A Transformer Approach to Identifying False Data Injection Attacks

2024 IEEE 8th Conference on Energy Internet and Energy System Integration (EI2) | 979-8-3315-2352-7/24/$31.00 ©2024 IEEE | DOI: 10.1109/EI264398.2024.10990711

Hu Li
School of Electronic and Electrical Engineering
Ningxia University
Yinchuan, China
18254811767@163.com

Huan Pan
School of Electronic and Electrical Engineering
Ningxia University
Yinchuan, China
pan198303@gmail.com

Jiayi Jin
School of Electronic and Electrical Engineering
Ningxia University
Yinchuan, China
jinjiayi2022@163.com

Mengna Sun
School of Electronic and Electrical Engineering
Ningxia University
Yinchuan, China
18306763792@163.com

***Abstract*—The smart grid is vulnerable to multiple cyberattacks, with false data injection attacks (FDIA) being particularly concerning, as they can evade conventional bad data detection (BDD) systems and disrupt power grid operations. Given the nonlinear nature of state estimation and grid data, along with the temporal continuity of state data, this study introduces a Transformer-based model for detecting FDIA. It utilizes the Hilbert-Huang Transform (HHT) to capture precise time-frequency characteristics from power system time-series data. These features are input into the Transformer, where multi-head attention captures long-range dependencies and improves recognition of complex grid patterns. The application of this technique on the IEEE 14-bus system demonstrates its capability to effectively recognize abnormal patterns with high accuracy. In addition, a basic sensitivity analysis of the control parameters was undertaken.**

Keywords—False data injection attacks, bad data detection, Hilbert-Huang Transform, Transformer

I. INTRODUCTION

The development of smart grids integrates cutting-edge information and communication technologies (ICT) with traditional power networks, with the goal of enhancing the efficiency and resilience of power transmission and management infrastructures. Real-time monitoring and optimization via numerous sensors transform traditional grids into cyber-physical power systems, integrating the information network and physical layers[1]. While ICT improves grid operations' intelligence, it also introduces security risks that could jeopardize smart grids' safe and economic operation[2].

Cyber-attacks increase uncertainty in power system operations and can lead to severe consequences[3], such as the 2015 Western Ukraine blackout[4] and the 2019 Venezuela outage in 21 states[5]. False data injection attacks (FDIA) have drawn significant attention among cyber-physical threats because of their sophisticated and covert nature[6]. By tampering with state estimation data, FDIA can avoid detection by the supervisory control and data acquisition (SCADA) system, resulting in the dispatch of inaccurate information to the grid control center and potentially causing system disruptions. Consequently, the detection and prevention of FDIA are vital for preserving the security and reliability of smart grid functionalities.

FDIA detection strategies are broadly classified into two categories: model-based and data-driven techniques[7]. Model-based approaches depend on system parameters, such as Kalman filter-based detection[8] and unknown input observers [9], but it is challenging to accurately obtain these parameters. In contrast, data-driven approaches avoid the necessity for system parameters by utilizing machine learning (ML) and deep learning (DL) algorithms, including decision trees[10], multilayer perceptrons (MLP)[11], recurrent neural networks (RNN)[12], convolutional neural networks (CNN)[13], and graph convolutional networks (GCN)[14] to detect FDIA. However, RNN models are comparatively slower when handling lengthy sequential data[15].

The Transformer is currently the leading sequence modeling architecture in deep learning, utilizing attention mechanisms to enable parallel processing and effectively capture long-range dependencies[16]. A hybrid approach integrating Transformer and LSTM for real-time intrusion detection was introduced in [17], but functional overlap between the two limits the full exploitation of data features. Building on this, the paper presents an FDIA detection framework utilizing a Transformer encoder:

(1) The Hilbert-Huang Transform (HHT) is applied to process continuous-time sample data, extracting the dynamic changes and abnormal patterns in power system data across different frequencies.

(2) The Transformer encoder is employed to model the sequence of measurements, utilizing the multi-head attention mechanism to effectively capture long-term dependencies and thoroughly analyze the internal feature representations of the measurement data.

