Adversarial Machine Learning Against False Data Injection Attack Detection for Smart Grid Demand Response

Zhang Guihai, *Graduate Student Member, IEEE,* and Biplab Sikdar, *Senior Member, IEEE*

***Abstract*—Distributed demand response (DR) is used in smart grids to allow utilities to balance the power supply with the demand by modulating the consumer's behavior by varying the price according to consumption patterns and forecasts. False data injection (FDI) attacks of DR can cause large economical losses for utilities, equipment damage, and issues with power flows. Recently, FDI attack detection methods based on deep learning models have been proposed and these methods have better detection performance as compared to traditional approaches. However, deep learning based models may be vulnerable to adversarial machine learning (AML) attacks. In this paper, we demonstrate the vulnerability of state-of-the-art deep learning based FDI attack detectors in DR scenarios to AML attacks. We propose a new black-box FDI attack framework to fabricate power demands in distributed DR scenarios that is capable of deceiving deep learning based FDI attack detection. The evaluation results show that the proposed AML framework can significantly decrease the FDI detection models accuracy and outperforms other AML techniques proposed in literature.**

***Index Terms*—Smart grid, demand response, false data injection, deep learning.**

I. Introduction

With the rapid development and integration of information and communication techniques, traditional power grids have transformed into smart grids. Smart grids contain heterogeneous components such as sensors, meters, and actuators to make it more intelligent, efficient, and distributed. To achieve cost effective power supply in smart grids, demand response (DR) schemes have been proposed and developed. In such schemes, utility companies drive consumer to change their power usage based on the supply by providing them with financial considerations, using schemes such as Day Ahead Pricing, Real-Time Pricing and Time of Use rates [1]. DR schemes also help users to have more control on decisions related to their power consumption and to reduce the electricity bill by decreasing usage during peak times. Moreover, with the reduced total peak demands, utility companies can maintain and operate the grid system with lower costs and utilize the grid better [2].

While DR has many advantages, its reliance on communication with smart meter makes it vulnerable to cyber attacks. For example, attackers can launch false data injection (FDI) attacks on the DR by injecting data, compromising transmission signals, or hacking meter devices to cause incorrect power flows, physical damage to the grid, insufficient generation, or to gain monetary benefits [3], [4]. Therefore, FDI detection is a necessary component for reliable DR schemes. In recent years, researches have shown that deep learning (DL) techniques may be used to detect FDI attacks in smart grids [5]–[9]. However, studies have shown that DL models are vulnerable to adversarial machine learning (AML) [10]. In AML attacks, perturbations are added to the input data of the DL model to deceive it into making false predictions.

The authors are with the National University of Singapore, Singapore.

This paper explores AML against DL based FDI detection methods, especially in distributed DR schemes. The objective of the paper is to critically evaluate such attacks and demonstrate the vulnerabilities of existing systems. In the overall scenario considered in this paper, the adversary performs FDI attacks by modifying the power consumption data and defenders use DL-based models to detect the FDI attacks. We propose a Simple Iterative FDI Attack framework that is capable for breaching state-of-the-art DL based FDI attacks and highlight the vulnerabilities of existing FDI detection techniques. The main contributions of this paper are as follows:

- We show the vulnerabilities of state-of-the-art DL-based models for FDI attack detection in distributed DR schemes, where attackers are able to bypass the detection mechanism.
- We propose the Simple Iterative FDI Attack framework which is a black-box method to modify the power consumption data which can bypass state-of-the-art DL-based FDI attack detection mechanisms with high probability.
- We analyze the performance of well-known AML algorithms against distributed DR schemes using both white-box and black-box methods.

The rest of the paper is organized as follows. Section II describes the related work on distributed DR schemes, DL models, and AML attack. Section III shows the details of attack models and presents the proposed Simple Iterative FDI Attack framework. Section IV presents the simulation results and finally, Section V concludes the paper.

