Received: 19 January 2021 | Revised: 25 June 2021 | Accepted: 14 July 2021

IET Energy Systems Integration

DOI: 10.1049/esi2.12034

ORIGINAL RESEARCH PAPER

A deep learning-based classification scheme for cyber-attack detection in power system

Yucheng Ding[1] | Kang Ma[2] | Tianjiao Pu[1] | Xingying Wang[1] | Ran Li[3] | Dongxia Zhang[1]

[1]China Electric Power Research Institute, Beijing, China

[2]Department of Electronic and Electrical Engineering, University of Bath, Bath, UK

[3]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China

Correspondence

Yucheng Ding, China Electric Power Research Institute, Beijing, China.
Email: dingyucheng007@gmail.com

Funding information

National Natural Science Foundation of China, Grant/Award Number: 61703379

Abstract

A smart grid improves power grid efficiency by using modern information and communication technologies. However, at the same time, the system might become increasingly vulnerable to cyberattacks. Among various emerging security problems, a false data injection attack (FDIA) is a new type of attack against the state estimation. In this article, a deep learning-based identification scheme is developed to detect and mitigate information corruption. The scheme implements a Conditional Deep Belief Network to analyse time-series input data and leverages captured features to detect the FDIA. The performance of the detection mechanism is validated by using the IEEE standard test system for simulation. Different attack scenarios and parameters are set to demonstrate the feasibility and effectiveness of the developed scheme. Compared with the support vector machine and the multilayer perceptrons, the experimental analyses indicate that the results of the proposed detection mechanism are better than those of the other two in terms of FDIA detection accuracy and robustness.

KEYWORDS

conditional deep belief network, cyber security, deep learning, false data injection attacks detection, feature extraction, smart grids, state estimation

1 | INTRODUCTION

A power system is a complex and interconnected network that transfers electrical energy from generators to users [1, 2]. The power grid is continuously operated and monitored by the Supervisory Control and Data Acquisition System (SCADA) to ensure a normal operating condition. In particular, the state of the power system is estimated by the measured value, and the system operators use the estimated state to control the actual operation [3–5].

By integrating various advanced communication technologies, the power system is moving towards the direction of the smart grid [6–8]. However, the power grid is facing increasing security challenges. Physical security and cyber security are two significant aspects of power system security. Physical security is the ability of a power system to maintain continuous supply in the event of equipment breakdowns. Cyber security refers to the security of a SCADA system that maintains the operation of the power system. Recently, cyberattacks have gradually threatened modern power systems due to the ubiquitous use of communication technologies [9–11]. Besides, because of the close interlinking between the physical and SCADA systems, the physical security of power systems can be compromised by cyber security vulnerabilities [12–14].

Cyberattacks have led to numerous incidents and have been a cause of concern for both power system operators and users. They can undermine or even completely disrupt the control system of the power grid. For instance, in 2010, the Iranian nuclear power plant was invaded by a Stuxnet worm that falsely altered the system status, which spread across the whole SCADA system and disrupted the system protection strategies. On 23

December 2015, three Ukrainian regional electric power distribution companies experienced a cyberattack, which caused power outages affecting nearly 225,000 customers for several hours. Barely a month after the incident, ransomware attacked the Israel Electric Authority through online phishing. The events are fresh examples of the vulnerability of a highly automated smart grid to cyberattacks. This article considers a new type of attack, the false data injection attack (FDIA), which is regarded as a severe threat to the SCADA system.

There is a growing body of literature that recognises the FDIA. Studies over the past decades have provided valuable information on the FDIA scenarios and the corresponding detection strategies. Reference [15] investigated the detection of the FDIA by a strategically selected set of measurements and state variables. The authors show that it is useful to defend against such attacks by protecting a set of basic measurements. Reference [16] considered the budget for protection and candidate methods for perfect protection and partial protection. In [17], the authors introduced the attack model with the least amount of effort and formulated the attack strategy in which several metres are selected for manipulation to cause maximum damage. To defend against the attacks, the authors also investigated the protection-based defence and detection-based defence. In [18], the problem of false data detection was modelled as a matrix separation problem. The nuclear norm minimisation method and low rank matrix factorisation method are presented. The authors in [19] introduced two distributed detection methods: the Distributed Observable Island Detection (DOID) algorithm and Distributed Time Approaching Detection (DTAD) algorithm. In [20], the equivalent measurement transformation and the residual researching method are utilised to identify false data. However, the existing works mainly focus on detecting the measurements at the current moment, and the temporal characteristics of the FDIA have not been studied yet. This article detects the FDIA by using the deep learning algorithm. It extracts features from time-series data and has the advantage of considering data correlation and solving non-linear problems. To some extent, the traditional method strongly depends on the prescribed bad data detection threshold and is sensitive to environmental noise, which can be overcome by the deep learning algorithm.

In the recent past, a Deep Belief Network (DBN) was proposed as an unsupervised learning method to learn the hierarchical representations and correlation from real-time data [21, 22]. It is one of the basic deep learning technologies built by stacking restricted Boltzmann machines (RBMs) [23–25]. By implementing automatic feature extraction, the DBN can achieve higher efficiency and accuracy than traditional machine learning algorithms [26–28]. Although the DBN has good performance in static modelling, it encounters challenges in capturing complicated temporal dynamics from time-series input [29]. In light of this, this article develops an extended version of the DBN, called the Conditional Deep Belief Network (CDBN), which updates a Conditional Gaussian–Bernoulli RBM (CGBRBM) to model temporal data [30–32]. The CDBN-based approach can then identify the hidden correlation and estimate the reliability of the measurement data. The main contributions of this article are as follows:

- The standard DBN is improved to deal with the continuous real-time series data of the power system flexibly and extract the time correlation.
- A CDBN-based FDIA detection scheme is proposed to evaluate the reliability of the measurement and ensure the safe and stable operation of the power grid.
- By simulating different attack scenarios, the performance of the proposed scheme is evaluated from multiple aspects to ensure its feasibility and effectiveness.

Section 2 presents the system model, the state estimation, and the conventional bad data detection (BDD) system. Section 3 mathematically models the FDIA. Section 4 presents the basic principles of the CDBN and formulates a deep learning-based detection scheme. Section 5 performs case studies to evaluate the performance and effectiveness of the developed methodology. The last section draws conclusions and suggests future work.

2 | SYSTEM MODEL

2.1 | State estimation in power systems

Generation, transmission, and distribution are the three main parts of a power system. In a power grid, the control centre must monitor the state of all buses and nodes to make operational decisions as quickly as possible. But it is impossible to measure all the data directly. The control centre estimates the operating conditions of the system by collecting the readings from remote metres.