(3) The extracted features are passed through a fully connected layer followed by a Softmax layer, which calculates the likelihood of an attack on the sample, thereby completing the detection process.

The structure of this paper is organized as follows: Section II introduces fundamental concepts, including power system state estimation and the principles of FDIA. Section III details the architecture of the proposed detection model and elucidates the functionality of each of its components.Section IV evaluates the proposed method on a constructed dataset, assessing the

This work is supported in part by National Natural Science Foundation (NNSF) of China under Grant 62463026, and in part by the Western Light Talent Training Program of Chinese Academy of Sciences.

model's detection accuracy from various aspects. Finally, this paper concludes with a summary of the overall work.

II. Related Theoretical Background

A. Power System State Estimation

The core concept of state estimation involves evaluating samples of system variables obtained from measurement devices to ascertain the operational status of the power system. Typical measurements include bus voltages, power injections, and power flows across branches. In contemporary smart grids, alternating current (AC) state estimation is extensively utilized, leveraging the nonlinear relationships between system states and measurements. The model can be described as follows:

$$z = h(\boldsymbol{x}) + \boldsymbol{e} \tag{1}$$

where $\boldsymbol{z} = (z_1, z_2 \ldots z_m)^T$ is the measurement vector, where m represents the number of measurements; $\boldsymbol{x} = (x_1, x_2 \ldots x_m)^T$ is state vector, such as voltage magnitudes and phase angles, with n indicating the number of state variables; $\boldsymbol{e} = (e_1, e_2 \ldots e_m)^T$ is the noise, assumed to follow a Gaussian distribution.

To detect erroneous data within the measurements, the weighted least squares method is employed to minimize the difference between the estimated state vector and the actual measurement vector. The values of the estimated state variables are obtained by solving the following equation:

$$\hat{\boldsymbol{x}} = \min_{x} \left(\boldsymbol{z} - h(\boldsymbol{x})\right)^{\mathrm{T}} \boldsymbol{R}^{-1} \left(\boldsymbol{z} - h(\boldsymbol{x})\right) \tag{2}$$

where $\boldsymbol{R}$ denotes the measurement covariance matrix.

B. BDD Mechanism

Measurement data acquired by the SCADA system may become unreliable due to some factors such as equipment malfunctions, transmission noise, or cyber-attacks[18]. Consequently, it is essential to cleanse the measurement data of any erroneous information using BDD before conducting state estimation. The BDD process involves evaluating the 2-norm of the residuals derived from the measurements and comparing them against a predefined threshold τ, as illustrated below:

$$\|\boldsymbol{r}\|_2 = \|\boldsymbol{z} - h(\hat{\boldsymbol{x}})\|_2 \le \tau \tag{3}$$

C. Principle of FDIA

The objective of a false data injection attack (FDIA) is to deceive system operators into accepting manipulated state estimation variables as legitimate. In this scenario, the vector $\boldsymbol{c}$ denotes the deviation in the power system's state variables. To disrupt the control center's normal operations, the attacker modifies the measurements received by the control center to $\boldsymbol{z}_a = \boldsymbol{z} + \boldsymbol{a}$, where a represents the attack vector. At this point, the 2-norm of the residual is calculated as follows:

$$\|\boldsymbol{r}_a\|_2 = \|\boldsymbol{z}_a - h(\boldsymbol{x}_a)\|_2 = \|\boldsymbol{z} + \boldsymbol{a} - h(\hat{\boldsymbol{x}} + \boldsymbol{c})\|_2 \tag{4}$$

To bypass the BDD mechanism, the attacker carefully designs the attack vector $\boldsymbol{a}$ to satisfy the condition $\boldsymbol{a} = \boldsymbol{h}(\hat{\boldsymbol{x}} + \boldsymbol{c}) - \boldsymbol{h}(\hat{\boldsymbol{x}})$, ensuring that the altered measurement residual vector aligns with the residual vector $\boldsymbol{r}$ under normal conditions. This alignment means that the residuals used for state estimation remain consistent, even after the attack:

$$\|\boldsymbol{r}_a\|_2 = \|\boldsymbol{z} - h(\hat{\boldsymbol{x}})\|_2 = \|\boldsymbol{r}\|_2 \le \tau \tag{5}$$

It is clear from (5) that if the attacker possesses complete knowledge of the power system's topology and the attack vector satisfies the necessary condition, the residuals of the state estimation will remain unchanged both before and after the attack. This allows the attacker to execute a stealthy FDIA, effectively masking the attack from detection mechanisms.

