Semi-supervised False Data Injection Attacks Detection in Smart Grid

Yasheng Zhou[1], Li Yang[1(✉)], and Yang Cao[2]

[1] School of Computer Science and Technology, Xidian University, Xi'an 710071, China
zhousheng396@163.com

[2] Guizhou Vocational College of Electronic Science and Technology, Guian 550025, China

Abstract. False data injection attacks (FDIAs) detection in smart grid, requires adequate labeled training samples to train a detection model. Due to the strong subjectivity, relying on expert knowledge and time-consuming nature of power system sample annotation, this task is intrinsically a small sample learning problem. In this paper, we propose a novel semi-supervised detection algorithm for FDIAs detection. The semi-supervised label propagation algorithm can dynamically propagate the label from labeled samples to unlabeled samples, automatically assign class labels to the unlabeled samples dataset, and enlarge the labeled samples dataset. Jointly use a small number of manually labeled samples dataset and a large number of auto-labeled samples dataset to construct a classifier via semi-supervised learning. Comparing the proposed algorithm with supervised learning algorithms, the results suggest that, with the scheme of semi-supervised learning from large unlabeled dataset, the proposed algorithm can significantly improve the accuracy of false data injection attacks detection.

Keywords: False data injection attacks · Semi-supervised learning · Label propagation · Small sample learning

1 Introduction

In recent years, with the improvement of the intelligent level of the power grid, the amount of communication data has increased exponentially, and the attack on the power monitoring system has been in various forms and the attack surface has been further expanded [1]. The integrity destruction and precise tampering of the measurement data of the supervisory control and data acquisition (SCADA) system can effectively affect the normal operation of the system and the final decision of the on-site personnel. Among the attacks on the power grid, the false data injection attack is a typical attack method that damages the integrity of the power grid information by tampering with the power measurement data [2]. It uses the loopholes in the state estimation of the power monitoring system to inject carefully designed false data into the data collection terminal, bypassing the bad data detection module in the state estimation, causing the dispatcher to misjudge the current power grid state, which can pose a great threat to the stability of the power system [3].

© ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2022
Published by Springer Nature Switzerland AG 2022. All Rights Reserved
J. Lin and Q. Tang (Eds.): AC3 2022, LNICST 448, pp. 189–200, 2022.
https://doi.org/10.1007/978-3-031-17081-2_12

The identification of false data is to distinguish false data from normal data [4], which is essentially a classification problem [5]. With the continuous development of artificial intelligence technology, machine learning algorithms are more and more widely used to solve such problems due to their efficient modeling and learning capabilities. Liu et al. first proposed the concept of false data injection attacks in the power grid, and implemented the false data injection attack in the DC environment when part of the power system topology information and all the topology information are known, respectively [6]. Yu et al. proposed blind data injection attack, that is, when the system topology information is not clear and the Jacobian matrix is not considered, the principle component analysis method is used to implement false data injection attack [7]. Luo et al. proposed a method for detecting and isolating false data based on unknown input observers for residual characteristics [8]. Wang et al. used three traditional machine learning methods: state perceptron method, K-Nearest Neighbor (KNN) method, and Support Vector Machine (SVM) method to detect false data injection attack [9]. Liu et al. proposed information fuzzy reasoning algorithm to detect exceptional event by fusing physical information [10].

Although, machine learning algorithms have achieved many successful applications in grid false data injection attack detection and anomaly detection [11, 12]. However, most of the machine learning algorithms used in FDIAs detection are mainly based on supervised learning, in which only labeled samples can be used for model training and unlabeled samples cannot be used. The model performance depends on the quantity and quality of labeled samples when using supervised learning algorithms. Hence training a classification model by using supervised learning technique requires a large number of labeled samples. On the one hand, the types of labeled samples need to be able to cover the complete actual situation, and the number of labeled samples should be large enough to ensure that the sample features learned by the model can be strengthened. On the other hand, the labeled samples should be authoritative and accurate. However, in the actual operation process of the SCADA system, it is extremely difficult to obtain a large number of accurately labeled samples. The labeled power samples rely on the knowledge of power experts, and the artificially constructed labeled samples are highly subjective, with the result that it is time-consuming and laborious to construct a large number of labeled samples dataset by manual labeling. Therefore, there are usually only a very small number of labeled samples and a large number of unlabeled samples can be used in FDIAs detection in the power grid, which leads to low accuracy of most existing detection methods, and makes FDIAs detection to be intrinsically a small sample learning problem.