Let $\mathbf{z} = [z_1, z_2, \ldots, z_m]^{\mathrm{T}}$ be an $m \times 1$ vector of all measurements, including loads and power injections at buses, power flows at transmission lines, and so on. $\mathbf{x} = [x_1, x_2, \ldots, x_n]^{\mathrm{T}}$ denotes an $n \times 1$ state vector, where $m \gg n$. $\mathbf{e} = [e_1, e_2, \ldots, e_m]^{\mathrm{T}}$ is a measurement error vector. We have

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

where $h(\cdot)$ shows the non-linear relationship between measurement $\mathbf{z}$ and state $\mathbf{x}$. In a DC power flow model, Equation (1) can be written in the form of a linear matrix:

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{e}, \tag{2}$$

where $\mathbf{H}$ is an $m \times n$ Jacobian matrix, and e $\sim \mathcal{N}(0, \sigma^2)$ is the environmental noise. On this basis, the state vector can be calculated by

$$\hat{\mathbf{x}} = \left(\mathbf{H}^T\mathbf{W}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{W}\mathbf{z}, \tag{3}$$

where

$$\mathbf{W} = \begin{bmatrix} \sigma_1^{-2} & & & \\ & \sigma_2^{-2} & & \\ & & \ddots & \\ & & & \sigma_m^{-2} \end{bmatrix} [5]. \tag{4}$$

2.2 | Conventional bad data detection

Erroneous data measurements can occur for a variety of reasons (e.g., device misconfiguration and malicious attacks). These measurements can get incorrect state estimates. Therefore, they must be recognised and removed in time. The BDD system can eliminate some random errors. When detecting and identifying the erroneous data, the L2-norm of measurement residual is first calculated. By comparing the calculated result r with a prescribed threshold τ, it reports normal data measurements if

$$r = \| \mathbf{z} - \mathbf{H}\hat{\mathbf{x}} \| < \tau \tag{5}$$

holds or bad ones otherwise.

3 | FALSE DATA INJECTION ATTACK

When an adversary launches the FDIA, he can manipulate measurement $\mathbf{z}$ to cause an arbitrary change in the estimated value without being detected by the BDD system [33]. Figure 1 presents the process when the state estimation is attacked. Under the condition of the FDIA, an original measurement $\mathbf{z}$ can be replaced by a compromised $\mathbf{z}_a$, where $\mathbf{z}_a = \mathbf{z} + \mathbf{a}$ and $\mathbf{a}$ is an $m \times 1$ malicious data vector. If so, the result of the state estimation then becomes $\hat{\mathbf{x}}_a$. In general, the BDD system is likely to recognise the random attack vector $\mathbf{a}$. However, in [33], it has been found that a few well-designed attack vectors (such as $\mathbf{a} = \mathbf{Hc}$) can bypass the BDD because the injected false data does not affect the residue:

$$\mathbf{z}_a - \mathbf{H}\hat{\mathbf{x}}_a = \mathbf{z} + \mathbf{a} - \mathbf{H}(\mathbf{x} + \mathbf{c}) = \mathbf{z} - \mathbf{H}\hat{\mathbf{x}}, \tag{6}$$

where

$$\begin{aligned} \hat{\mathbf{x}}_a &= \left(\mathbf{H}^T\mathbf{W}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{W}\mathbf{z}_a \\ &= \left(\mathbf{H}^T\mathbf{W}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{W}(\mathbf{z}+\mathbf{a}) \\ &= \left(\mathbf{H}^T\mathbf{W}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{W}\mathbf{z} + \left(\mathbf{H}^T\mathbf{W}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{W}\mathbf{a} \\ &= \hat{\mathbf{x}} + \left(\mathbf{H}^T\mathbf{W}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{W}\mathbf{H}\mathbf{c} \\ &= \hat{\mathbf{x}} + \mathbf{c}, \end{aligned} \tag{7}$$

and $\mathbf{c} = [c_1, c_2, \ldots, c_n]^{\mathrm{T}}$ is an arbitrary $n \times 1$ vector. Therefore, the attack is stealthy and can inject any malicious data into the state estimation.

However, adversaries can usually only compromise a limited number of measurements; so two main realistic attack scenarios are considered as follows:

1) Least-effort attack [17]: $k = 1$, adversaries manipulate the minimum number of measurements to launch the FDIA.
2) Multiple attacks [33]: $k > 1$, adversaries can compromise up to k measurements to launch the FDIA,

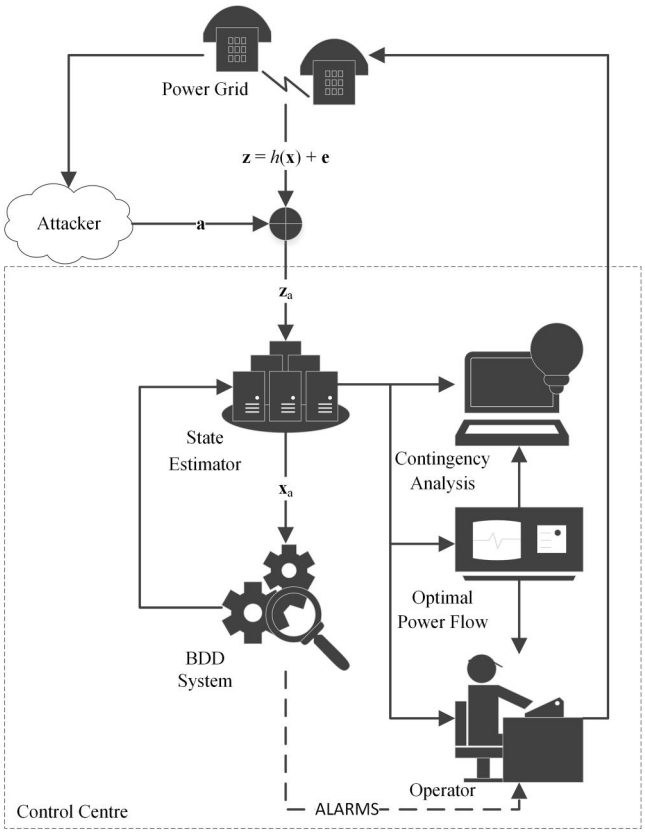

FIGURE 1 The state estimation under the attack [13]

where k is the number of attacked measurements. But the FDIAs are not constrained by these two scenarios. In the IEEE 14-bus test system, Figure 2 shows the difference in the economic dispatch of the power system before and after measurement $\mathbf{z}$ is attacked. We can see that the total generation and the production cost are higher than those of the original case. Furthermore, as the attack intensity increases, the difference increases accordingly. We find that the FDIA can leave the system out of control and even cause security risks. Our developed scheme can specifically detect this kind of attack.

4 | DEEP-LEARNING-BASED IDENTIFICATION SCHEME

In order to detect the FDIA, a deep-learning-based identification scheme is developed. We propose a CDBN by combining a conventional DBN with a CGBRBM, which can process real-valued data and consider the impact of previous measurements on current detection results. Figure 3 shows the framework of the CDBN. We employ a CGBRBM and stack K–1 standard RBMs on top, where K is the number of hidden layers. To indicate whether the measurements are attacked by the FDIA, a BP output unit is added at the end of the scheme to make it a binary classifier.

