.53	7.56
SJSL	87.25	91.22	9.56
SVR	87.12	89.01	6.23
LightGBM	89.46	92.41	7.80
XGBoost	89.26	91.76	12.50
GPR	90.44	91.88	20.77
ExtremeC3Net	90.46	92.01	8.63
DA-RNN	91.55	92.16	24.78
DANet	91.56	93.23	15.16
LSTM	91.23	93.54	14.22
CGNet	91.54	93.78	7.55
PSPNet	91.77	93.96	20.16
DeepLabv3	91.64	94.54	21.22
Ours	**94.11**	**98.56**	**6.94**

Table 1: Power system data anomaly detection index comparison.

The comparison between Precision (%) and Recall (%) is shown in Figure 7. The shorter the time, the higher the Precision (%), the higher the Recall (%), the better the effect. In the figure, it can be understood that the more the point is located in the upper left corner, the better the effect. The color of our model on the graph is red, which is the closest to the upper left corner, and the performance is the most superior.

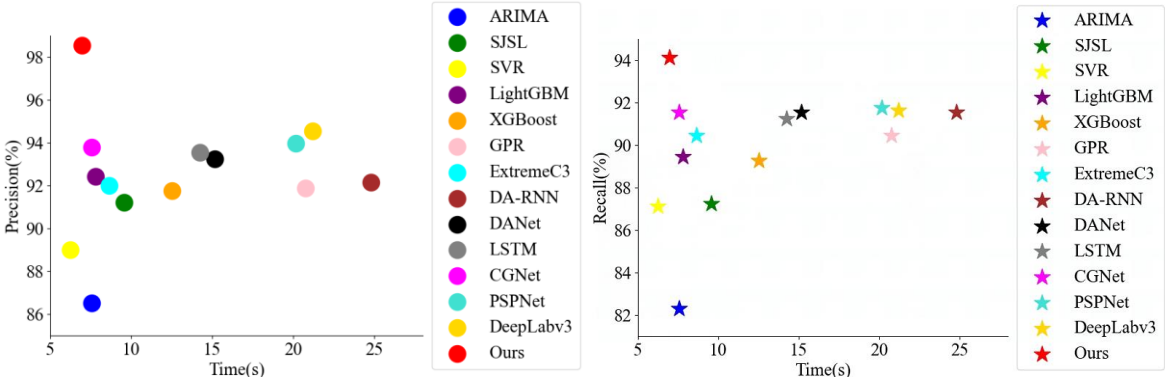

Figure 7: Model performance comparison: (a) precision comparison chart, and (b) recall comparison chart.

3.2 Experimental Simulation Comparison

In Figure 8, four models with the best performance of precision (%) were selected for actual prediction comparison. (a) is GPR prediction graph. (b) is DA-RNN prediction graph. (c) is LSTM prediction graph. (d) is Ours prediction graph. It can be seen from the diagram that the curve displayed by the blue line is the curve where the normal point is located, the abnormal value is marked by the red point, and the range marked by the pink is the range where the normal value is located. The prediction results of CGNet are shown in (a). The pink range in the graph is large, and the model cannot accurately judge whether it is in the normal range, so that there are many missed and false detections. It can be seen from the graph that the four missed areas are marked by the gold box, and the missed and false detections are more serious. The PSPNet prediction map is shown in (b). The pink area in the map is still large, and the prediction of the data does not achieve a better effect. This is due to the lack of feature extraction, and there are also a large number of missed detection and false detection. It can be seen from the figure that there are 4 missed detection areas. The DeepLabv3 prediction map is shown in (c). The pink area in the figure is reduced from the previous two figures, and the missed detection and false detection are also reduced, but it is still serious. In (d) is the detection result of Ours, which is basically consistent with the label, and the detection effect is good. There is no missed detection and false detection when the pink area is not very large. This is because our model has sufficient global information extraction and high recognition accuracy.

Figure 9 is the simulation diagram of power system data section anomaly detection, (a)(b)(c)(d) is the data prediction of four different substations respectively. Comparing Figure 9 (a) and Figure 9 (b), it can be seen that there is a sudden rise of load in A-A substation during the period of 400 to 600 minutes, while the adjacent A-B substation does not rise during this period, so it can be determined that there is data anomaly in A-A substation during the period of 400 to 500 minutes. Comparing Figure 9 (c) and Figure 9 (d), it can be seen that there is a sudden increase in load in both B-A substation and B-B substation. Therefore, the algorithm will determine that this situation may be due to the sudden increase in load caused by stealing electricity and leakage in large enterprises. The algorithm in this paper is an anomaly detection of the whole data, which can quickly identify the sudden drop of single-site data.