To address this issue, semi-supervised learning technique, which is capable of simultaneously using labelled and unlabeled training samples, should be used to construct the detection model. This paper proposes a novel semi-supervised-based false data injection attack detection (SS_FDIA) algorithm for FDIAs detection. We adopt the label propagation algorithm based on semi-supervised learning technique to automatically assign a reasonable class label to the large number of unlabeled samples dataset, when only a small number of labeled samples are used. Then we jointly use both a small number of manually labeled samples dataset and a large number of auto-labeled samples dataset to co-train a semi-supervised classifier. Finally, we evaluate the proposed SS_FDIA

algorithm against the two state-of-the-art supervised learning methods: the decision tree (DT) method and the random forest (RF) method in the same false data injection attacks dataset.

2 False Data Injection Attacks

In the actual operating environment of power system, the sources of bad data include not only data abnormalities caused by power equipment trouble, but also false data maliciously constructed by attackers. Attackers can upload false data by attacking terminal device or implementing man-in-the-middle attacks on communication links. For attackers, it is very essential to design the optimal false data vector and construct the false data that can bypass the bad data detection module of power system based on the existing attack resources and system topology knowledge against the identification method of the state estimation module.

In power system, the relationship between the measurement value and the true state value is defined as:

$$z = Hx + v \tag{1}$$

where $z = (z_1, z_2, \cdots, z_m)^T$ is the m-dimensional measurement vector, $x = (x_1, x_2, \cdots, x_n)^T$ is the n-dimensional true state vector, H is a $m \times n$ Jacobian matrix, and v is noise. False data injection attacks are implemented by modifying the measurement vector, which constructs a malicious measurement vector a, shown as follows:

$$z_a = z + a \tag{2}$$

if z_a can pass the bad data detection module of power system, it indicates that the false data injection attack is successful. When the attacker knows the Jacobian matrix H, if the maliciously constructed vector a in Eq. (2) is a linear combination of the column vectors of the Jacobian matrix H, shown as follows:

$$\mathrm{a} = \mathrm{Hc} \tag{3}$$

where $c = (c_1, c_2, \cdots, c_n)^T$ is an any non-zero n-dimensional vector, the measurement vector z_a after which is injected the malicious measurement vector a can bypass the residual-based bad data detection module of the power system. Since the malicious measurement vector a has an infinite number of solutions, it is possible to construct successful false data and realize effective tampering of power monitoring data.

3 Method

3.1 Overview

The proposed SS_FDIA consists of four major steps, including (1) constructing samples dataset which contains a small number of labeled samples and a large number of unlabeled samples (2) baseline classifier construction (3) Unlabeled samples labeling based on label propagation, and (4) semi-supervised re-training of the classifier. The detailed diagram of proposed SS_FDIA algorithm for FDIAs detection is shown in Fig. 1.