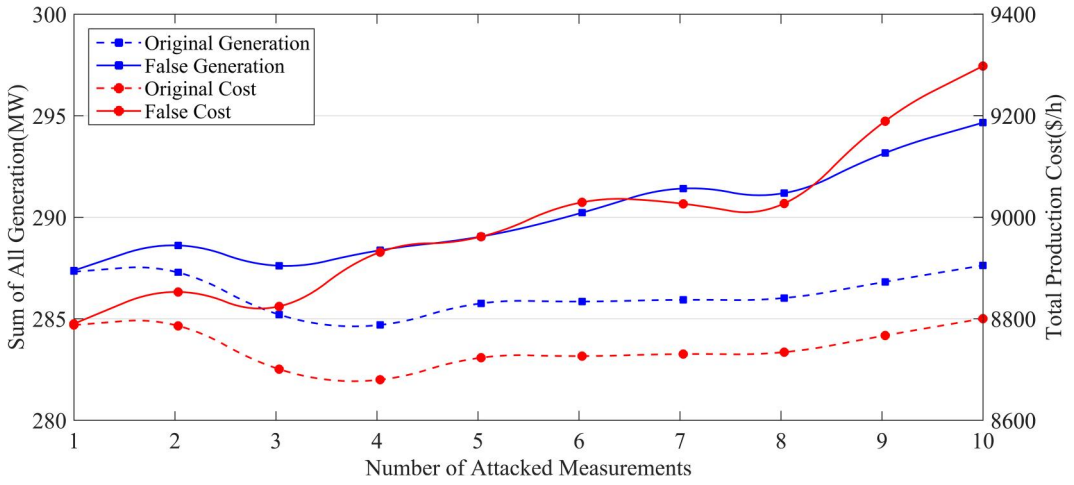

FIGURE 2 The immediate damaging effect to the power system

4.1 | Conventional RBM

The RBM is a two-layer neural network, which is the core of the CDNB. As Figure 4 shows, its two layers are the visible layer and the hidden layer. The units between adjacent layers are connected, but there is no connection inside each layer. The visible layer corresponds to the measurement, and the hidden layer can represent feature extraction.

The RBM is an energy-based undirected generation model and its system energy is

$$\mathrm{E}(\mathbf{v},\ \mathbf{h}) = \sum_{i=1}^{n}\sum_{j=1}^{m} v_i w_{ij} h_j - \sum_{i=1}^{n} a_i v_i - \sum_{j=1}^{m} b_j h_j. \qquad (8)$$

where v_i and h_j are the state of visible unit i and hidden unit j, w_{ij} is the weight between them, a_i and b_j index the standard biases, n and m are the numbers of visible and hidden units, respectively. According to the property of the RBM, given the state of the visible layer, the activation probability of the jth hidden unit is as follows:

$$P(h_j = 1 \mid \mathbf{v}) = sigm\left(\sum_{i=1}^{n} w_{ij} v_i + b_j\right). \qquad (9)$$

Similarly, given the state of the hidden layer, the activation probability of the ith visible unit is as follows:

$$P(v_i = 1 \mid \mathbf{h}) = sigm\left(\sum_{j=1}^{m} w_{ij} h_j + a_i\right), \qquad (10)$$

where $sigm(x) = 1/(1 + \exp(x))$.

The goal of the RBM training is to obtain the parameters to maximise the likelihood function by gradient descent. By calculating the derivative of the log-likelihood, the weights and the biases can be updated as follows:

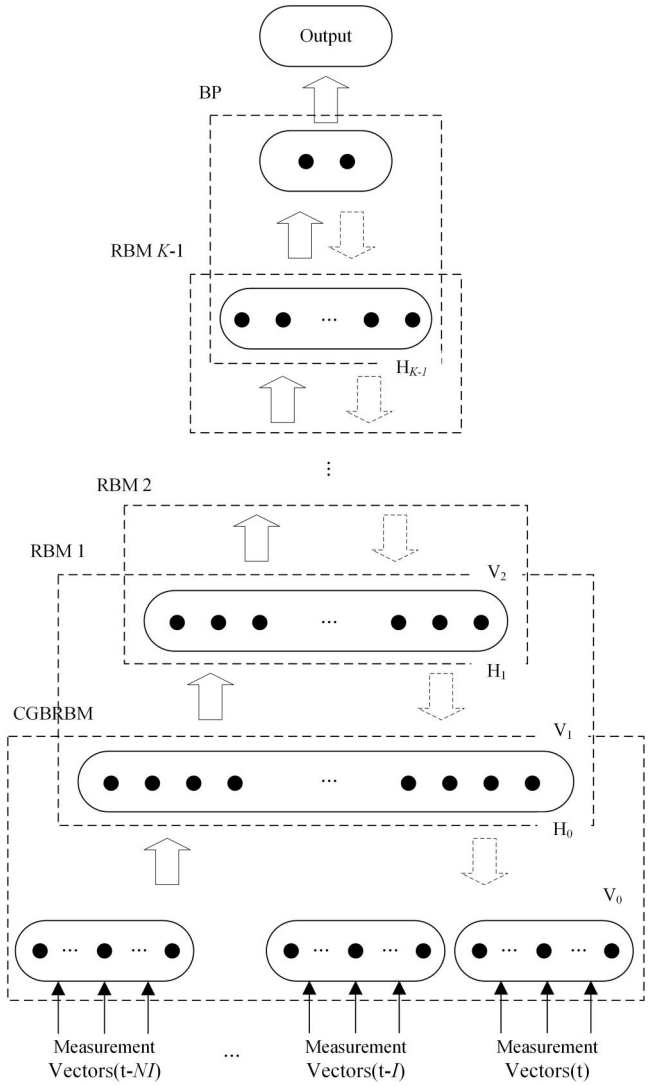

FIGURE 3 The structure of the Conditional Deep Belief Network

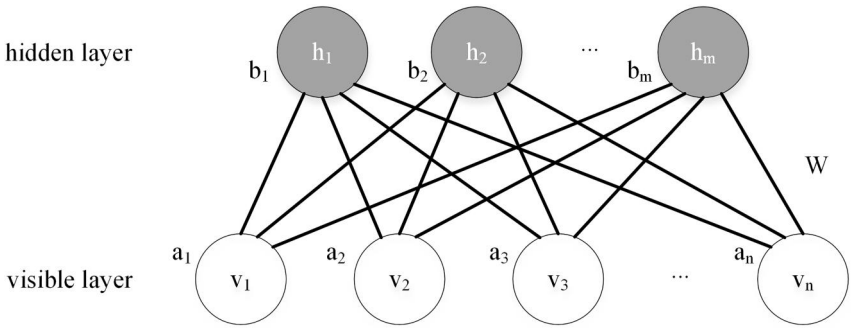

FIGURE 4 The structure of the restricted Boltzmann machine

$$\begin{cases} w_{ij} = w_{ij} + \varepsilon\left(\langle v_i h_j\rangle_{\text{data}} - \langle v_i h_j\rangle_{\text{model}}\right), \\ a_i = a_i + \varepsilon\left(\langle v_i\rangle_{\text{data}} - \langle v_i\rangle_{\text{model}}\right), \\ b_j = b_j + \varepsilon\left(\langle h_j\rangle_{\text{data}} - \langle h_j\rangle_{\text{model}}\right), \end{cases} \tag{11}$$

where ε is the learning rate, $\langle . \rangle_{\text{data}}$ and $\langle . \rangle_{\text{model}}$ are the expectations calculated from the data and model distributions, respectively. $\langle . \rangle_{\text{data}}$ is easily obtained by Equations (9) and (10). However, getting $\langle . \rangle_{\text{model}}$ is much more difficult. To simplify the process, Hinton proposed an efficient and straightforward Contrast Divergence (CD) algorithm based on Gibbs sampling [34].

