

Research Article

Implementing AI Solutions for Advanced Cyber-Attack Detection in Smart Grid

Lilia Tightiz ¹, Rashid Nasimov,² and Morteza Azimi Nasab ³

¹School of Computing, Gachon University, 1342 Seongnam-daero, Sujeong-gu, Seongnam-si 13120, Gyeonggi-do, Republic of Korea

²Department of Information Systems and Technologies, Tashkent State University of Economics, Tashkent 100066, Uzbekistan

³Department of Electrical Engineering, Borujerd Branch, Islamic Azad University, Borujerd, Iran

Correspondence should be addressed to Morteza Azimi Nasab; m.aziminasab@iaub.ac.ir

Received 30 April 2024; Accepted 13 September 2024

Academic Editor: Mohamed Louzazni

Copyright © 2024 Lilia Tightiz et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As the backbone of modern power systems, the smart grid is one of the most critical applications of the Internet of Things. The intelligent electricity system faces various types of unauthorized malicious access, that is, cyber-attacks. With the development and application of information and communication technologies in traditional power systems, the improvement of physical-cyber systems in the smart grid also increases. Nowadays, the deployment of defensive measures has surged in response to the growing number of attacks aimed at the smart grid. Therefore, in this paper, we investigate cyber-attack identification using artificial intelligence-based models and identify them by estimating the state vector of the electricity network. We accomplish simulations by extracting data from the five-bus IEEE network to test the effectiveness of the proposed algorithm. False data attack vectors are then injected into the healthy measurements. In this way, to check the detection power of the proposed algorithms, 2,000 measurement samples have been taken from the five-bus network, half of which are considered healthy data and the other half as manipulated data. After labeling healthy and false data, machine-learning algorithms such as decision trees and k -nearest neighbor (KNN) have been used to investigate and identify this type of attack. Comparative analysis of the two proposed algorithms against commonly used methods demonstrates significantly improved accuracy. Specifically, according to the best depth for the decision-tree algorithm and k for the KNN algorithm, it is drawn with $k=3$ and the decision-tree algorithm with a depth of 9. According to the algorithms proposed in this article, the decision-tree algorithm with a depth of 9 in p -value of 0.45 and 0.64 and the nearest neighbor algorithm with $k=3$ in p is equal to 0.72, 0.98, and 1 represent better accuracy. Also, the results indicate that the two proposed algorithms have performed much more favorably than other classification methods. Additionally, the detection accuracy increases with higher p -values for these two algorithms. This problem shows that the detectors can detect false data injection attacks that cause more severe disturbances in the system.

1. Introduction

Electric energy has evolved from localized production in small plants to large generation and long-distance transmission due to the increased demand of industrialization [1]. Integration of the present power grids with communication networks results in the smart grid [2]. These are important for the efficient management of populations that have fast growth rates with electricity use on the rise. Smart grids have equipped themselves with advanced metering infrastructure, optimizing energy use and demand management [3]. However, they pose issues of high operational costs, technical complexities, and data security concerns. The idea of smart

technology plays an essential role in avoiding power black-outs and ensuring secure access to electricity, thus averting huge social and economic losses [4, 5].

Communication networks underlying smart grids have become vulnerable, thereby making the power system more vulnerable to attacks [6, 7]. A number of research works have documented the effects of various modes of attack on this infrastructure. These include innovative methods against the functionality of smart meters, cyberattacks manipulating network data in undetectable ways, and coordinated ways that exploit both the cyber and physical vulnerabilities of the power grid [8, 9]. It has also been shown that attacks

on network topologies and system state estimations raise operational costs while damaging SCADA systems [10]. The second set of strategies, such as the load redistribution attacks, disturb performance due to changes in loads, line currents, and measurements, hence increasing the cost of operation tremendously and even risking compromise on the reliability of the power system [11]. These findings bring out in clear terms the critical need for robust security measures to protect against such diverse and impactful threats.

False data injection attack (FDIA) is one of the most sophisticated cyber-attacks on smart grids. This attack should be understood in terms of targeting primary systems and specific subsystems in a smart grid, as compromising them will have direct implications for their operation in terms of physical infrastructure and economic stability [12].

These important challenges of cybersecurity in the smart grid have been addressed by researchers, who have identified two major methodologies to detect and mitigate FDIA: Model-based detection approach and data-driven detection approach. These strategic methods are very critical to ensuring operational stability and security in power grids to prevent catastrophic failures and huge economic losses [13].