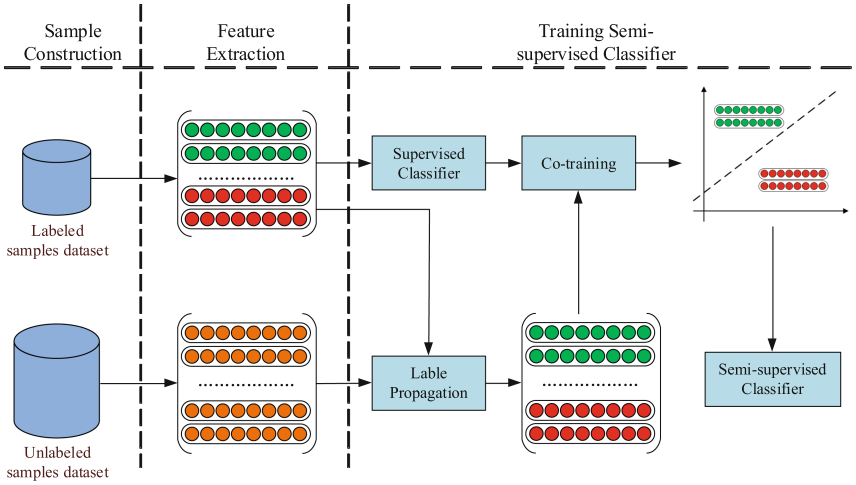

Fig. 1. Diagram of proposed SS_FDIA algorithm for FDIAs detection

3.2 Samples Dataset Construction

For false data injection attacks detection, we need simulate the grid environment. In this paper, we use IEEE-30 bus system for algorithm analysis, as shown in Fig. 2. We simulate the IEEE-30 bus system by using MatPower to construct the required samples dataset. It is generally acknowledged that there exists noises in the real power system and the measurement error of the SCADA system is normally distributed with the standard deviation σ about 0.5% to 2% of the normal measurement range. Therefore, in the experiments, the real value obtained in MatPower is randomly added with Gaussian noise error to construct the required dataset.

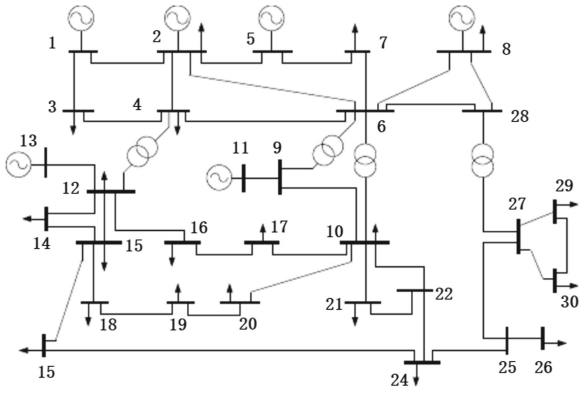

Fig. 2. IEEE-30 bus system

The samples dataset constructed for experiments are divided into normal samples and attacked samples. To construct normal samples, we take the measurement data on

the IEEE-30 bus system at different times and randomly add Gaussian noise error to it, with the standard deviation σ within 1%. These samples are labeled as "normal". To construct attacked samples, we take vector $\{a|a = Hc, c \neq 0\}$ in Eq. (3) as attack vector to build attacked samples labeled as "attacked". In this study, the modulus of the vector $\|c\|$ is set to a small value, which exactly meet the formula $\|c\| \leq 0.001$ in the experiments. Therefore, these normal samples and attacked samples we constructed for study have close data distribution, which result in that false data injection attacks have higher concealment and are more difficult to detect.

3.3 Baseline Classifier Construction

We use a small number of labeled samples to train a supervised classifier. In this paper, we train a DT (Decision Tree) classifier and a RF (Random Forest, random forest) classifier as the baseline classifiers. The baseline classifiers trained by using small number of labeled samples have poor classification performance. Thus, in the next stage, we adopt the semi-supervised learning method to retrain these classifiers.

3.4 Label Propagation and Semi-Supervised Classifier

To expand the training labeled samples dataset, we adopt the label propagation algorithm to automatically propagate the category labels from the small number of labeled samples dataset to the large number of unlabeled samples dataset according to the similarity between each sample [13].

Let the samples dataset, which contains both labeled samples and unlabeled samples, be denoted by $X = \{x_1, x_2, \cdots, x_N\}$ and the corresponding labels set is denoted by $Y = \{y_1, y_2, \cdots, y_N\}$, where $x_i \in R^D$ represents the i-th sample's D-dimensional feature, and $y_i \in \{\lambda_1, \lambda_2\}$ represents the corresponding class label. Let the number of labeled samples be denoted by l and the number of unlabeled samples be denoted by u. Therefore, $N = l + u$. The rank of sample x_i relative to x is given by

$$\rho(x_i, x_j) = \left|\left\{x \in X \,|d(x_i, x) < d(x_i, x_j)\right\}\right| \tag{4}$$

where $d(x_i, x_j)$ represents the distance between x_i and x_j. Let $\tau_d(x_i)$ and τ_ρ be positive threshold values for item distances and ranks, respectively. The region of influence of sample x_i is defined as the set of samples simultaneously falling within distance $\tau_d(x_i)$ of x_i and rank τ_ρ of x_i, shown as follows

$$Infl(x_i) = \{x \in X \,|d(x_i, x) \leq \tau_d(x_i) \cap \rho(x_i, x) \leq \tau\} \tag{5}$$

sample x_i can influence sample x_j, if $x_j \in Infl(x_i)$ which means that x_j lies within the region of influence associated with x_i [14].