4.2 | Conditional Gaussian–Bernoulli RBM

In the standard type of the RBM, input data are binary and static, but the measurements in the power system are usually real-valued and time-series data. To address this limitation, we adopt a conditional Gaussian–Bernoulli RBM (CGBRBM) as the basis for the detection algorithm.

It can be seen from Figure 5 that the CGBRBM is a variant of the conventional RBM. First, the input units are linear with Gaussian noise, whereas the hidden units are still binary. The second improvement is that the time-series data can be modelled by considering the visible variables in previous time steps. The energy function of the CGBRBM is as follows:

$$\begin{aligned} E(\mathbf{v}_t, \ldots, \mathbf{v}_{t-NI}, \mathbf{h}) = & \sum_{i=1}^{n} \frac{\left(v_{i,t} - a_{i,t}\right)^2}{2\sigma_i^2} \\ & - \sum_{j=1}^{m} b_{j,t} h_j - \sum_{i=1}^{n}\sum_{j=1}^{m} \frac{v_{i,t}}{\sigma_i} w_{ij} h_j, \end{aligned} \tag{12}$$

where $v_{i,t}$ is the ith real-valued visible element at time step t, h_j is the state of hidden unit j, w_{ij} expresses the weight between $v_{i,t}$ and h_j, σ_i is the standard deviation of the ith visible element, N is the size of the observation window at the previous time, I represents the time interval between two adjacent time steps, and n and m are the numbers of visible and hidden units, respectively. $\mathbf{a}_t = \mathbf{a} + \sum_{k=1}^{N} \mathbf{v}_{t-kI} \boldsymbol{A}_k$ and $\mathbf{b}_t = \mathbf{b} + \sum_{k=1}^{N} \mathbf{v}_{t-kI} \boldsymbol{B}_k$ represent the dynamic biases from the past to the visible bias vector $\mathbf{a}$ and the hidden bias vector $\mathbf{b}$, where $k = 1, \ldots, N$, $\mathbf{v}_{t-kI}$ is the kth previous visible vector, $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$ are the weight matrices of the kth previous visible vector to the current visible unit and the hidden unit, respectively. According to Equation (12), the corresponding activation probabilities become

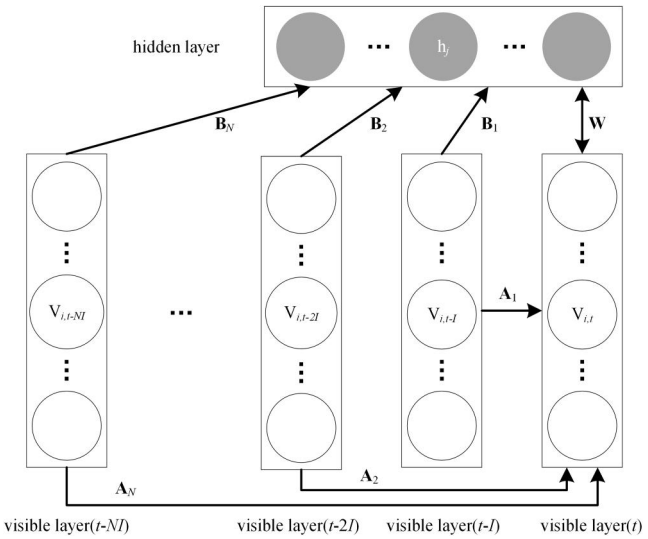

FIGURE 5 The structure of the conditional Gaussian–Bernoulli RBM

$$P\left(h_j = 1 \mid \mathbf{v}_t, \ldots, \mathbf{v}_{t-NI}\right) = sigm\left(\sum_{i=1}^{n} w_{ij} \frac{v_{i,t}}{\sigma_i} + b_{j,t}\right), \tag{13}$$

$$P\left(v_{i,t} = 1 \mid \mathbf{h}\right) = \mathcal{N}\left(\sum_{j=1}^{m} w_{ij} h_j + a_{i,t}, \sigma_i^2\right), \tag{14}$$

where (μ, σ^2) is a Gaussian distribution with mean μ and variance σ^2. In practice, when ${\sigma_i}^2$ is fixed to 1, it can make the learning work better [30]. So, in this case, similar to the conventional RBM, by using the CD algorithm, we can update the weights and the biases as follows:

$$\begin{cases} w_{ij} = w_{ij} + \varepsilon\left(\langle v_{i,t} h_j\rangle_{\text{data}} - \langle v_{i,t} h_j\rangle_{\text{model}}\right), \\ a_{ijk} = a_{ijk} + \varepsilon\left(\langle v_{i,t-kI} v_{j,t}\rangle_{\text{data}} - \langle v_{i,t-kI} v_{j,t}\rangle_{\text{model}}\right), \\ b_{ijk} = b_{ijk} + \varepsilon\left(\langle v_{i,t-kI} h_j\rangle_{\text{data}} - \langle v_{i,t-kI} h_j\rangle_{\text{model}}\right), \\ a_{i,t} = a_{i,t} + \varepsilon\left(\langle v_{i,t}\rangle_{\text{data}} - \langle v_{i,t}\rangle_{\text{model}}\right), \\ b_j = b_j + \varepsilon\left(\langle h_j\rangle_{\text{data}} - \langle h_j\rangle_{\text{model}}\right), \end{cases} \tag{15}$$

where a_{ijk} and b_{ijk} are the elements of $\boldsymbol{A}_k$ and $\boldsymbol{B}_k$.

4.3 | CDBN

The CDBN is a probability generation model. It is a deep learning classifier composed of the CGBRBM, the RBM, and the BP [35]. As Figure 3 shows, the data are first input into the CGBRBM at the bottom for training and feature extraction. Then, the extracted features are used as the input values of another RBM. In this way, more RBM layers can be stacked [21]. As shown in Figure 6, the training process of the CDBN model consists of two steps [23]: layer-wise unsupervised learning and fine-turning.

The first step is an unsupervised learning process. By using the CD algorithm, the RBM of each layer is trained layer-by-layer. Finally, we get the CDBN with a few layers, the parameters of which are suitable for extracting the characteristics of this type of data [24].