Model-based detection approach leverages the implicit physical and operational models of a power system in detecting anomalies. In essence, this strategy identifies the discrepancies between the observed data and what should be happening within the grid's operations that could indicate a cyber-attack. State estimation methods belong to this category, which consist of anomaly detection employing network configurations and measurements [14]. Several research works have employed model-based methods, including relatively complex techniques like the Kalman filter and the Hilbert–Huang transform. Aforementioned techniques are of the subcategory model-based since they incorporate the underlying system models and operational dynamics to identify anomalies that may indicate the onset of cyber-attacks in mitigating the sophisticated threats that FDIA present to smart grids. For instance, a dynamic state estimation technique called the Kalman filter has dominated the power systems area in the detection of subtle changes that are indicative of FDIA [15]. Similarly, some researchers have applied the Hilbert–Huang transform in an attempt to identify spectral energy changes in voltage and current signals in smart DC microgrids with success, providing a robust way of detecting and diagnosing FDIA in real-time [16]. These model-based approaches are, hence, important in offering reliable defense against the evolution of cyber threats to energy systems.

The data-driven approach to detection makes use of machine learning and statistical analysis to learn from historical data in search of patterns or anomalies that could indicate cyber-attacks [17]. In contrast to the model-based approach, it does not need the physical models of the grid but rather the changing patterns in the data. As such, it becomes very effective against new and evolving attack vectors. Crucial among them would be the machine-learning models that will implement supervised and unsupervised learning algorithms to detect data tampering, while the methods for

statistical anomaly detection will use statistical methods to identify unexpected changes within grid data that may indicate the presence of FDIA [18, 19].

Lu et al. [20] applied a convolutional neural network (CNN) based on representation learning and made it possible for intelligent localization and recovery after attacks in cyber-physical power systems. They have integrated this data-driven detection approach with the model-based recovery approach through the minimum mean squared error estimator for the efficient restoration of system functionalities from an attack. It is a method that not only improves the accuracy of attack localization but also features quick and reliable system recovery, balancing innovative detection techniques with robust recovery processes. Jiang et al. [21] investigated a model-free reinforcement learning approach in a partially observable Markov decision process framework for detecting FDIA in state estimation. Basically, the main idea in this paper is data-driven methodology in which interacting policies adapt through interaction with the environment and require no predefined models of attacker behaviors or dynamics of the system. Ünal et al. [22] applied different state-of-the-art machine-learning algorithms, such as support vector machines (SVM), artificial neural networks (ANN), and ensemble learning techniques, to electricity consumption data for anomaly detection, which is an indication of FDIA. Particularly, in this paper, the authors emphasize the role of big data technologies and distributed computing frameworks that could bring in scalability advantages—very important in dealing with large volumes of data produced by smart grids. Cui et al. [23] developed a machine-learning-based cyberattack detection method, capturing spatiotemporal patterns using a graph Laplacian framework. Here, both unsupervised and supervised machine-learning techniques have been used to analyze system-wide measurements for detecting anomalies indicative of cyberattacks. Mohammadi et al. [24] proposed a new machine-learning classifier to detect FDIA by changes in consumer behavior and demand patterns in a P2P energy trading system. Gomathi et al. [25] have compared several machine-learning models on their effectiveness in detecting FDIA using a dataset from smart grid communication. In their case, the models compared are Naive Bayes, *k*-nearest neighbor (KNN), SVM, logistic regression (LR), decision trees, and random forest. With the same approach, Abudin et al. [26] presented an analytical survey on most machine-learning techniques for false data injection detection in power grids and assessed their effectiveness under different scenarios. The authors of this paper compare SVM, KNN, ANN, ensemble methods, autoencoders, and deep learning models—all of which have different advantages in detecting anomalies in smart grid data. With the same line of thought, Kermani et al. [27] deployed the XGBoost method for the detection and correction of false data in the energy balance to enhance operational efficiency within interconnected local energy networks. Chen et al. [28] applied graph neural networks and, in particular, used graph edge-conditioned convolutional networks that consider topology information besides node and edge features. The methodology enables the use of structural data inherent in

the power system network but often left underutilized in other models. On top of data-driven methods for FDIA detection, sophisticated enhancements come from evolutionary algorithms such as differential evolution and gray wolf optimization. Lu et al. [29] employed differential evolution to adapt the system defenses dynamically in a real-time fashion, merging data-driven responsiveness with model-informed adaptability. Similarly, Dehghani et al. [30] used a gray wolf optimization algorithm to fine-tune deep models in an attempt to optimize their performance for robust FDIA detection in DC microgrids.