II. Related Work

A. Distributed Demand Response Schemes

Distributed DR schemes are aimed at increasing the effectiveness and efficiency of power grids by helping to maintain the balance between supply and demand. DR mechanisms achieve this balance by using time-varying pricing to change the power consumption patterns of users. Since manual response to dynamically changing price incentives is inconvenient for users, distributed DR schemes implement a combination of home

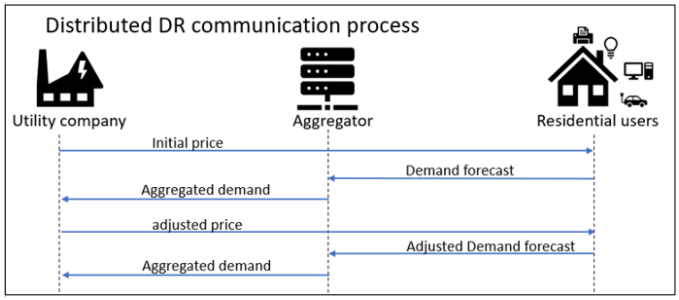

Fig. 1. Distributed DR communication sequence.

energy management system (HEMS) and advanced metering infrastructure (AMI) to assist users. The HEMS automatically performs the optimization process for users to decide the best schedule of electricity consumption, while satisfying the power requirement and lowering the overall cost [11], [12]. Within the distributed DR scheme, individual users have their own cost function to minimize their bill or to maximize the power consumption. Due to privacy concerns related to collection of individual demands [13], utility companies usually do not get the power demand directly. Instead, a trusted and independent aggregator is used that is positioned between users and utility companies. Utility companies adjust the power supply and unit price based on the aggregated demand forecasts of residential users. Figure 1 shows the communication process of distributed DR schemes. At a specific period before the electricity usage (e.g., a day), the utility company sends an initial unit price to users. HEMS uses predefined settings like application starting time, running duration, and the initial unit price to schedule the households consumption. The individual demand forecasts from residents are then sent to the aggregator. Next, the aggregator combines all the forecasts and sends the total demand forecast to the utility company. After receiving the cumulative demand forecasts, the utility company adjusts the unit price according to its own optimization function and sends the new unit price back to users. This process is repeated until both users and utility company achieve their optimization goals.

As shown in [3], FDI attacks on power demands can lead to false peak consumptions at specific time slots, causing them to have high unit price. Then, other users will adjust their applications demand away from the slot. As a result, there is low real demand during the attacked time slot and the attacker can enjoy a lower unit price during this time slot.

B. Deep Learning Models

Deep learning techniques use neural networks to learn multiple representations of the underlying patterns in the data. While its applications originated in the image and computer vision field, they are now widely considered in power systems. The use of DL methods for load forecasting has been explored in [14]. In [15], an IoT-based DL system is introduced which considers the influences of environmental factors. DL-based architectures are proposed in [16] and [17] to detect and classify power quality disturbances, while [18] focus on identification of compromised devices in smart grids. As for the FDI attacks, [9], [10], [19] focus on detecting stealthy and covert attack. However, these papers focus on FDI attack detection for state estimations and the monitoring of grid power flows. The existing work on DL-based FDI attack detection in power demand applications such as distributed DR schemes is limited. In [20], a convolutional neural network (CNN) is used to detect FDI attacks in distributed DR schemes and their results show successful detection of FDI attacks with higher accuracy.

C. Adversarial Machine Learning

AML was introduced in 2013 [21] to work against neural networks used in computer vision. Subsequently, AML methods have been widely researched. Fast Gradient Sign Method (FGSM) [22], Fast Gradient Value (FGV) [23], and DeepFool [24] are the state-of-art AML algorithms. There are some existing studies that have considered AML in power system DL applications. The authors of [25] discuss the vulnerabilities of DL applications in power system. In [26], adversarial DL against power state estimation using Saliency Map Attack is explored while [27] exploited vulnerabilities in load forecasting based on historical data.

To the best of our knowledge, there is no existing research on AML against FDI attack detection in DR applications. Therefore, to fill the research gap, this paper focuses on AML attacks against the FDI attack detection model in distributed DR schemes. The state-of-the-art CNN model from [20] is selected as the attack target because it is the latest and best FDI detection method for distributed DR applications.