In order to optimise the parameters mapped to each layer, the whole CDBN model should be fine-tuned. This process uses the labelled data and the BP network for top-down supervised learning. The binary output node can be calculated by Equation (9), and it can be utilised to represent the compromised label and the normal one. In the calculation of the kth hidden layer, the weights and the biases are updated in the following:

$$\begin{cases} \Delta W_{k,i,j} = -\eta \delta_{k,j} p_{k-1,i}, \\ \Delta b_{k,j} = -\eta \delta_{k,j}, \end{cases} \tag{16}$$

where η is the learning rate, $p_{k-1,i}$ is the ith activation probability of the $(k-1)$th hidden layer, and

$$\delta_{k,j} = p_{k,j}\left(1 - p_{k-1,j}\right) \sum_{h}^{H} \delta_{k+1,h} W_{k+1,j,h}, \tag{17}$$

where $p_{k,j}$ is the jth activation probability of the kth hidden layer, $W_{k+1,j,h}$ is the j,hth element of the $(k+1)$th layer weight matrix, H is the number of elements. Correspondingly, for the output layer, the updated values of the weights and the biases are as follows:

$$\begin{cases} \Delta W_{i,o} = -\eta \delta_o p_{K,i}, \\ \Delta b_o = -\eta \delta_o, \end{cases} \tag{18}$$

where $p_{K,i}$ is the ith activation probability of the last RBM layer, and

$$\delta_o = p_o(1 - p_o)(l_o - L), \tag{19}$$

where p_o is the activation probability of the output layer and l_o and L represent the predicted value and the actual one, respectively.

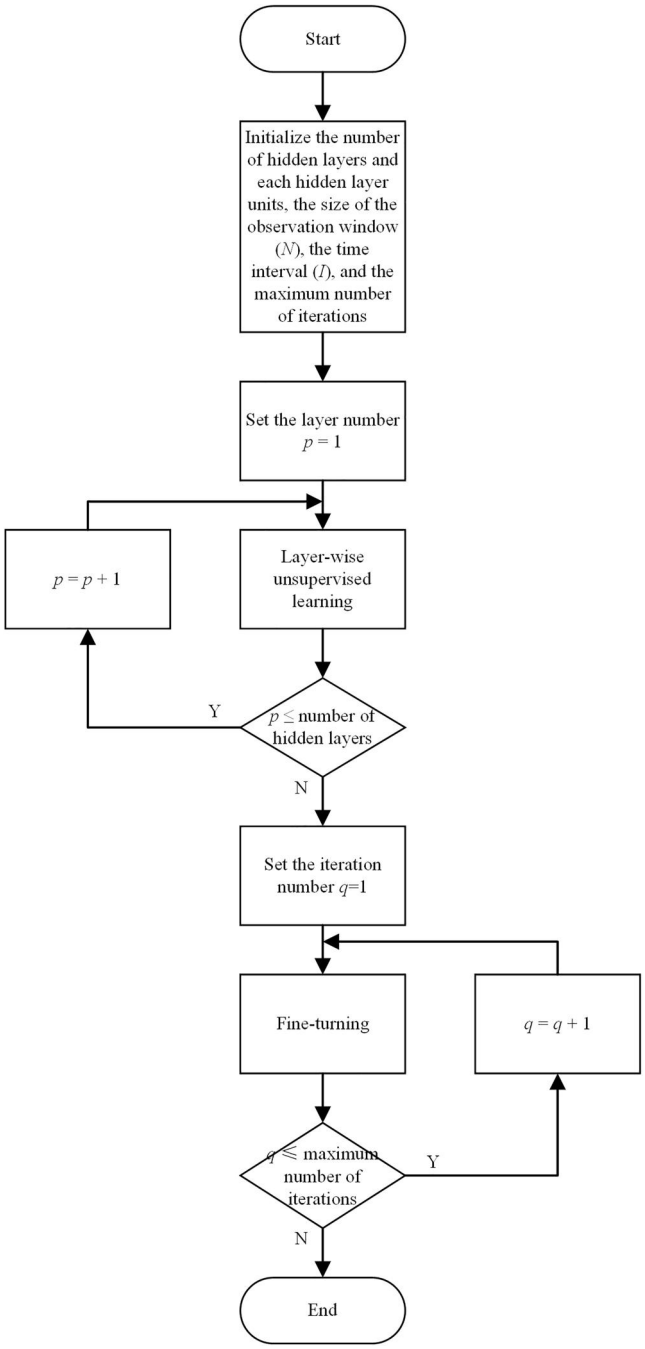

FIGURE 6 The training process of the Conditional Deep Belief Network

As shown in Figure 7, the detection process of our scheme can be mainly divided into three steps: data preprocessing stage, training stage, and testing stage. The first stage is to obtain the measurement vector **z** and inject the attack vector **a** into it according to a certain proportion. After the normalisation process, some sample data are selected as the training set and others as the test set. Next, by completing layer-wise unsupervised learning and fine-turning, the model is trained in the second stage. Finally, the trained model is used to predict whether the sample data in the test set is under attack. By

comparing with the actual value, the accuracy of our developed scheme can be evaluated.

5 | SIMULATION

In this simulation, the performance of our developed scheme is evaluated in the IEEE 14-bus test system. All the data used in the simulation, including the vector of measurements and the Jacobian matrix H, are based on the MATPOWER toolbox. In the IEEE 14-bus test system, by changing the active and reactive power of the load, we first use MATPOWER to complete the power flow calculation for 30,000 consecutive moments. Then, some values (including the branch power flow, the active and reactive power of the generator, and the node voltage, a total of 39 values) are selected from the calculation results of each power flow, and Gaussian noise is injected into them. Finally, the calculation result is regarded as the measurement of state estimation. There are 30,000 measurements in total, and the number of elements in each measurement is 39. Next, according to the method in [33], the FDIA is launched randomly on 15,000 measurements. The measurement residual after the attack is guaranteed to be less than the prescribed threshold τ so as to avoid bad data detection. These 30,000 measurements are divided into three parts on average, that are used as the training set, the verification set, and the test set, respectively. For the above two scenarios (least-effort attack and multiple attacks), we consider the following four aspects to evaluate the performance of the mechanism. Each value of the simulation is an average among 30 independent trials.

5.1 | Structural design

5.1.1 | Effect of the height and width of the CDBN

We first study the effect of the number of hidden layers and the number of units per layer on the performance of our developed scheme. In this simulation, the number of attacked measurements k is set to 1, the size of the observation window (N) is 1, the time interval (I) is 2, and the number of hidden layers is changed from 2 to 5, the hidden layer units range from 20 to 60. From Figure 8, when there are 3 hidden layers and the number of units in each layer is 30, we can see that the accuracy can be up to 97.3%. This result may be explained by the fact that setting too few or too many layers and hidden units may cause under-fitting and over-fitting, respectively [36].