Building upon the advancements in FDIA detection through sophisticated data-driven and model-informed methodologies as explored, this paper presents a refined approach using machine-learning algorithms such as decision trees, KNN, and SVM. Table 1 compares these methods against various techniques, highlighting their effectiveness in detecting and predicting unusual data activities. Leveraging the DC load distribution model enhances the efficiency of real-time data analysis, significantly reducing the processing burden and boosting the detection of data thefts. This approach addresses the challenges within the smart grid security domain, where smart grids utilize physical measurements and state variables collected every few seconds to assess grid status. Traditional solutions often falter against complex cyber threats like malware or malicious data injections, underscoring the critical need for robust security measures to counter diverse and sophisticated cyber threats in smart grids. Our comparative analysis using an IEEE five-bus network dataset showcases the superior accuracy of these methodologies.

The remainder of the paper is organized as follows: Section 2 reviews different kinds of cyber-attacks on the smart grid, while Section 3 presents its modeling for understanding possible vulnerabilities. In Section 4, various state estimation techniques that can be applied for the detection of anomalies are discussed. Section 5 investigates the consequences of malicious data on grid operations. In Section 6, algorithms for the detection of FDIA are presented and evaluated. The results and efficiency of the methods are analyzed in Section 7, followed by their comparative evaluation against benchmark methods in Section 8. Finally, Section 9 concludes this paper with a summary of the findings and future research directions.

2. Attacks on Smart Grids

At the moment, electric power is the most vital type of energy in the world. With the use of advanced power grids, the security systems ensure a continuous power supply. Physical and cybersecurity is part of the modern power system. Physical security is aimed at ensuring the power system resists severe disturbances, while cybersecurity ensures the safety of computer systems and associated communication networks. The growing application of information technology has increased the potential risk of cyber-attack threats to a modern power system. FDIA has long been recognized since 2009 as one of the important classes of cyber-attacks that can inflict damage on power system state estimation [31]. In case of success, an FDIA is capable of compromising the control center's decision-making by feeding it the wrong

data, which may result in major physical or financial damages to power systems, as shown in Figure 1. Figure 1 illustrates the various aspects of an FDIA on a smart power grid, which is an example of a critical infrastructure network. A number of attackers are placed around the central hub for any smart grid—a control center—in the middle. Connected to various segments of the power grid, a control center becomes the prime target for such cyber-attacks. Each attacker, at vantage points all around the power grid, symbolizes the distributed nature of cyber threats, all of which can emanate from a number of sources. This clearly points to various attacks against the different nodes of grid data communication and dispersion in place at power generation, transmission, and distribution facilities. Cyclic arrows indicate not only that grid operational processes are continuously interacting with each other and the control center but also where data interference by attackers can be expected. The graphical representation underlines the vulnerabilities of the cyber-physical interface of smart power systems and shows that robust cybersecurity measures shall be undertaken to mitigate related threats.

Figure 2 gives a schematic overview of the foundational research elements needed to understand and combat FDIA in power systems. Three primary vectors of attack exist, as can be noted from the flowchart: limited access to meters, incomplete information, and manipulated topology—what an attacker can use to inject malicious data into the system. The central element “FDIA” is the focal point from which these different research strands are fanning out, which suggests its central role in the study of cybersecurity within smart grids. The diagram breaks down into three significant areas of research: theoretical, which deals with the design of feasible injected data vectors that would remain undetected; defensive research, further divided into major and minor strategies for the protection of the grid; and practical impact on key system elements of FDIAs such as SCADA, energy management system, and monitoring management system. This structured representation will not only provide a road-map for the critical areas of FDIA research. Still, it will also integrate theoretical and practical perspectives under one umbrella of the defense strategy against such cyber threats.

3. Modeling a Smart Power Grids

Dispatching equations, which express the energy distribution in all transmission lines, are employed in the state estimation process. The load distribution in the AC dispatching model is based on nonlinear equations that take reactive and active power into account. Because the equations are linear, employing the DC dispatching model is a more stable and convenient approach, even though it is less precise than the AC model. The active power that is transferred in the DC mode from the i th bus to the j th bus in Figure 3 is given by Equation (1):

$$P_{ij} = \frac{V_i V_j}{X_{ij}} \sin(\theta_i - \theta_j). \quad (1)$$

In the electrical network with n buses, the m bus has a voltage of V_i and a phase angle equal to θ_i , and X_{ij} shows the reactance between the m bus and the j bus in the power

TABLE 1: Comparative analysis of our data-driven method against various machine-learning techniques.