III. Attack Model

A. FDI Attack

Consider the same scenario as in [20], one whole day is divided into 48 time slots of each 30 minutes. The attacker aims to increase the demand power in some time slot to project a fake peak demand. The adversary may do it for its own personal monetary gains or to cause damage, inefficiency, or losses in the power grid. Let the normal aggregated demand forecasts be $D = [d_1, d_2, \cdots, d_n]$ and $n = 48$. After the FDI attack, we have $\hat{D} = [\hat{d_1}, \hat{d_2}, \cdots, \hat{d_n}]$. Under a targeted FDI attacker, the adversary has a specific time slot as the target, e.g., to increase the values for 30th time slot, in which case the desired $\hat{d_{30}}$ should be larger than d_{30}. Under a non-targeted FDI attack, there is no specific time slot and the demand value may be increased at random time slot i, and therefore $\hat{d_i}$ is larger than d_i. A successful FDI attack should increase the peak value at any time slot by at least 0.1% of the daily aggregated demands. CNN-based models are good to use in time series data classification [28] and therefore could be use in distributed DR schemes to detect the FDI attack. The f_θ is the detection model and $f_\theta(D)$ is the prediction. Therefore, the FDI attack aims to achieve false negative evasions while obtaining higher values of D. The attack may be modeled as:

$$max: \quad \hat{D} \tag{1}$$

$$subject\ to: \ f_\theta(\hat{D}) = \text{normal}. \tag{2}$$

B. AML Methods

AML methods usually compute the gradient of targeted label's loss with respect to input for a given DL model and thereafter to update the input, resulting in a misclassification. For distributed DR schemes considered in this paper, the CNN-based model is used for FDI detection. Therefore, this model may be deceived by AML methods such as FGV [23], FGSM [22], and DeepFool [24].

1) FGV: This method is a one-step update to the input which calculates the gradients of the loss with respect to the input and maximize the loss. This update can be expressed as

$$D' = D + \epsilon \times \nabla_D L(f_\theta(D), Y). \tag{3}$$

In our case, D is the vector of demand values, Y is the label (attack or no attack), L is the loss function, and ϵ is a constant to control the size of the update. The final result D' is the adversarial data which may lead f_θ to give wrong classification.

2) FGSM: This method is similar to the FGV method, except that it uses the sign of the gradients:

$$D' = D + \epsilon \times sign(\nabla_D L(f_\theta(D), Y)). \tag{4}$$

3) DeepFool: DeepFool is an iterative method which keeps updating the input until the input is slightly beyond the hyperplane of classification. Considering the FDI attack in distributed DR schemes, the large peak values of D will decrease continuously until it is predicted as normal (no attack) by the CNN-based FDI detection model. The following expression is processed repeatedly until a "normal" label Y is achieved:

$$D' = D - \frac{f_\theta(D)}{\|\nabla_D L(f_\theta(D), Y)\|^2} \times \nabla_D L(f_\theta(D), Y). \tag{5}$$

In general, adversarial attacks can be of two types: white-box and black-box. In the white-box attack case, the adversary knows everything about the model being attacked. Therefore, the adversarial data are generated from a pre-trained model which is exactly the same as the model f. In the black-box attack case, the adversary has no knowledge about the detection model and needs to create its own pre-trained model f'. In the context of this paper, a black-box case refers to the scenario where the adversary has no knowledge about the original model f but can access to the training dataset for that model. Therefore, it is possible for the adversary to generate a f' which has similar prediction performances and to generate the adversarial data $\hat{D}$.

C. Simple Iterative FDI Attack framework

From results reported in existing literature, it is known that iterative AML methods like DeepFool can generate perfect adversarial data but they have low transferability. This is because these methods have a repeated process to obtain the adversarial samples and stop immediately when the samples are misclassified by a given model. Therefore the samples are unique to a given model and the transferability are low. Moreover, the AML methods like FGV and FGSM which use a single step to modify the input by using the gradient descent algorithm to find the adversarial directions that can lead to misclassifications seem to have better transferability among different models. To achieve better transferability while performing a black-box FDI attack detection in DR applications, we propose the Simple Iterative FDI Attack (SIFA).