5.1.2 | Effect of the observation window structure

Next, we consider the effect of the size of the observation window at the previous time (N) and the time interval (I) between two adjacent time steps on the effectiveness of our scheme, where N and I are defined in Section 4. Generally speaking, by choosing an appropriate combination of N and I, the accuracy of the mechanism can be significantly improved. If N is larger or smaller than what is required, the observation

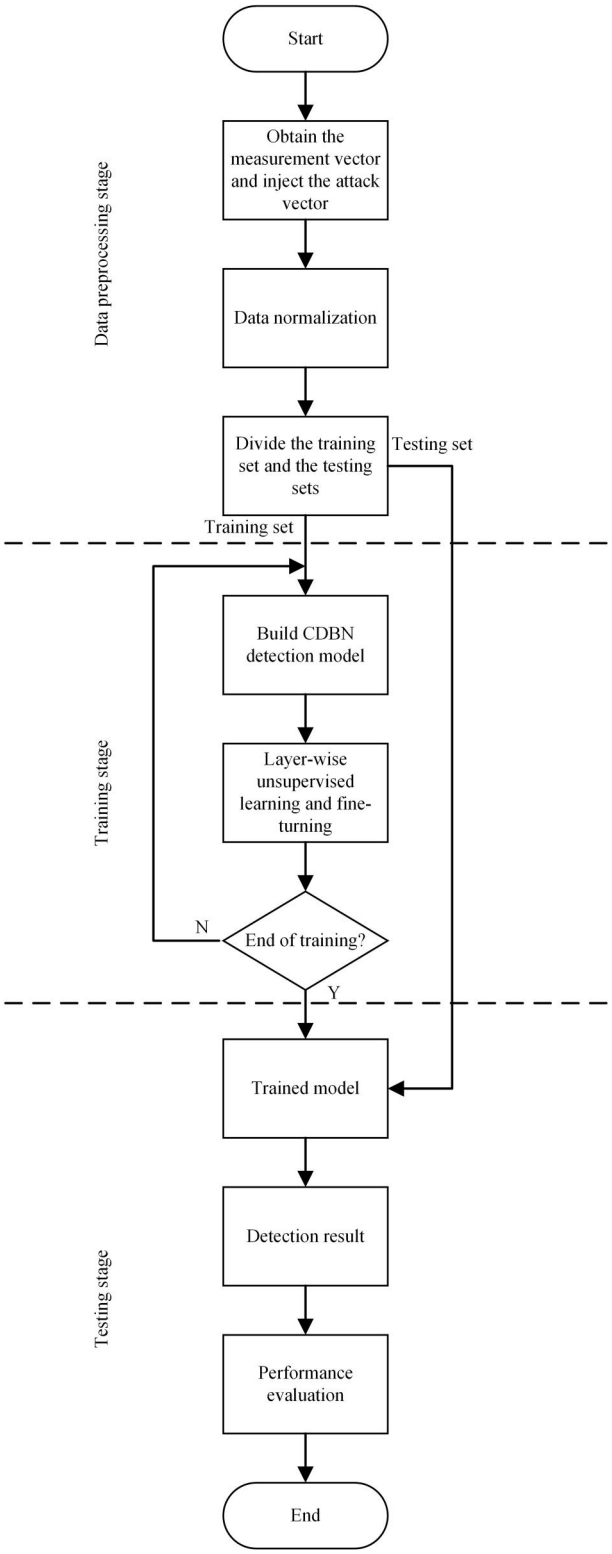

FIGURE 7 The Conditional Deep Belief Network based detection model flow chart

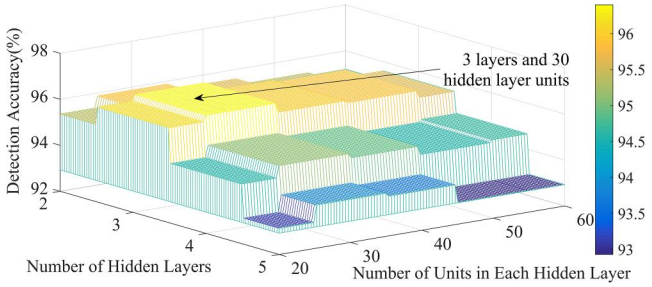

FIGURE 8 Accuracy of different hidden layers and different hidden layer units

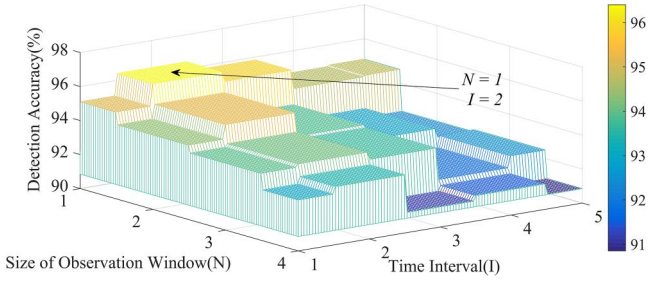

FIGURE 9 Accuracy of different N and I

window cannot adequately reflect the recent changes in the measurements. Similarly, a larger I tends to smooth out or even ignore some short-term but critical fluctuations, whereas a smaller I may cause this change to be too dramatic and lose its reference value [37]. According to the conclusion of the previous section, we build a CDBN structure with 3 hidden layers and 30 units in each layer. The range of N is set from 1 to 4, and I is increased from 1 to 5. We can see the simulation results in Figure 9. Considering the accuracy and the availability, $N = 1$ and $I = 2$ are a reasonable choice for detecting the FDIA.

5.2 | Multi-scenario verification

In this experiment, we discuss the accuracy of our developed scheme in the least-effort attack ($k = 1$) and multiple attacks ($k > 1$), respectively. According to Section 5.1, we simulate a three-layer CDBN model with 30 units per layer, set N to 1, and I to 2. Besides, by using the same data set, we compare the performance of our method with the SVM and the multilayer perceptrons (MLP), where the radial basis function (RBF) kernel is used in the SVM and the MLP consists of three hidden layers with 30 units. From Figure 10, we can see that the accuracy of our CDBN-based method is higher than the other two. Furthermore, with the increase of k, the detection performance is stable, and the accuracy can reach up to 98.4%. Specifically, when $k = 1, 4, 7, 10$, the confusion matrix and the receiver-operating characteristics (ROC) curves of the method are as shown in Table 1 and Figure 11, respectively [38]. It is one of the essential metrics for evaluating the performance of a classification model. As shown in Figure 11, the area under curve (AUC) is close to 1. It shows that the developed scheme has excellent performance and can accurately identify the FDIA.

In addition, the simulation is implemented on a 64-bit computer with the processor of 3.2 GHz clock speed. When the CDBN-based method is used for FDIA detection, the average detection time is only 1.12 ms, which can satisfy the requirement of real-time detection.

5.3 | Robustness verification

In the previous experiment, we set $\mathcal{N}(0, 0.25)$ as the environmental noise. But the real environment may be much worse. To evaluate the robustness, in this part, we fix the number of attacked measurements (k) to 4, and the standard deviation σ of environmental noise $\sim \mathcal{N}(0, \sigma)$ changes from 0.25 to 2.5. The settings of the other structural parameters are the same as Section 5.2. Figure 12 compares the accuracy obtained from the SVM, the MLP, and our developed scheme. Closer inspection of the figure shows that as the noise level increases, the accuracy of the three methods decreases. It is understandable because the higher the noise level, the harder it is to distinguish between normal and compromised measurements. However, the accuracy of the developed method is always the highest of the three. Especially when $\sigma < 2.0$, the accuracy can be more than 90%. That is to say, when the difference caused by the environmental noise is smaller than that caused by the FDIA, our CDBN-based method is competent and has good robustness.