Reference	Method	Advantages	Disadvantages	Comparison with our method
[25]	Logistic regression	Simple to implement and interpret at binary classification	Struggles with nonlinear problems, not flexible with feature interactions	Our method offers enhanced flexibility and efficiency with nonlinear data over logistic regression.
[25]	Random forest	High accuracy, robust against overfitting, interpretable results	High computational time, requires substantial computational resources	Our method using decision trees and KNN has lower computational time and requires fewer resources.
[27]	XGBoost	Excellent performance on varied data types, handles large datasets effectively	Can be prone to overfitting, requires careful tuning	Our method provides a simpler and more intuitive approach with comparable accuracy but less complexity.
[9]	SVM	High accuracy in linear and nonlinear data, reduces overfitting risk	Requires fine-tuning of parameters, high computational time	Our method using decision trees and KNN requires less parameter tuning and has lower computational time.
[23, 28]	Graph neural networks (GNN)	High accuracy in detecting complex attacks, ability to learn complex features	Requires large datasets for training, high computational time	Our method using decision trees and KNN requires less data and has lower computational time.
[20]	Convolutional neural networks (CNN)	High accuracy in detecting local features, deep learning capability	Requires large datasets for training, high computational time	Our method using decision trees and KNN requires less data and has lower computational time.
[21]	Reinforcement learning	Ability to learn from interactions, high accuracy in dynamic environments	Requires fine-tuning of parameters, high computational time	Our method using decision trees and KNN requires less parameter tuning and has lower computational time.
[29, 30]	Evolutionary algorithms	Global optimization search capability, high flexibility	High computational time, complex implementation	Our method using decision trees and KNN has lower computational time and simpler implementation.