There are five major steps in the process of label propagation from labeled samples to unlabeled samples [15], as shown in Table 1.

(1) Initialize an influence graph G respect to the neighborhood relationships of items in X. Construct the influence graph $G(V_l \cup V_u, E)$ according to the sample set X,

Table 1. Label Propagation Algorithm.

Label Propagation algorithm
Input**:** Samples set X and the corresponding label set Y
Output**:** Score matrix S
1. $N \leftarrow \lvert X \rvert$ 1., $t \leftarrow 2$
2.Construct an influence graph G respect to the neighborhoodrelationships of items in X;
3.Compute the $N \times N$ adjacency matrix A of G;
4.Compute the $N \times N$ propagation matrix P from A;
5.Initialize the $N \times t$ score matrix S;
6.**repeat**
7.$S^{'} \leftarrow S$
8.$S \leftarrow PS^{'}$
9.**until** $S \neq S^{'}$

where V_l represents the small number of labeled samples dataset, and V_u represents the large number of unlabeled samples dataset. The edge set E is composed of two types of edges: strong edges that connect two mutually influenced samples and weak edges that connect two singly influenced samples.

(2) Compute the N × N adjacency matrix A of G. Entries of A can be computed by:

$$a_{i,j} = \begin{cases} \alpha \cdot \mathrm{sim}(x_i, x_j), \ if \ \langle j, i \rangle \text{is a strong edge} \\ \mathrm{sim}(x_i, x_j), \ if \ \langle j, i \rangle \text{is a weak edge} \\ 0, \text{otherwise} \end{cases} \tag{6}$$

where $\alpha \geq 1$ is an amplifying factor that favors strong edges, and $\mathrm{sim}(x_i, x_j)$ denotes the similarity value between two samples, shown as follows

$$sim(x_i, x_j) = 1 - \frac{d(x_i, x_j) - d_{min}}{d_{max} - d_{min}} \tag{7}$$

where d_{min} and d_{max} are the minimum and maximum pairwise distance between different samples in the graph, respectively.

(3) Compute the N × N propagation matrix P from A.Entries of P can be computed by

$$p_{i,j} = \begin{cases} a_{i,j}, if \ node \ i \in V_l \\ \beta \cdot \frac{a_{i,j}}{\sum_{q=1}^{N_T} a_{i,q}}, \ otherwise \end{cases} \tag{8}$$

where $\beta \in (0, 1)$ is a damping factor used to penalize nodes that are far away from source nodes, and to accelerate the convergence.

(4) Initialize the N × t score matrix S.Entries of the N × t initial score matrix $S^{(0)}$ can be computed as

$$s_{i,j}^{(0)} = \begin{cases} 1, \text{ if } x_i \ is \ associated \ with \ \lambda_j \\ 0, \quad otherwise \end{cases} \tag{9}$$

(5) Iteratively update score matrix S.$S^{(q)}$ is computed from the previous state $S^{(q-1)}$ according to the formula

$$S^{(q)} = PS^{(q-1)} \tag{10}$$

until the change of each element of S in two successive iterations is lower than the tolerance ε.

Once score matrix S converges, we assign each unlabeled sample a class label according to the score matrix S. Thus, we can jointly use both a small number of manually labeled samples dataset and a large number of auto-labeled samples dataset, to co-train a classifier in a semi-supervised learning way through label propagation method.

4 Results

4.1 Dataset

The entire samples dataset we construct includes 400 labeled samples (which can be divided into 200 normal samples and 200 attacked samples) and 380 unlabeled samples. For the sake of verifying the superiority of SS_FIDA algorithm in small sample learning problem, the 400 labeled samples are divided into a training dataset of 20 samples and a test dataset of 380 samples. The samples distribution is shown in Table 2.

Table 2. Samples distribution

Training dataset		Testing dataset
Labeled samples	Unlabeled samples	Labeled samples
20	380	380

4.2 Experimental Results

We performed the experiments 10 times and used the average classification accuracy as the evaluation criteria of the classification performance of each algorithm. We evaluated the proposed semi-supervised SS_FDIA algorithm against the two state-of-the-art supervised learning methods: DT (Decision Tree) method and RF (Random Forest) method. These classifiers were trained by using a training dataset of only 20 labeled samples and the classification performance were depicted in Fig. 3. The average classification accuracy of DT method is 76.71% and the accuracy of RF method is 81.26%. The average classification accuracy of SS_FDIA(DT) classifier, which is co-trained by jointly using SS_FDIA algorithm and DT method, is 88.68%. The average classification accuracy of SS_FDIA(RF) classifier, which is co-trained by jointly using SS_FDIA algorithm and RF method, is 99.0%.