With the objective to generate FDI samples against the distributed DR scheme, the values in the demand forecast D should be as large as possible and should follow the constraints in Equation (2). Therefore, the SIFA method uses the normal demand forecasts as inputs and the perturbations that are computed using the pre-trained model f_θ are added to the inputs to produce adversarial FDI demands. The SIFA method is proposed under the assumption that the attacker has the access to the historical demand forecasts to retrieve the normal demand values. The generated samples should be able to bypass the DL model-based detection techniques in order to successfully execute a FDI attack against the DR scheme. The general process of the SIFA method can be expressed as:

- Obtain pre-trained model f_θ (black-box/ white-box).
- Compute a perturbation $pert$ using the pre-trained model.
- Obtain adversarial FDI demand $D_a = D + pert$.
- D_a should bypass the detection model f_d: $f_d(D_a) = normal$
- The overall idea can be modeled as:

$$max: \quad D_a \tag{6}$$

$$\text{subject to}: f_\theta(D_a) = \text{normal}. \tag{7}$$

The proposed SIFA method also uses the gradient descent algorithm to find the direction that can lead to misclassifications by updating the samples in the same direction of the gradient descent. However, we make this process to be repeated. The gradient g of the loss function with respect to the input D is obtained by $g = \nabla_D L(f_\theta(D), Y)$ where f_θ is the pre-trained model and Y is the label corresponding to "attack". Normally, the single-step FGV directly uses this gradient to obtain the small perturbations $pert = \epsilon \times g$ and ϵ controls the extent of the modifications. However, this is a single step to update the values in D and the changes along the gradient direction may not be big enough for a model to make a misclassification. Therefore, to further modify the D, we propose to compute it iteratively using:

$$D_{k+1} = D_k + \epsilon \times \nabla_{D_k} L(f_\theta(D_k), Y). \tag{8}$$

In this way, D keeps changing its values in the direction of gradient of the loss and it moves further away from the original label after many steps. Moreover, in order to avoid local maxima or local minima of the values, it is preferable to use flexible weightages of changes. Therefore, the constant ϵ is changed to $\frac{\epsilon}{k}$ where k is the current number of steps. The degree of change for D is large at the beginning steps and gradually decrease for subsequent steps. The computation now becomes:

$$D_{k+1} = D_k + \frac{\epsilon}{k} \times \nabla_{D_k} L(f_\theta(D_k), Y). \tag{9}$$

As the name SIFA suggests, the proposed method is simple and only introduces minimal changes but is effective at generating adversarial FDI samples in distributed DR applications and to bypass the FDI detection mechanism. Compared with the existing FGV and FGSM methods, the proposed SIFA method has more computation complexity due to the iterative

Algorithm 1 Simple Iterative FDI Attack method

Require: D, ϵ, $step$

$k = 1$

while $k <= step$ **do**

  compute gradient $g = \nabla_d L(f_\theta(D), Y_a)$

  update $D = D + \frac{\epsilon}{k} \times g$

  $k = k + 1$

end while

adversarial data $D_{adv} = D$

return D_{adv}

TABLE I
CNN MODEL STRUCTURES

Layers	model f_d(lr=0.001)	model f_b(lr=0.001)
1	input 48	input 48
2	Cov1d 48, kernel 7	Cov1d 128, kernel 5
3	Cov1d 24, kernel 7	nil
4	flatten	flatten
5	dense 90	dense 256
6	dropout 0.3	dropout 0.35
7	dense 2 softmax	dense 2 softmax

calculations. The steps to generate adversarial FDI demands are listed in Algorithm 1. The input D is the normal demand forecast, ϵ is a constant to control the extent of change, and $step$ is a constant to limit the maximum number of iterations. The gradient g is computed with respect to D using the pre-trained model f_θ. Y_a is the label of being attacked because we are finding the perturbations along the gradient direction to attacked label. Therefore, D is modified gradually to get the final adversarial FDI sample D_{adv} which has higher likelihood of being misclassified by the detection model.