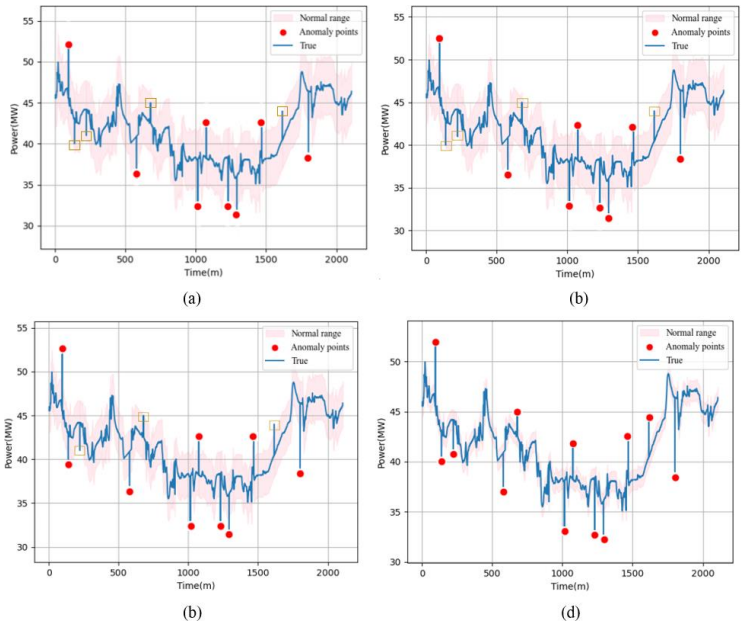

Figure 8: Anomaly detection performance comparison, (a) GPR prediction, (b) DA-RNN prediction, (c) LSTM prediction, and (d) our prediction.

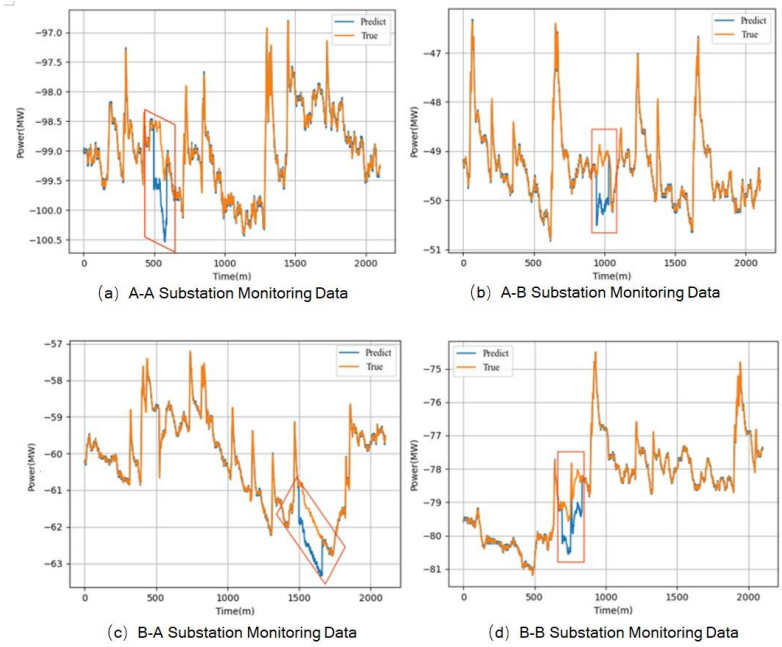

Figure 9: Substation data anomaly detection, (a) A-A substation monitoring data, (b) A-B substation monitoring data, (c) B-A substation monitoring data, and (d) B-B substation monitoring data.

4 CONCLUSION

In order to improve the security protection ability of the power information system, this paper combines the deep learning model with the lightweight gradient lifting tree method, introduces the idea of transformer global attention, and proposes an abnormal behavior detection method of the power monitoring system based on the Swin transformer neural network, which realizes the rapid and accurate identification of the abnormal behavior of the power monitoring system, and reduces the difficulty of the deployment of the computational time overhead. Using the MFE method, a multi-scale method is added at the same feature level to further supplement the information in the backbone network, thereby further improving the accuracy of the network. The experimental results show that the detection accuracy and model training convergence speed of the proposed method are significantly better than other existing methods in the process of anomaly detection of power data.

Xiongbo Xiao, https://orcid.org/0009-0002-6595-7172
Zhonglin Yang, https://orcid.org/0009-0009-6815-072X
Xueping Gao, https://orcid.org/0000-0003-2947-5096

REFERENCES

[1] Anwar, A.; Mahmood, A. N.: Cyber security of smart grid infrastructure, arXiv, abs/1401.3936, 2014, https://api.semanticscholar.org/CorpusID:2435552.
[2] Ghasempour, A.: Internet of things in smart grid: architecture, applications, services, key technologies, and challenges, Inventions, 4(1), 2019, 22, https://doi.org/10.3390/inventions4010022.