III. FDIA Detection Model

A. Overall Model Architecture

As stated in Section I, an attacker is capable of executing an FDIA while remaining undetected. However, deviations in system states can be captured by the HHT. Based on this, we propose an FDIA detection method that combines HHT with a Transformer encoder. The architecture of the model, depicted in Fig.1, consists of an input layer, an HHT layer, a Transformer encoder, a fully connected layer, and a Softmax layer. The model employs HHT to break down historical data and extract relevant features, which are then fed into the Transformer encoder for training. The trained classifier can then detect potential attacks when new data arrives.

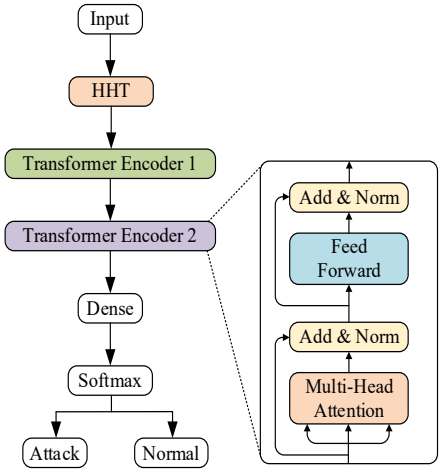


Fig.1: The overall framework of the detection model.

B. Data Feature Extraction Based on HHT

The Hilbert-Huang Transform (HHT) is designed for analyzing nonlinear and non-stationary time series[19]. In power systems, node data, such as voltage, current, and power, are often nonlinear and non-stationary, especially during grid faults or load fluctuations. HHT effectively extracts time-frequency characteristics from this data, aiding in fault detection and stability analysis. Its adaptability allows it to capture local signal features, offering new insights and tools for power system data analysis.

The core of HHT involves decomposing complex signals into intrinsic mode functions (IMFs) and subsequently extracting their instantaneous frequency and amplitude using Hilbert Transform (HT). This process occurs in two stages:

initially, empirical mode decomposition (EMD) separates the signal into IMFs, each representing distinct frequency components. Then, HT is applied to each IMF to reveal the dynamic characteristics and structure hidden in the data.

1) EMD Mechanism

EMD is a fundamental part of HHT, which decomposes the original signal $S(t)$ into a series of IMFs and a residual component R_n. The IMFs, denoted as $c_1(t)$, $c_2(t)$,…, $c_n(t)$, capture various frequency components of the signal, while the residual component represents the portion of the signal not captured by the IMFs. The first intrinsic mode function, IMF1, contains the highest frequency component of the waveform and is typically used as the input for subsequent processing in the HHT method. This decomposition process can be expressed by the following equation:

$$S(t) = \sum_{i=1}^{n} c_i(t) + r_n(t) \tag{6}$$

2) Hilbert Transform

The HT is a powerful mathematical tool used to analyze the IMFs generated by the EMD technique. In signal processing, the HT technique is used to analyze and interpret the spectral properties of each IMF. The instantaneous frequency and amplitude over time can be obtained by applying the HT to each IMF. The expression for the HT is as follows:

$$H\left[c_i(t)\right] = \int_{-\infty}^{\infty} \frac{c_i(\tau)}{\pi(t-\tau)} d\tau \tag{7}$$