IV. RESULTS

We conduct simulations using real-life data and compare the proposed technique with FGV, FGSM, and DeepFool. The dataset used for the evaluation is that used in [20] which is synthesized from the Pecan street dataset. There are a total of 186093 sets of demand forecasts in the training dataset and 10054 of them are FDI attacked. There are a total of 91659 sets of demand forecasts in the test dataset and 4946 of them are FDI attacked. A CNN model for FDI detection is the attack target which is built following the same parameters as in [20] and is denoted by f_d. Another CNN model f_b is built separately without the knowledge of f_d, which is used to mimic black-box attacks. The model f_b is trained with different parameters that result in the best prediction performance using same training dataset. The details of the CNN models are shown in Table I and Table II shows their best detection results.

For the AML attack simulations, 10000 normal demands that are randomly selected from training dataset are used to generate the FDI attacked demands. For the black-box setting, the adversarial samples are generated using the model f_b which shows attackers are trying to produce FDI demands based on their own model. For the white-box setting, the adversarial samples are generated using the model f_d which is the same as the detection model f_d and this allows us to consider attackers

TABLE II
CLASSIFICATION PERFORMANCES OF WHITE-BOX (f_d) AND BLACK-BOX (f_b) MODELS

Model	Accuracy	precision	Recall	F1
f_d	0.978307	0.922259	0.654488	0.765048
f_b	0.978463	0.952496	0.632430	0.760145

that have complete knowledge of the detection model. The performance of the AML attacks are determined by evaluating the prediction recall made by the detection model f_d on the demands generated. The recall is the percentage of captured adversarial FDI data among the total generated adversarial FDI data.

A. State-of-The-Art AML Methods

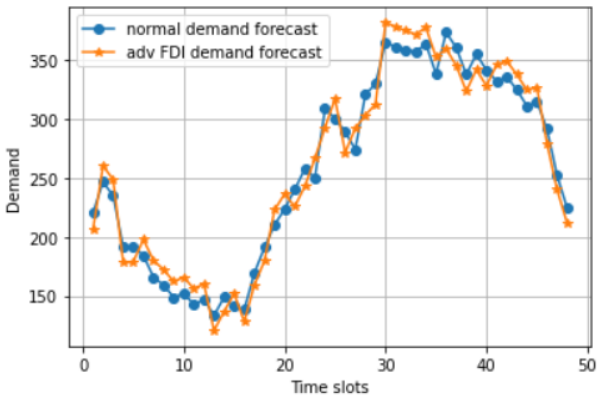

Fig. 2. Plot of normal and FDI attacked demand forecasts generated by FGSM.

Figure 2 presents a comparison between the normal and attacked demand forecasts generated by FGSM method (when $epsilon = 0.27$). It can be seen that there are obvious increments in the demand forecasts during time slots 30-35. These fake peak demands correspond to the FDI attack on the distributed DR scheme.

Figure 3 shows the detection recall for both FGV and FGSM methods. The ϵ values are from 0 to 1 with a step size of 0.01. It shows FGV has better attack performance than FGSM under both black-box and white-box settings. FGV achieves a recall of 0.25 under white-box setting and a recall of 0.6316 under black-box setting with $\epsilon = 0.01$. This shows that FGV is able to bypass the CNN-based FDI detection with much higher probabilities. We note that FGSM cannot generate true adversarial demands with small ϵ values under white-box and black-box scenarios, and therefore, there is no recall. FGSM can generate adversarial demands only when ϵ is large enough. FGSM has a recall of 0.9429 under a white-box scenario when $\epsilon = 0.27$, and a recall of 0.9793 under a white-box scenario when $\epsilon = 0.27$. The results show that FGSM is able to generate adversarial demands with larger values of ϵ but the likelihood of these generated attacked demands to be detected by CNN-based FDI detection is quite high. Figure 4 shows the corresponding false negatives when using FGV and FGSM. The false negative number shows how many of the generated FDI demands are

TABLE III
DEEPFOOL PERFORMANCES

Setting	Recall	False Negatives
white-box	0	8273
black-box	0.7056	2512

able to evade the FDI detection mechanism. The FGV method has 2329 false negatives under the white-box setting. The FGV method under black-box and FGSM under white-box scenarios both have about 500 false negatives.