[3] Tan, Z.; Zhang, J.; Wang, J.; Xu, J.: Day-ahead electricity price forecasting using wavelet transform combined with ARIMA and GARCH models, Applied energy, 87(11), 2010, 3606–3610, https://doi.org/10.1016/j.apenergy.2010.05.012.
[4] Wei, Y.; Zhang, X.; Shi, Y.; Xia, L.; Pan, S.; Wu, J.; Han, M.; Zhao, X.: A review of data-driven approaches for prediction and classification of building energy consumption, Renewable and Sustainable Energy Reviews, 82, 2018, 1027–1047, https://doi.org/10.1007/978-981-16-2778-1_2.
[5] Wu, R.; Zhang, A.; Tian, X.; Zhang T.: Anomaly detection algorithm for power data based on improved K-means, Journal of East China Normal University (Natural Science Edition), 2020(4), 2020, 79–87, https://xblk.ecnu.edu.cn/EN/10.3969/j.issn.1000-5641.201921012.
[6] Zheng, K.; Wang, Y.; Chen, Q.; Li, Y.: Electricity theft detecting based on density-clustering method, In Proceedings of the 2017 IEEE Innovative Smart Grid Technologies-Asia (ISGT-Asia), 2017, 1–6, https://doi.org/10.1109/ISGT-Asia.2017.8378347.
[7] Sánchez-García, R. J.; Fennelly, M.; Norris, S.; Wright, N.; Niblo, G.; Brodzki, J.; Bialek, J. W.: Hierarchical spectral clustering of power grids, IEEE Transactions on Power Systems, 29(5), 2014, 2229–2237, https://doi.org/10.1109/TPWRS.2014.2306756.
[8] Moulin, L.; Da Silva, A. A.; El-Sharkawi, M.; Marks, R. J.: Support vector machines for transient stability analysis of large-scale power systems, IEEE Transactions on Power systems, 19(2), 2004, 818–825, https://doi.org/10.1109/TPWRS.2004.826018.
[9] Wen, S.; Wang, Y.; Tang, Y.; Xu, Y.; Li, P.; Zhao, T.: Real-time identification of power fluctuations based on LSTM recurrent neural network: A case study on Singapore power system, IEEE Transactions on Industrial Informatics, 15(9), 2019, 5266–5275, https://doi.org/10.1109/TII.2019.2910416.
[10] Mao, W.; Cao, X.; Zhou Q.; Yan, T.; Zhang, Y.: Anomaly detection for power consumption data based on isolated forest, In Proceedings of the 2018 international conference on power system technology (POWERCON), 2018, 4169–4174, https://doi.org/10.1109/POWERCON.2018.8602251.
[11] Jiang, Y.; Zheng, L.; Ding, X.: Ultra-short-term prediction of photovoltaic output based on an LSTM-ARMA combined model driven by EEMD, Journal of Renewable and Sustainable Energy, 13(4), 2021, https://doi.org/10.1063/5.0056980.
[12] Ramirez-Gonzalez, M.; Sevilla, F. S.; Korba, P.: Power System Inertia Estimation Using a Residual Neural Network Based Approach, In Proceedings of the 2022 4th Global Power, Energy and Communication Conference (GPECOM), IEEE, 2022, 355–360, https://doi.org/10.1109/GPECOM55404.2022.9815784.
[13] Tabakhpour, A.; Abdelaziz, M. M.: Neural network model for false data detection in power system state estimation, In Proceedings of the 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE), IEEE, 2019, 1–5, https://doi.org/10.1109/CCECE.2019.8861919.
[14] Yi, H.; Zhang, Z.; Wang, P.: State Estimation of Distribution Network with the Improved Deep Residual Neural Network, In Proceedings of the 2022 IEEE/IAS Industrial and Commercial Power System Asia (I&CPS Asia), IEEE, 2022, 972–977, https://doi.org/10.1109/ICPSAsia55496.2022.9949725.
[15] Simonyan, K.; Zisserman, A.: Very deep convolutional networks for large-scale image recognition, arXiv, 2014, https://doi.org/10.48550/arXiv.1409.1556.
[16] He, K.; Zhang, X.; Ren, S.; Sun, J.: Deep residual learning for image recognition, In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, IEEE, 2016, 770–778, https://doi.org/10.1109/CVPR.2016.90.
[17] Chen, L. C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation, In Proceedings of the European conference on computer vision (ECCV), 2018, 801–818, https://doi.org/10.48550/arXiv.1802.02611.

[18] Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J.: Pyramid scene parsing network, In Proceedings of the IEEE conference on computer vision and pattern recognition, IEEE, 2017, 2881–2890, https://doi.org/10.1109/CVPR.2017.660.
[19] Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.: Inception-v4, inception-resnet and the impact of residual connections on learning, In Proceedings of the AAAI conference on artificial intelligence, 31, 2017, https://doi.org/10.48550/arXiv.1602.07261.
[20] Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T. Y.: Lightgbm: A highly efficient gradient boosting decision tree, Advances in neural information processing systems (NIP 2017), 2017, https://doi.org/10.5555/3294996.3295074.
[21] Park, H.; Sjösund, L. L.; Yoo, Y.; Bang, J.; Kwak, N.: Extremec3net: Extreme lightweight portrait segmentation networks using advanced c3-modules, arXiv, 2019, https://doi.org/10.48550/arXiv.1908.03093.
[22] Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H.: Dual attention network for scene segmentation, In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, IEEE, 2019, 3146–3154, https://doi.org/10.48550/arXiv.1809.02983.
[23] Wu, T.; Tang, S.; Zhang, R.; Cao, J.; Zhang, Y.: Cgnet: A light-weight context guided network for semantic segmentation, IEEE Transactions on Image Processing, 30, 2020, 1169–1179, https://doi.org/10.1109/TIP.2020.3042065.