The functions $c_i(t)$ and $H[c_i(t)]$ form complex conjugate pair, resulting in the analytic signal $z_i(t)$:

$$z_i(t) = c_i(t) + jH\left[c_i(t)\right] \tag{8}$$

At the same time, $z_i(t)$ can also be expressed as:

$$z_i(t) = a_i(t)\exp\left(j\omega_i(t)\right) \tag{9}$$

Combining the instantaneous amplitude $a_i(t)$ and phase $\theta_i(t)$, we get:

$$a_i(t) = \sqrt{c_i^2(t) + H^2\left[c_i(t)\right]} \tag{10}$$

$$\theta_i(t) = \tan^{-1}\left[\frac{H\left[c_i(t)\right]}{c_i(t)}\right] \tag{11}$$

The instantaneous frequency $\omega_i(t)$ is described as:

$$\omega_i(t) = \frac{d\theta_i(t)}{dt} \tag{12}$$

Consequently, the original data can be represented as follows:

$$S(t) = \mathrm{Re}\sum_{i=1}^{n} a_i(t)\exp\left(j\int \omega_i(t)dt\right) \tag{13}$$

where Re denotes taking the real part.

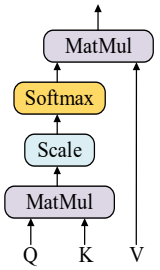
C. Attack Detector Based on Transformer encoder

1) Attention Mechanism

Attention mechanisms are extensively applied in DL, especially for sequential data tasks, as it focuses on different positions within the sequence. The attention mechanism has been shown to aid in tasks like image classification. DL replicates human cognitive and visual processes, allowing neural networks to concentrate on the most important and pertinent aspects of the input data.

2) Scaled Dot-Product Attention Mechanism

Attention mechanism includes both additive and dot-product attention[20]. Due to the efficiency of matrix multiplication, dot-product attention is more computationally and spatially efficient. Therefore, this study utilizes the scaled dot-product attention mechanism, as shown in Fig. 2.



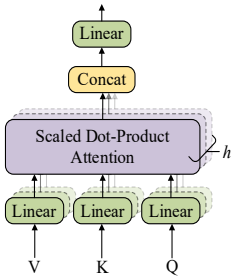


Fig.2: Scaled dot-product attention. Fig.3: Multi-head attention.

The primary idea behind the scaled dot-product attention mechanism is to modify the attention scores to ensure a more stable training process. Initially, the dot product of $\boldsymbol{Q}$ and $\boldsymbol{K}$ is calculated and then scaled by dividing by $\sqrt{d_k}$, which helps stabilize the gradients. The Softmax function is subsequently applied to determine the weights of the value vectors. Finally, a weighted sum is performed to produce the final attention output, as demonstrated in (14). $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$ are identical and represent the input features.

$$Attention(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = Softmax\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{14}$$

where d_k is the dimension of $\boldsymbol{Q}$ and $\boldsymbol{K}$.

3) Multi-Head Attention Mechanism

While single-head attention focuses on sequential relationships, its capacity is limited. Multi-head attention mechanism enhances the model's ability to learn from different subspaces by introducing multiple attention heads[16]. Real-world measurement data, including power, voltage, current, and active/reactive power, exhibit varied relationships. This study employs multi-head attention to capture these variations, allowing the model to simultaneously focus on information from different subspaces.

As shown in Fig. 3, first, the input $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$ undergo h linear transformations, mapping the inputs to multiple different representational subspaces, Here, h denotes the number of attention heads. Subsequently, the scaled dot-product attention module operates concurrently to compute the attention output for each individual head.the outputs from the h-attention heads are concatenated together.Lastly, the concatenated output is passed through an additional linear transformation to generate the final result of the multi-head attention mechanism.

$$head_i = Attention\left(\boldsymbol{Q}_i,\boldsymbol{K}_i,\boldsymbol{V}_i\right), i = 1,...,8 \tag{15}$$

$$\boldsymbol{Q}_i = \boldsymbol{Q}\boldsymbol{W}_i^Q, \boldsymbol{K}_i = \boldsymbol{K}\boldsymbol{W}_i^K, \boldsymbol{V}_i = \boldsymbol{V}\boldsymbol{W}_i^V, i = 1,...,8 \tag{16}$$

$$MultiHead(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = Concact(head_1,\ldots,head_8)\boldsymbol{W}^O \quad (17)$$

where i denotes the index of a specific "head" within the advanced mechanism of multi-head attention, while $\boldsymbol{Q_i}$, $\boldsymbol{K_i}$, and $\boldsymbol{V_i}$ correspond to the query, key, and value outputs resulting from the linear transformations of the i-th head, respectively; Additionally, $\boldsymbol{W^Q}$, $\boldsymbol{W^K}$, $\boldsymbol{W^V}$, and $\boldsymbol{W^O}$ represent the respective weight matrices.