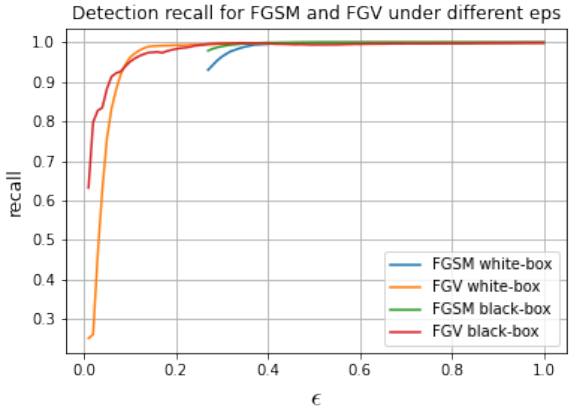

Fig. 3. Prediction recall under black-box and white-box settings.

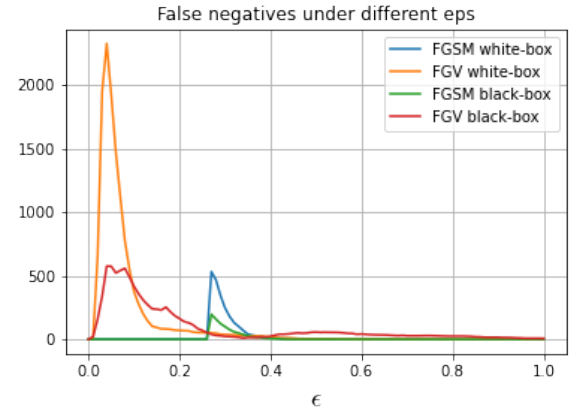

Fig. 4. False negatives under black-box and white-box settings.

Since DeepFool is an iterative method and does not use the parameter ϵ, its results are displayed separately in Table III. Under the white-box setting, DeepFool gets 8273 false negatives and the recall is 0. This shows that if attackers have the perfect knowledge of any detection model, they can run the optimal FDI attack on the distributed DR system and bypass the detection mechanism. For the black-box setting, DeepFool results in 2512 false negatives and the recall of prediction is 0.7056. Among these three methods, DeepFool performs the best to bypass FDI detection with a recall of 0 under white-box settings. For black-box settings, FGV performs the best with a prediction recall of 0.6316.

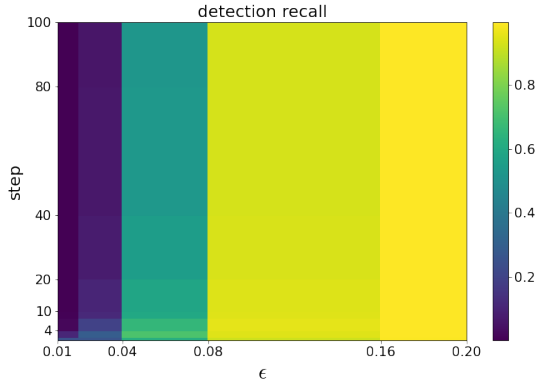

Fig. 5. Prediction recall using SIFA method under white-box setting

B. SIFA Method

The proposed SIFA method is also done in both white-box and black-box settings. The 2D plots are used to show the performance under different values of ϵ and $step$. The right column color bar shows the intensity level of recall values. From dark blue to light yellow, it represents value from 0 to 1. Figure 5 shows the detection recall under a white-box setting. The lowest detection recall is about 0.0007 when $\epsilon = 0.01$ and $step = 100$. This shows that the proposed SIFA method's performance under white-box attack is comparable to the DeepFool method and is much better than the FGV and FGSM methods. It also shows that greater the value of ϵ, the larger the detection recall, and greater the $step$ value the smaller the detection recall, when the SIFA method is used under white-box setting. Figure 6 shows the attack performance under black-box setting. From the results, the best performance is obtained when $\epsilon = 0.01$ and $step = 8$ and the detection recall is 0.4235, which is smaller than the FGV recall of 0.6316. Therefore, the SIFA method performs the best with highest bypassing rate among all the AML techniques under black-box attack. It can be seen from the results that the detection recall would gradually increase to 1 when ϵ or $step$ increase. In addition, the maximum false negatives under white-box setting for the SIFA method is 6767 and it is 1062 under black-box setting.