5.4 | Scalability verification

In this section, we evaluate the scalability of the detection algorithm by using the IEEE 118-bus test system. In this test system, the environmental noise is set to $\mathcal{N}(0, 0.25)$, and the number of elements in each measurement is 360. We assume that the attacker can launch an FDIA on k values. After a large number of experiments, for the IEEE 118-bus test system, when we simulate a three-layer CDBN model with 400 units per layer, set N to 1, and I to 2, the CDBN-based method has the best performance. Figure 13 shows the performance comparison of the developed method with the SVM and the MLP, where the RBF kernel is used in the SVM and the MLP consists of three hidden layers with 400 units. It can be seen from the simulation results that as the attack intensity increases, the accuracy of the CDBN-based method is relatively stable and above 90%, while the accuracy of the SVM-based method is less than 75%. Moreover, because the MLP-based method does not consider the time correlation of the input data, the detection performance is not very good. The simulation results show that the developed scheme has strong scalability, is suitable for FDIA detection with different attack intensity, and can ensure high accuracy when dealing with complex, large-scale systems. In addition, the average detection time of FDIA for the IEEE 118-bus test system is 2.46 ms, which can realise real-time detection.

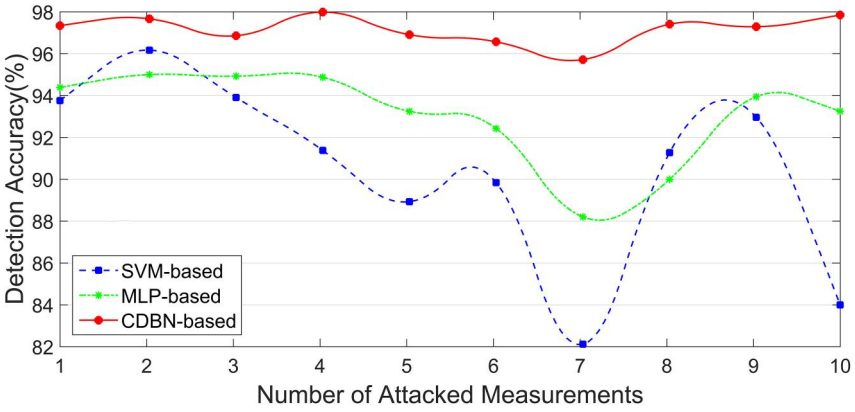

FIGURE 10 Accuracy of different number of attacked measurements

TABLE 1 Confusion matrix when $k = 1, 4, 7, 10$

Number of attacked measurements	Actual label	Identified to be normal	Identified to be compromised
1	Normal	4931	69
	Compromised	197	4803
4	Normal	4937	63
	Compromised	91	4909
7	Normal	4852	145
	Compromised	338	4664
10	Normal	4978	22
	Compromised	193	4807

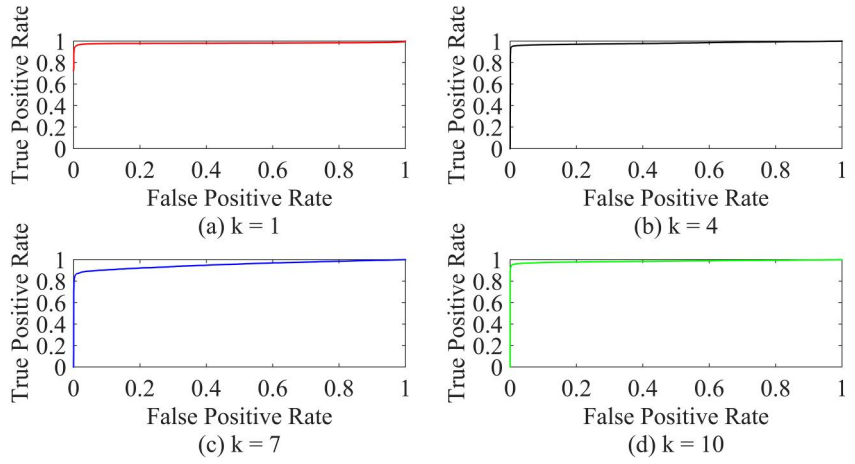

FIGURE 11 Receiver-operating characteristics curve when $k = 1, 4, 7, 10$

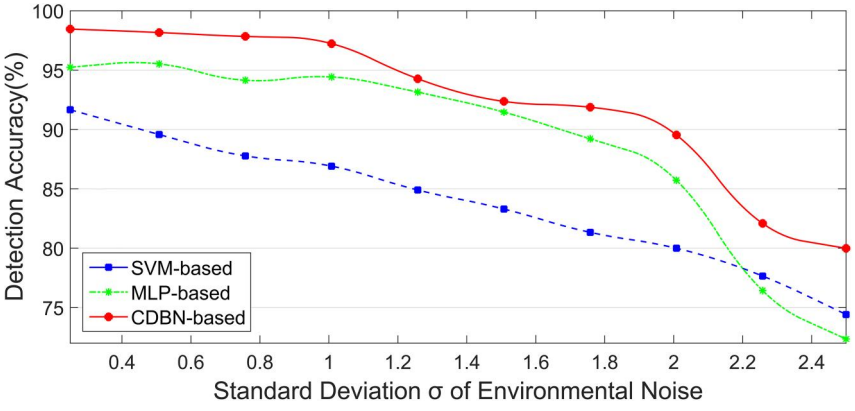

FIGURE 12 Accuracy of different standard deviation σ

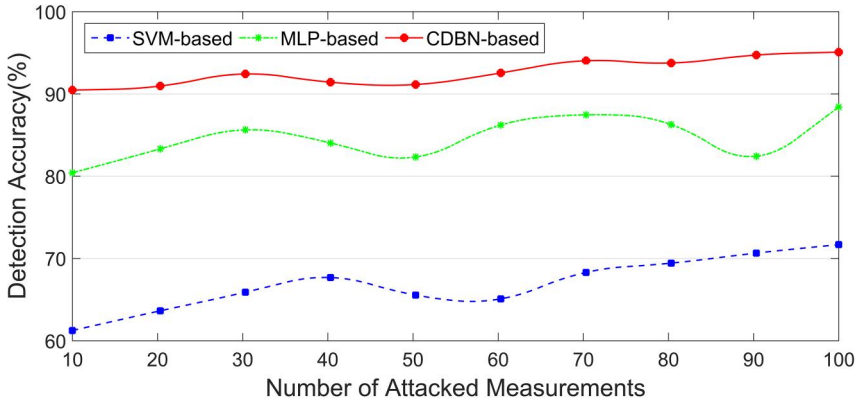

FIGURE 13 Accuracy of different number of attacked measurements in IEEE 118-bus power test system

6 | CONCLUSION

This article conducts an in-depth study of the state estimation, analyses the basic principles of the FDIA, and focusses on the detection of power system cyberattacks. By integrating the DBN structure with the CGBRBM, that can process time-series real-valued measurement data, we introduce a deep learning-based scheme to recognise the potential FDIA for maintaining the stability of the smart grid. It can extract the high-dimensional temporal behaviour features from the input data to construct a classification model and perform detection. In the simulation, we first optimise the model parameters suitable for the FDIA detection. By simulating two realistic attack scenarios, according to the determined optimal parameters, the performance is then demonstrated. The results indicate that our scheme can efficiently detect the FDIA and achieve better accuracy and robustness than the SVM and the MLP. In our future work, more sophisticated attack scenarios will be investigated based on the developed mechanism. Additionally, to be more widely used in the field of the FDIA detection, we will explore our scheme in the AC power system model.