4) Transformer encoder

Vaswani et al. introduced the Transformer model, which set a new benchmark in performance for numerous NLP tasks[16]. The Transformer architecture comprises both encoder and decoder components. The encoder transforms the input sequence into hidden representations and is predominantly utilized for classification tasks. The Transformer encoder is composed of a multi-head attention layer, a feedforward neural network layer, residual connections, and layer normalization.

In the Transformer encoder, each position, a feedforward neural network is used to handle the output from the attention mechanism, enhancing the model's nonlinear modeling capability. The feedforward neural network layer, consisting of two fully connected layers, is relatively simple in design. The initial layer employs the rectified linear unit (ReLU) activation function, while the second uses a linear activation function. The structure is expressed as follows:

$$FFN = \max\left(0, \boldsymbol{X}\boldsymbol{W}_1 + \boldsymbol{b}_1\right)\boldsymbol{W}_2 + \boldsymbol{b}_2 \quad (18)$$

Where $\boldsymbol{X}$ denotes the output generated by the multi-head attention layer, and the output matrix of the feedforward neural network has the same dimensions as $\boldsymbol{X}$; $\boldsymbol{W_1}$, $\boldsymbol{b_1}$, $\boldsymbol{W_2}$, and $\boldsymbol{b_2}$ are the parameters of the two activation functions, respectively.

Residual connections alongside layer normalization are integrated throughout the Transformer network to improve optimization. As a result, the ultimate output of the multi-head self-attention and feedforward network layers is formulated as LayerNorm(x+Sublayer(x)), where Sublayer(x) denotes the function of each individual layer.

Subsequently, after being processed by the Transformer encoder, its output is fed into the dense layer and Softmax layer to obtain the final recognition probability:

$$y = \text{Softmax}\left(\boldsymbol{W}_f m + \boldsymbol{b}\right) \quad (19)$$

where the output of the Transformer encoder is $\boldsymbol{m}$, $\boldsymbol{W_f}$ denotes the weight matrix, $\boldsymbol{b}$ represents the bias term, and y is the output probability from the Softmax layer.

Each sub-layer produces outputs with specific dimensions, enabling effective training. Stacked attention and nonlinear layers allow the Transformer encoder to generate a final, consistent sequence representation.The final output of the model can be expressed as:

$$y_i = \begin{cases} 1, & Attack \\ 0, & Normal \end{cases} \quad (20)$$

The Adam optimizer is used to determine the optimal values for all learning parameters in the attack detection model, facilitating efficient convergence during training. Binary cross-entropy loss serves as the training objective, quantifying the difference between predicted probabilities and actual labels, which helps the model minimize classification errors and improve its accuracy in detecting false data injections.

$$min\left\{-\sum_{i=1}^{D}\left[y_i \log \hat{y}_i + \left(1-y_i\right)\log\left(1-\hat{y}_i\right)\right]\right\} \quad (21)$$

where y_i denotes the actual value of the sample, whereas $\hat{y}_i$ represents the value predicted by the model.

IV. Simulation Analysis

A. Dataset

The IEEE 14-bus test system is selected as the primary testing environment in this section. All network topology details, node data, and branch parameters of the system are sourced from Matpower. Utilizing Matpower's capabilities, a total of 15,000 measurement data samples are meticulously simulated, where standard samples are labeled as 0 and FDIA samples are designated as 1. Each sample within the IEEE 14-bus system encompasses 54 distinct measurements, including voltage magnitudes, bus phase angles, and active/reactive power injections at both the buses and branches. Subsequently, both normal and FDIA samples undergo processing using the HHT for comprehensive feature extraction. The samples are randomly partitioned in an 8:2 ratio to form the training and testing datasets.