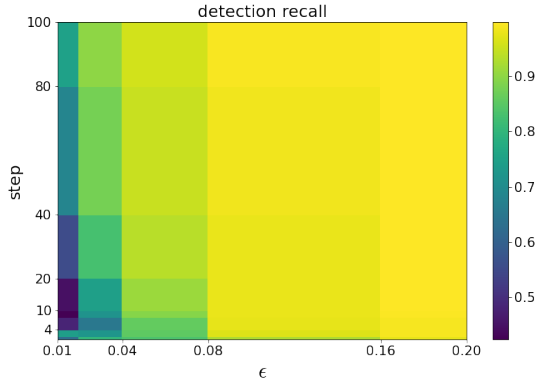

Fig. 6. Prediction recall using SIFA method under black-box setting

TABLE IV
COMPARISON OF RECALL OF ALL FOUR METHODS

Setting	FGV	FGSM	DeepFool	SIFA(our method)
white-box	0.2500	0.9429	**0**	0.0007
black-box	0.6316	0.9793	0.7056	**0.4235**

Both these values achieved by the SIFA method are smaller than that for the DeepFool method but larger than that of FGV and FGSM methods. Table IV shows the comparison of the lowest prediction recalls for all four methods and the lowest recall value among the four is shown in bold fonts. It can be seen that the proposed SIFA method performs well under both white-box and black-box settings. This method has good transferability and can effectively generate adversarial FDI samples to bypass the detection methodology.

V. CONCLUSION

Distributed DR schemes are an emerging way to achieve effective and efficient power supply management in smart grids. Owing to their vulnerability to cyber attacks, various defense mechanisms have been proposed for DR schemes, with DL-based methods being the most promising for detecting FDI attacks on demand forecasts. However, these methods are vulnerable to AML attacks. Well-fabricated samples could deceive the DL models and lead to unexpected malicious results. In this paper, we reviewed the DL-based approach to detect FDI in distributed DR schemes and showed that AML attacks can generate false peak demand forecasts that can bypass detection. We proposed the Simple Iterative FDI Attack and the evaluation results demonstrated that the proposed method is effective at generating fake demand forecasts and to evade the CNN-based detection mechanism with higher rate under black-box attack. Future work in this direction could focus on the different attack purposes such as grid blackout instead of energy theft. And also the defense and mitigation methods could be researched.

REFERENCES

[1] M. H. Albadi and E. F. El-Saadany, "Demand response in electricity markets: An overview," in *2007 IEEE Power Engineering Society General Meeting*, 2007, pp. 1–5.

[2] C. Barreto, A. A. Cárdenas, N. Quijano, and E. Mojica-Nava, "Cps: Market analysis of attacks against demand response in the smart grid," in *Proceedings of the 30th Annual Computer Security Applications Conference*, 2014, pp. 136–145.

[3] T. Dayaratne, C. Rudolph, A. Liebman, M. Salehi, and S. He, "High impact false data injection attack against real-time pricing in smart grids," in *2019 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*. IEEE, 2019, pp. 1–5.

[4] Y. Huang, H. Li, K. A. Campbell, and Z. Han, "Defending false data injection attack on smart grid network using adaptive cusum test," in *2011 45th Annual Conference on Information Sciences and Systems*. IEEE, 2011, pp. 1–6.

[5] Y. He, G. J. Mendis, and J. Wei, "Real-time detection of false data injection attacks in smart grid: A deep learning-based intelligent mechanism," *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2505–2516, 2017.

[6] S. Ahmed, Y. Lee, S.-H. Hyun, and I. Koo, "Unsupervised machine learning-based detection of covert data integrity assault in smart grid networks utilizing isolation forest," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 10, pp. 2765–2777, 2019.