It reveals that the proposed SS_FDIA algorithm outperforms the state-of-the-art DT method and RF method. The semi-supervised SS_FDIA algorithm learning from

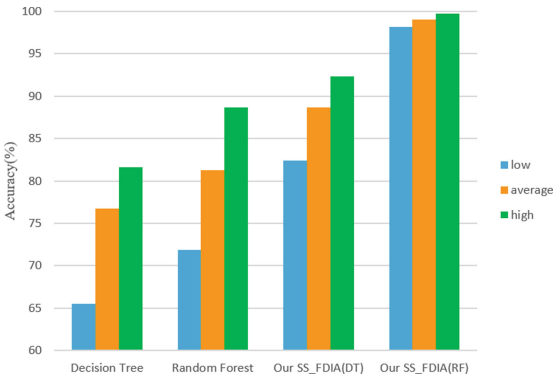

Fig. 3. Classification accuracy of different algorithms

unlabeled samples by using label propagation mechanism can substantially improve the classification accuracy. Compared with DT method and RF method, the classification accuracy of SS_FDIA algorithm is improved by 11.91% and 17.74%, respectively. It demonstrates that we can improve the performance of false data injection attacks detection by learning from unlabeled samples, though these samples may contain a lot of uncertainty.

5 Discussion

5.1 Amount of Unlabeled Samples

Generally, the accuracy of classifiers improves with the increase of labeled training samples. However, too many unlabeled training samples may deteriorate the classification performance in semi-supervised learning [16, 17]. Therefore, we performed FDIAs detection tasks to determine how many unlabeled samples should be used to facilitate training. Figure 4 shows the classification performance of SS_FDIA algorithm when using different numbers of unlabeled samples. The SS_FDIA(RF) classifier co-trained by the SS_FDIA algorithm and RF method achieves high classification accuracy when using 110 to 170 unlabeled samples and the classification accuracy can reach up to 99.74% when using 170 unlabeled samples. When using 190 ~ 310 unlabeled samples, the SS_FDIA(DT) classifier co-trained by the SS_FDIA algorithm and DT method achieves high classification accuracy and the classification accuracy can reach up to 92.63% when using 270 unlabeled samples. Later, when more unlabeled samples are used, the computational complexity will increase but the classification accuracy will not further improve.

Experimental results suggest that SS_FDIA can utilize unlabeled samples to improve the performance of the classification model to some extent to solve the small sample learning problem. As the number of unlabeled samples increases, the classification accuracy first increases, and after reaching a certain level, it will not continue to grow.

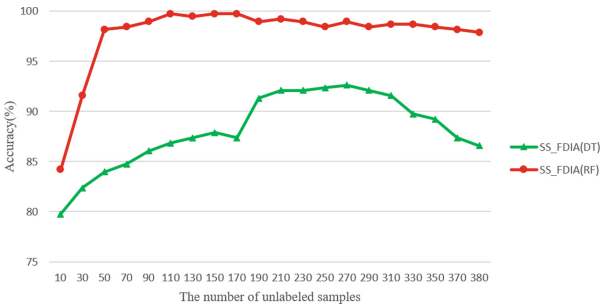

Fig. 4. Classification accuracy curve of the SS_FDIA algorithm when learning different numbers of unlabeled samples

5.2 Robustness of Parameters

In this paper, the proposed SS_FDIA algorithm can dynamically propagate the label information from 20 labeled samples dataset to large number of unlabeled samples dataset and automatically assign class labels to unlabeled samples in conformity with the similarity between each sample. The label propagation process involves two important parameters α and β, where $\alpha \geq 1$ and $\beta \in (0, 1)$. We ran experiments 10 times and calculated the average classification accuracy to investigate the impact of different parameter α and β on label propagation algorithm performance. The classification accuracy curve of parameter α and β are shown in Fig. 5 and Fig. 6. The label propagation algorithm achieves high classification accuracy when the value of α is between [2, 5] and the value of β is between [0.3, 0.7]. Therefore, in this study, we adopt the default parameters in the label propagation algorithm as $\alpha = 5$ and $\beta = 0.5$.