B. Simulation Setup

This section primarily conducts simulation tests on the Pycharm simulation platform. All simulations were conducted in Python 3.8, utilizing libraries such as Tensorflow and Sklearn on a PC. The Transformer encoder model is built using deep learning frameworks like Tensorflow. The specific parameters of the model structure are shown in Table I:

Table I: The structural parameters of the model

Parameter Name	Values
Node embedding size	54
Activation function	ReLU
Convolutional layer	1
Convolutional kernel	3
Convolutional kernel size	3*3
Multi-head attention layers	2
Fully connected layers	4
Optimizer	Adam
Batch size	32
Epochs	100
Learning rate α	0.0001

C. Evaluation Metrics

Table II: Confusion matrix

Actual \ Prediction	Normal	Abnormal
Normal	TN	FP
Abnormal	FN	TP

The model's performance is evaluated using a confusion matrix, as shown in Table II. Using the values of TP, FP, TN, and FN, five key metrics are calculated: precision, recall, accuracy, F1-score, and AUC. These metrics collectively assess the model's effectiveness in detecting FDIA. Precision indicates the proportion of correctly classified abnormal nodes, while recall measures the model's ability to accurately identify abnormal nodes. The F1-score is the harmonic mean of precision and recall, balancing the model's performance. Accuracy reflects the model's overall classification effectiveness. AUC is derived from the ROC curve, where each point corresponds to a specific threshold, with TPR as the true positive rate and FPR as the false positive rate. The formulas for these metrics are presented as follows:

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FN}+\text{FP}} \tag{22}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \tag{23}$$

$$\text{F1-Score} = \frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \tag{24}$$

$$\text{TPR} = \frac{\text{TP}}{\text{TP}+\text{FN}}\%, \quad \text{FPR} = \frac{\text{FP}}{\text{FP}+\text{TN}}\% \tag{25}$$

D. Performance Analysis

To prevent overfitting, mini-batch gradient descent is used, dividing the training data into mini-batches of 32 samples for efficient updates and improved generalization. Additionally, 20% of the data is set aside as a validation set to assess generalization. An initial learning rate of 0.001 speeds up optimization, and an early stopping strategy with a 30-epoch patience threshold stops training if validation loss shows no significant improvement.

As illustrated in Fig. 4, the model trains for 100 epochs, with validation loss monitored continuously. Initially, a 0.001 learning rate boosts loss and accuracy, accelerating convergence. As validation loss stabilizes, the learning rate is reduced, enabling finer adjustments and improving FDIA detection robustness.

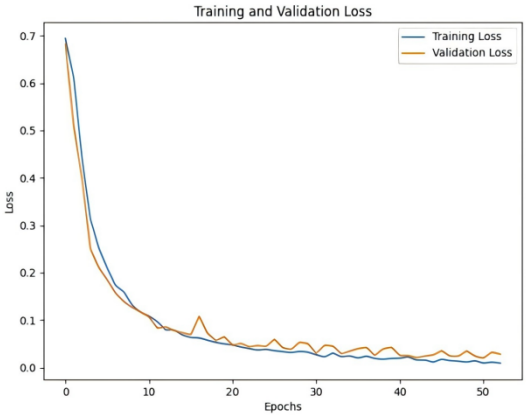


Fig. 4: The loss curve of the model

The optimized model's performance in detecting FDIA on the IEEE 14-bus system is thoroughly analyzed. As shown in Fig. 5, the model initially exhibits considerable errors and low accuracy. However, as the training progresses, accuracy markedly improves and quickly converges, indicating that the model effectively adapts to the data and makes precise predictions.

Fig. 5: The accuracy curve of the model

1) Analysis of the Number of Heads

The mechanism of multi-head attention empowers the model to focus on multiple positions within the input sequence simultaneously, enhancing its ability to capture diverse relationships and dependencies.Each "head" captures the relevance of specific positions. The number of heads, *h*, is a crucial hyperparameter influencing performance, complexity, and generalization. This section analyzes *h* by setting values between 1 and 15 with a step size of 1 for training, and FDIA detection is evaluated using accuracy and other metrics, as shown in Fig. 6.