[7] M. Ashrafuzzaman, Y. Chakhchoukh, A. A. Jillepalli, P. T. Tosic, D. C. de Leon, F. T. Sheldon, and B. K. Johnson, "Detecting stealthy false data injection attacks in power grids using deep learning," in *2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC)*. IEEE, 2018, pp. 219–225.

[8] Z. Zheng, Y. Yang, X. Niu, H.-N. Dai, and Y. Zhou, "Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 4, pp. 1606–1615, 2017.

[9] X. Niu, J. Li, J. Sun, and K. Tomsovic, "Dynamic detection of false data injection attack in smart grid using deep learning," in *2019 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, 2019, pp. 1–6.

[10] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*. IEEE, 2017, pp. 39–57.

[11] Y. Ozturk, D. Senthilkumar, S. Kumar, and G. Lee, "An intelligent home energy management system to improve demand response," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 694–701, 2013.

[12] A. Al-Ali, I. A. Zualkernan, M. Rashid, R. Gupta, and M. Alikarar, "A smart home energy management system using iot and big data analytics approach," *IEEE Transactions on Consumer Electronics*, vol. 63, no. 4, pp. 426–434, 2017.

[13] M. R. Asghar, G. Dán, D. Miorandi, and I. Chlamtac, "Smart meter data privacy: A survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2820–2835, 2017.

[14] A. Almalaq and G. Edwards, "A review of deep learning methods applied on load forecasting," in *2017 16th IEEE international conference on machine learning and applications (ICMLA)*. IEEE, 2017, pp. 511–516.

[15] L. Li, K. Ota, and M. Dong, "When weather matters: Iot-based electrical load forecasting for smart grid," *IEEE Communications Magazine*, vol. 55, no. 10, pp. 46–51, 2017.

[16] N. Mohan, K. Soman, and R. Vinayakumar, "Deep power: Deep learning architectures for power quality disturbances classification," in *2017 International Conference on Technological Advancements in Power and Energy (TAP Energy)*. IEEE, 2017, pp. 1–6.

[17] D. Wang, X. Wang, Y. Zhang, and L. Jin, "Detection of power grid disturbances and cyber-attacks based on machine learning," *Journal of Information Security and Applications*, vol. 46, pp. 42–52, 2019.

[18] C. Kaygusuz, L. Babun, H. Aksu, and A. S. Uluagac, "Detection of compromised smart grid devices with machine learning and convolution techniques," in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–6.

[19] S. Ahmed, Y. Lee, S.-H. Hyun, and I. Koo, "Feature selection–based detection of covert cyber deception assaults in smart grid communications networks using machine learning," *IEEE Access*, vol. 6, pp. 27 518–27 529, 2018.

[20] T. Dayaratne, M. Salehi, C. Rudolph, and A. Liebman, "False data injection attack detection for secure distributed demand response in smart grids," in *European Symposium on Research in Computer Security*. Springer, 2021, pp. 1–9.

[21] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.

[22] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[23] A. Rozsa, E. M. Rudd, and T. E. Boult, "Adversarial diversity and hard positive generation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 25–32.

[24] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2574–2582.

[25] Y. Chen, Y. Tan, and D. Deka, "Is machine learning in power systems vulnerable?" in *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*. IEEE, 2018, pp. 1–6.

[26] A. Sayghe, J. Zhao, and C. Konstantinou, "Evasion attacks with adversarial deep learning against power system state estimation," in *2020 IEEE Power & Energy Society General Meeting (PESGM)*. IEEE, 2020, pp. 1–5.

[27] Y. Chen, Y. Tan, and B. Zhang, "Exploiting vulnerabilities of load forecasting through adversarial attacks," in *Proceedings of the Tenth ACM International Conference on Future Energy Systems*, 2019, pp. 1–11.

[28] B. Zhao, H. Lu, S. Chen, J. Liu, and D. Wu, "Convolutional neural networks for time series classification," *Journal of Systems Engineering and Electronics*, vol. 28, no. 1, pp. 162–169, 2017.