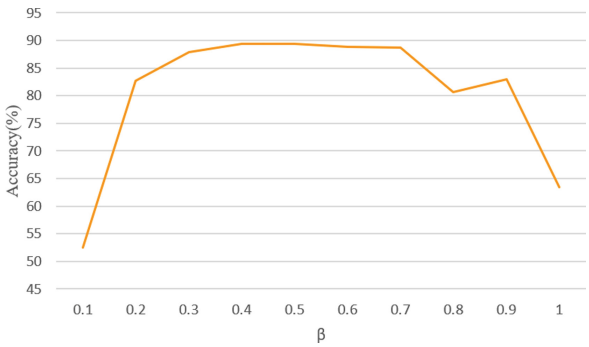

Fig. 5. Classification accuracy curve of parameter β

It reveals that SS_FDIA algorithm has strong robustness to parameters and strong generalization capability, which can achieve good experimental results with the parameters within a large range and easily converge to the optimal solution.

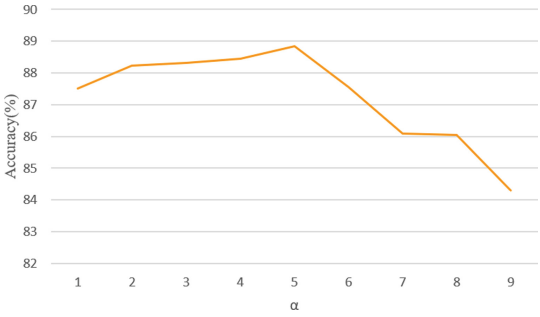

Fig. 6. Classification accuracy curve of parameter α

5.3 Computational Complexity

Table 3 shows the time cost of each stage of the random forest (RF) algorithm, decision tree (DT) algorithm and SS_FDIA algorithm. Applying the proposed SS_FDIA algorithm to FDIAs detection consists of two phrases: offline training and online testing. Since the SS_FDIA algorithm need to propagate class labels from the labeled samples to the unlabeled samples in the training phrase to automatically expand the labeled training samples dataset, and then use the labeled samples and unlabeled samples to retrain a semi-supervised classifier, the training phrase is quite time-consuming. Nevertheless, the SS_FDIA algorithm is very efficient in the testing phase, costing around 0.0006 s to detect each sample. It indicates that compared to traditional supervised learning methods, the proposed SS_FDIA algorithm can be better applied to real-time FDIAs detection in power system.

Table 3. Time complexity of different algorithms

Steps methods	Offline training				Testing each sample
	Feature Extraction	Training supervised classifier	Label propagation	Training semi-supervised classifier	
RF/DT	0.0001	0.00001	N/A	N/A	0.0005
SS_FDIA	0.0001	0.00001	92.8838	0.0087	0.0006

5.4 Amount of Labeled Samples

Table 4 gives the number of labeled training samples required by different algorithms: Wang [18], Lu [19], Xue [20] and proposed SS_FDIA algorithm. For false data injection attacks detection, the accuracy of the classification model depends on the number of labeled training samples. It is a very arduous task to train a high-performance classification model without large number of labeled training samples. Nevertheless, it's difficult

to gather labeled attack samples in the real power grid and man-made samples are lack of objectivity. Hence, the results produced by most existing solutions are less accurate for FDIAs detection. Compared with other algorithms, the proposed SS_FDIA algorithm has achieved better classification performance when using a very small number (20) of labeled samples.

It reveals that the proposed SS_FDIA algorithm has a sense of superiority in solving the small sample learning problem for FDIAs detection in power system.

Table 4. Number of labeled training samples required by different algorithms.

Methods	Wang	Lu	Xue	Our SS_FDIA
Number of samples	30000	24192	3003	20

6 Conclusion

In this paper, we propose a novel FDIAs detection algorithm by using a large number of unlabeled samples in a semi-supervised way to alleviate the difficulties caused by the lack of high-quality adequate labeled samples, which can be ascribed to the time-consuming nature of attacked samples manual annotation. To expand the training samples dataset required by model learning, we apply semi-supervised label propagation method to automatically assign class labels for large unlabeled samples. Then, we jointly use a small number of manually labeled samples dataset and a large number of auto-labeled samples dataset to co-train a classifier via semi-supervised learning for FDIAs detection. Our results suggest that the proposed SS_FDIA algorithm can improve the accuracy of FDIAs detection by using large unlabeled training sample dataset in an appropriately designed semi-supervised learning way, which is of great significance for solving the small sample learning problem in FDIAs detection.