ACKNOWLEDGEMENT

This research has been funded by the Project Research on Forecasting Method of Smart Grid Big Data Based on Random Projection Neural Networks (61703379) supported by the National Natural Science Foundation of China.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

ORCID

Yucheng Ding https://orcid.org/0000-0002-0737-9303

REFERENCES

1. Fang, X., et al.: Smart grid—the new and improved power grid: A survey. IEEE Commun. Surv. Tutorials. 14(4), 944–980 (2012)
2. Deng, R., et al.: A survey on demand response in smart grids: Mathematical models and approaches. IEEE Trans. Ind. Informatics. 11(3), 570–582 (2015)
3. Wu, F.F.: Power system state estimation: A survey. Int. J. Electr. Power Energy Syst. 12(2), 80–87 (1990)
4. Monticelli, A.: Electric power system state estimation. Proc. IEEE. 88(2), 262–282 (2000)
5. Exposito, A.G., Abur, A.: Operating states of a power system. In: Power System State Estimation: Theory and Implementation (vol. 1, pp. 1–6). CRC Press (2004)
6. Gungor, V.C., et al.: Smart grid technologies: communication technologies and standards. IEEE Trans. Ind. Informatics. 7(4), 529–539 (2011)
7. Deng, R., et al.: Residential energy consumption scheduling: a coupled-constraint game approach. IEEE Trans. Smart Grid. 5(3), 1340–1350 (2014)
8. Zhao, C., et al.: Consensus-based energy management in smart grid with transmission losses and directed communication. IEEE Trans. Smart Grid. 8(5), 2049–2061 (2017)
9. Baumeister, T.: Literature review on smart grid cyber security. Collaborative Software Development Laboratory at the University of Hawaii (2010)
10. Sorebo, G.N., Echols, M.C.: Smart grid risks. In: Smart Grid Security: An End-to-End View of Security in the New Electrical Grid (vol. 1, pp. 9–14). CRC Press (2016)
11. Wood, A.J., Wollenberg, B.F., Sheblé, G.B.: Power system security. In: Power Generation, Operation, and Control (vol. 7, 3rd ed., pp. 296–313). John Wiley & Sons (2013)
12. Sridhar, S., Hahn, A., Govindarasu, M.: Cyber–physical system security for the electric power grid. Proc. IEEE. 100(1), 210–224 (2012)
13. Teixeira, A., et al.: Cyber security analysis of state estimators in electric power systems. In: 49th IEEE Conference on Decision and Control (CDC), pp. 5991–5998. (2010)
14. Kosut, O., et al.: Malicious data attacks on the smart grid. IEEE Trans. Smart Grid. 2(4), 645–658 (2011)
15. Bobba, R.B., et al.: Detecting false data injection attacks on dc state estimation. In: Preprints of the First Workshop on Secure Control Systems, CPSWEEK (2010)
16. Dan, G., Sandberg, H.: Stealth attacks and protection schemes for state estimators in power systems. In: 2010 First IEEE International Conference on Smart Grid Communications, pp. 214–219. (2010)
17. Qingyu, Y., et al.: On false data-injection attacks against power system state estimation: Modeling and countermeasures. IEEE Trans. Parall. Distrib. Syst. 25(3), 717–729 (2014)
18. Liu, L., et al.: Detecting false data injection attacks on power grid by sparse optimization. IEEE Trans. Smart Grid. 5(2), 612–621 (2014)
19. Liu, Y., et al.: Research on efficient detection methods for false data injection in smart grid. In: 2014 International Conference on Wireless Communication and Sensor Network, pp. 188–192. (2014)
20. Hu, Z., et al.: False data injection attacks identification for smart grids. In: 2015 Third International Conference on Technological Advances

in Electrical, Electronics and Computer Engineering (TAEECE), pp. 139–143. (2015)
21. Hinton, G.E., Osindero, S., Teh, Y.-W.: A fast learning algorithm for deep belief nets. Neural Comput. 18(7), 1527–1554 (2006)
22. Sukhbaatar, S., et al.: Robust generation of dynamical patterns in human motion by a deep belief nets. In: Asian Conference on Machine Learning, pp. 231–246. (2011)
23. Fischer, A., Igel, C.: An introduction to restricted Boltzmann machines. In: Iberoamerican Congress on Pattern Recognition, pp. 14–36. (2012)
24. Bengio, Y., et al.: Greedy layer-wise training of deep networks. In: Advances in Neural Information Processing Systems, pp. 153–160. (2007)
25. Bengio, Y.: Learning deep architectures for AI. Foundations Trends® Mach. Learn. 2(1), 1–127 (2009)
26. Lee, H., et al.: Unsupervised learning of hierarchical representations with convolutional deep belief networks. Commun. ACM. 54(10), 95–103 (2011)
27. Mnih, V., Hinton, G.E.: Learning to label aerial images from noisy data. In: Proceedings of the 29th International Conference on Machine Learning (ICML-12), pp. 567–574. (2012)
28. Yan, Y., et al.: Learning document semantic representation with hybrid deep belief network. Comput. Intell. Neurosci. 2015, 28 (2015)
29. Chen, X.-W., Lin, X.: Big data deep learning: Challenges and perspectives. IEEE Access. 2, 514–525 (2014)
30. Taylor, G.W., Hinton, G.E., Roweis, S.T.: Modeling human motion using binary latent variables. In: Advances in Neural Information Processing Systems, pp. 1345–1352. (2007)
31. Taylor, G.W.: Composable, distributed-state models for high-dimensional time series, Ph.D. dissertation. Department of Computer Science, University of Toronto, Toronto (2009)
32. Wei, J., Mendis, G.J.: A deep learning-based cyber-physical strategy to mitigate false data injection attack in smart grids. In: 2016 Joint Workshop on Cyber-Physical Security and Resilience in Smart Grids (CPSR-SG), pp. 1–6. (2016)
33. Liu, Y., Ning, P., Reiter, M.K.: False data injection attacks against state estimation in electric power grids. ACM Trans. Inf. Syst. Secur. 14(1), 13 (2011)
34. Hinton, G.E.: Training products of experts by minimizing contrastive divergence. Neural Comput. 14(8), 1771–1800 (2002)
35. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature. 323(6088), 533–536 (1986)
36. Ke, J., Liu, X.: Empirical analysis of optimal hidden neurons in neural network modeling for stock prediction. In: 2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application, pp. 828–832. (2008)
37. Deypir, M., Sadreddini, M.H., Hashemi, S.: Towards a variable size sliding window model for frequent itemset mining over data streams. Comput. Ind. Eng. 63(1), 161–172 (2012)
38. Gönen, M.: Single continuous predictor. In: Analyzing Receiver Operating Characteristic Curves with SAS, vol. 3, pp. 15–36. SAS Institute (2007)

How to cite this article: Ding, Y., et al.: A deep learning-based classification scheme for cyber-attack detection in power system. IET Energy Syst. Integr. 3(3), 274–284 (2021). https://doi.org/10.1049/esi2.12034