As shown in Fig. 6, accuracy increases as the *h* value rises from 1 to 8, indicating that multi-head attention helps the model extract diverse feature relationships and improves FDIA detection. Accuracy peaks at *h*=8 and then stabilizes, suggesting that approximately 8 intrinsic relationships exist in the data, covering voltage, phase angles, and power injections. These relationships enhance detection performance, so the number of attention heads in the proposed model is set to 8.

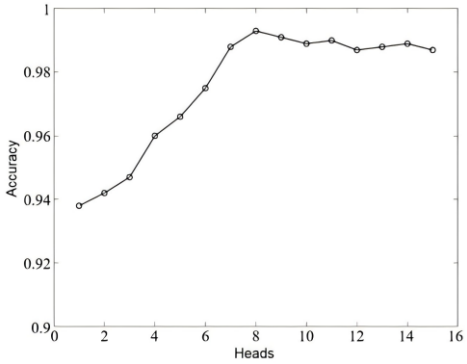


Fig.6: The accuracy change curve under different *h*

2) Comprehensive Comparative Analysis

This section compares the proposed method to traditional algorithms, as shown in Table III. The proposed model achieves notable improvements, with 99.3% accuracy and 99.7%

precision on the IEEE 14-bus system, demonstrating its effectiveness in detecting false data injections. These results highlight the robustness and practical potential of this approach for smart grid security, showing that advanced techniques like the Hilbert-Huang Transform and Transformer architecture significantly boost detection capabilities over conventional methods.

Tabel III: Comparison of detection effects

	Acc	Pre	Rec	F1	time
HHT+DT	0.928	0.928	0.928	0.928	0.033
HHT+MLP	0.931	0.933	0.933	0.933	0.058
HHT+GCN	0.936	0.945	0.931	0.937	0.211
HHT+ANN	0.943	0.954	0.925	0.939	0.112
HHT+CNN	0.971	0.984	0.958	0.971	0.527
Proposed	**0.993**	**0.997**	**0.988**	**0.993**	1.886

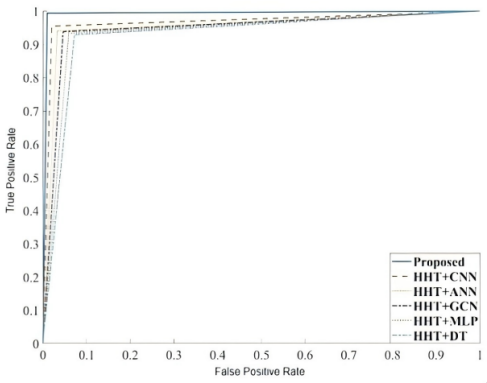


Fig. 7: Comparison of ROC curve and AUC value

Fig. 7 shows the ROC curves and AUC values for six methods, demonstrating changes in TPR and FPR as the threshold shifts from 0 to 1. The proposed method's ROC curve ascends more steeply, reaching an AUC of about 0.99, indicating strong learning ability and effective differentiation between normal and false data samples. This high AUC highlights the model's effectiveness and reliability for smart grid security in real-world applications.

3) Ablation Study

An ablation study, as shown in Table IV, revealed a performance drop without HHT, confirming its essential role. HHT extracts time-frequency features, enabling the Transformer to capture dependencies and subtle anomalies, thereby enhancing detection.

Tabel IV: Comparison ablation research

	Acc	Pre	Rec	F1
Without HHT	0.874	0.882	0.855	0.868
Complete	**0.993**	**0.993**	**0.993**	**0.993**

V. CONCLUSION

This paper combines HHT with the Transformer model to improve FDIA detection in smart grids. HHT enriches input data by extracting frequency and time-domain features, while the Transformer's self-attention captures complex parameter relationships. Simulation on the IEEE 14-bus system shows this approach significantly surpasses traditional methods in accuracy. Parameter tests further refine detection precision. Future work will extend HHT and Transformer applications to areas like industrial control and communication networks, optimize the model with advanced techniques, and address complex FDIA cases, aiming for robust, adaptable smart grid security solutions.