References

1. Zhu, H., Zhao, L., Qin, K., et al.: Active protection stratery of power monitoring network security based on big data analysis. Electr. Measur. Instrum. **57**(21), 133–139 (2020)
2. Zhang, Z., Deng, R., Cheng, P., et al.: On feasibility of coordinated time-delay and false data injection attacks on cyber-physical systems. IEEE Internet Things J. (2021)
3. Ahmed, M., Pathan, A.S.K.: False data injection attack (FDIA): an overview and new metrics for fair evaluation of its countermeasure. Complex Adapt. Syst. Model. **8**(1), 1–14 (2020)
4. Yin, X., Zhu, Y., Hu, J.: A subgrid-oriented privacy-preserving microservice framework based on deep neural network for false data injection attack detection in smart grids. IEEE Trans. Industr. Inf. **18**(3), 1957–1967 (2021)
5. Cao, J., Wang, D., Qu, Z., et al.: A novel false data injection attack detection model of the cyber-physical power system. IEEE Access **8**, 95109–95125 (2020)
6. Liu, Y., Ning, P., Reiter, M.K.: False data injection attacks against state estimation in electric power grids. ACM Trans. Inf. Syst. Secur. (TISSEC) **14**(1), 1–33 (2011)

7. Yu, Z.H., Chin, W.L.: Blind false data injection attack using PCA approximation method in smart grid. IEEE Trans. Smart Grid **6**(3), 1219–1226 (2015)
8. Luo, X., Wang, X., Pan, X., et al.: Detection and isolation of false data injection attack for smart grids via unknown input observers. Gener. Transm. Distrib. IET **13**(8), 1277–1286 (2019)
9. Wang, B., Zhao, Y., Zhang, S., et al.: Study of monitoring false data injection attacks based on machine-learning in electric systems. J. Electron. Inf. Sci. **2**(2), 122–128 (2017)
10. Liu, Y., Fang, Y., Sun, H., et al.: Cyber-physical fuzzy inference based attack detection method in smart grid. China Sci. Paper **11**(14), 1619–1625 (2016)
11. Zheng, S., Liang, Q., Peng, X., et al.: Research on abnormal power consumption behavior identification based on fuzzy clustering. Electr. Measur. Instrum. **57**(19), 40–44 (2020)
12. Zhou, Z., Chen, Q., Ma, B., et al.: An improved YOLO target detection method with its application in cable device abnormal condition recognition. Electr. Measur. Instrum. **57**(02), 14–20 (2020)
13. Hong, D., Naoto, Y., Jocelyn, C., et al.: Learning to propagate labels on graphs: An iterative multitask regression framework for semi-supervised hyperspectral dimensionality reduction. ISPRS J. Photogrammetry Remote Sens.: Official Publ. Int. Soc. Photogrammetry Remote Sens. (ISPRS) **158**, 35–49 (2019)
14. Li, N., Xia, Y.: Affective image classification via semi-supervised learning from web images. Multimedia Tools Appl. **77**(23), 30633–30650 (2018). https://doi.org/10.1007/s11042-018-6131-1
15. Houle, M.E., Oria, V., Satoh, S., et al.: Annotation propagation in image databases using similarity graphs. ACM Trans. Multimedia Comput. Commun. Appl. **10**(1), 1–21 (2013)
16. Dai, D., Van Gool, L.: Ensemble projection for semi-supervised image classification. IEEE Int. Conf. Comput. Vis. (ICCV) **2013**, 2072–2079 (2013)
17. Li, Y.F., Zhou, Z.H.: Towards making unlabeled data never hurt. IEEE Trans. Pattern Anal. Mach. Intell. **37**(1), 175–188 (2015)
18. Wang, G.: Research on detection method of power system false data injection attack based on machine learning. Northeast Electric Power University (2019)
19. Lu, J.: Research on false data attack detection of smart grid based on machine learning. North China Electric Power University (2019)
20. Xue, D.: Detecting false data injection attacks in smart grid. Chongqing University of Posts and Telecommunications (2019)