REFERENCES

[1] Yohanandhan R V, Elavarasan R M, Manoharan P, et al. Cyber-physical power system (CPPS): A review on modeling, simulation, and analysis with cyber security applications[J]. IEEE Access, 2020, 8:151019-151064.

[2] Zhao L, Liu Z, Sun G, et al. Cost analysis of false data injection attack based on nonlinear state estimation[J]. Power System Protection and Control, 2019, 47(19): 38-45.

[3] Li Y, Zhang M, Chen C. A deep-learning intelligent system incorporating data augmentation for short-term voltage stability assessment of power systems[J]. Applied Energy, 2022, 308: 118347.

[4] G. Liang, S. R. Weller, J. Zhao, F. Luo, et al. The 2015 Ukraine Blackout: Implications for false data injection attacks[J]. IEEE Transactions on Power Systems, 2017, 32(4): 3317-3318.

[5] Li F, Yan X, Xie Y, et al. A review of cyber-attack methods in cyber-physical power system[C]//2019 IEEE 8th International Conference on Advanced Power System Automation and Protection (APAP). IEEE, 2019.

[6] Liu Y, Ning P, Reiter K M. False data injection attacks against state estimation in electric power grids[J]. ACM Transactions on Information and System Security (TISSEC), 2011, 14(1): 1-33.

[7] Boyaci O, Narimani M R, Davis K R, et al. Joint detection and localization of stealth false data injection attacks in smart grids using graph neural networks[J]. IEEE Transactions on Smart Grid, 2021, 13(1): 807-819.

[8] A. Sargolzaei, K. Yazdani, A. Abbaspour, et al. Detection and mitigation of false data injection attacks in networked control systems[J]. IEEE Transactions on Industrial Informatics, June 2020,16(6), 4281-4292.

[9] Ye J, Yu X. Detection and estimation of false data injection attacks for load frequency control systems[J]. Journal of Modern Power Systems and Clean Energy, 2022, 10(04), 861-870.

[10] Acosta M R C, Ahmed S, Garcia C E, et al. Extremely randomized trees-based scheme for stealthy cyber-attack detection in smart grid networks[J]. IEEE access, 2020, 8: 19921-19933.

[11] Ozay M, Esnaola I, Vural F T Y, et al. Machine learning methods for attack detection in the smart grid[J]. IEEE transactions on neural networks and learning systems, 2015, 27(8): 1773-1786.

[12] Deng Q, Sun J. False data injection attack detection in a power grid using RNN[C]//IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society. IEEE, 2018: 5983-5988.

[13] S. Wang, S. Bi and Y. -J. A. Zhang, Locational detection of the false data injection attack in a smart grid: A multilabel classification approach[J]. IEEE Internet of Things Journal, Sept. 2020, 7(9), 8218-8227.

[14] Edeh V, Mehdi K, Mehdi S, et al. Detection of false data injection attacks in cyber‐physical systems using graph convolutional network[J]. Electric Power Systems Research, 2023, 217: 109118.

[15] Niu X, Li J, Sun J, et al. Dynamic detection of false data injection attack in smart grid using deep learning[C]//2019 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT). IEEE, 2019: 1-6.

[16] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]// Advances in Neural Information Processing Systems, 2017: 5998-6008.

[17] Baul A, Sarker G C, Sadhu P K, et al. XTM: A novel Transformer and LSTM-based model for detection and localization of formally verified FDI attack in smart grid[J]. Electronics, 2023, 12(4): 797.

[18] Luong M T. Effective approaches to attention-based neural machine translation[J]. arXiv preprint arXiv:1508.04025, 2015.

[19] Moslem D, Mohammad G, Taher N, et al. False data injection attack detection based on Hilbert-Huang Transform in AC Smart Islands[J]. IEEE ACCESS, 2020, 8179002-179017.

[20] Chen B, Tang Y. False data injection attack detection in smart grid based on Transformer encoder[J]. Computer Applications and Software, 2022, 39(07): 336-342.