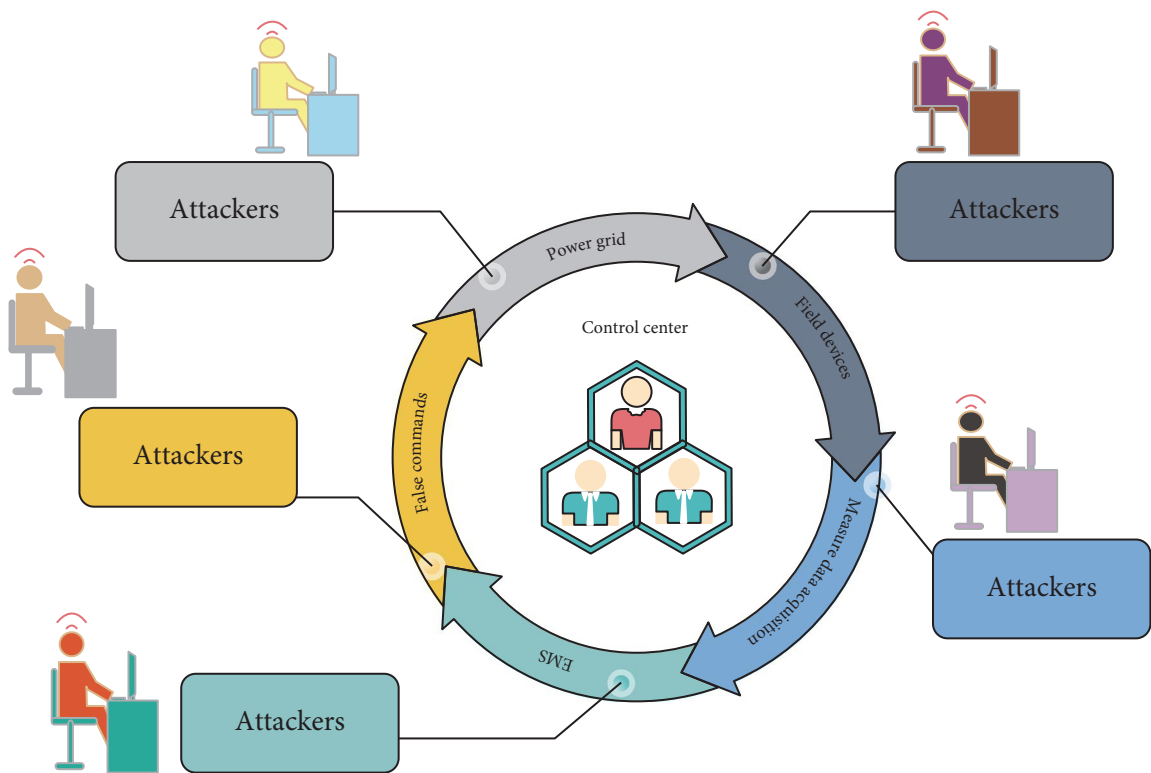


FIGURE 1: FDIA in smart power grids.

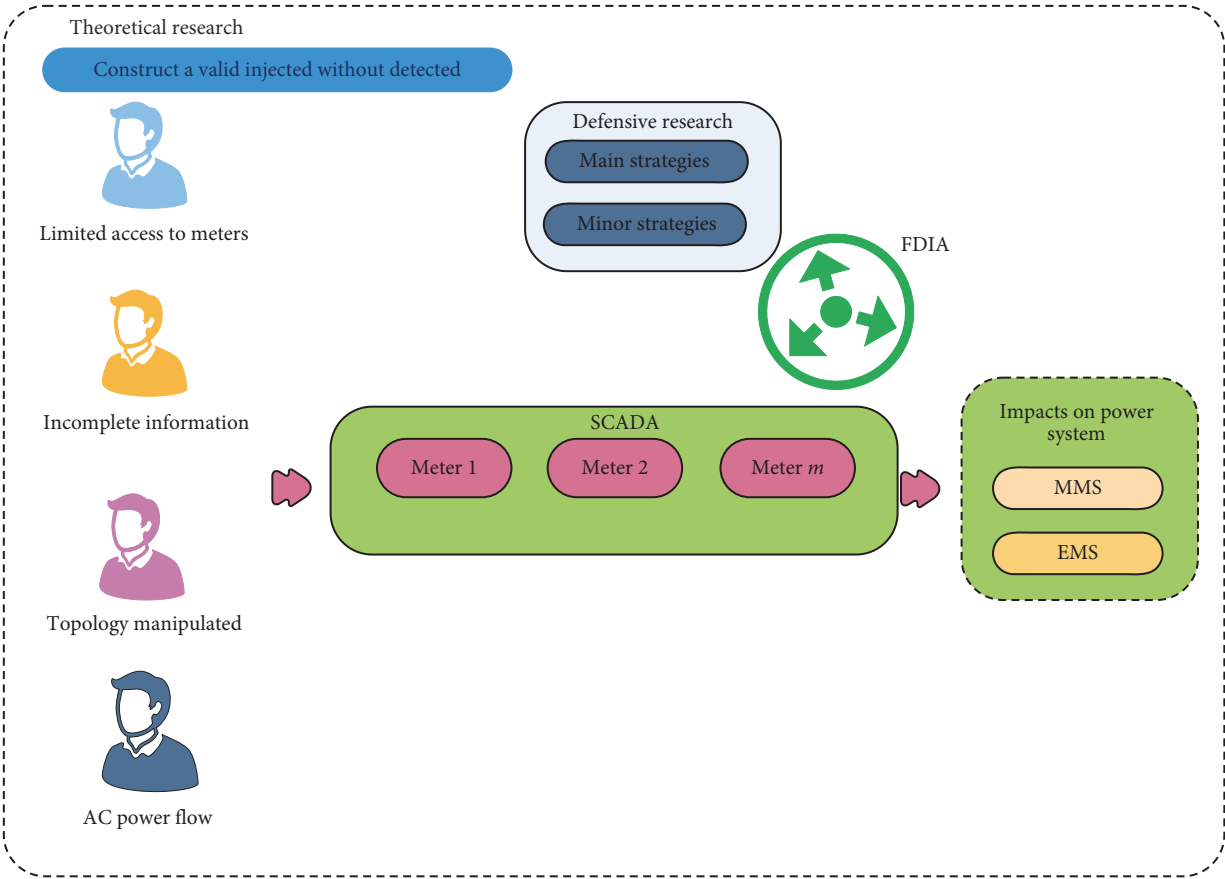


FIGURE 2: Basic research on attacks.

criterion, assuming a normal distribution for the measurement errors. Therefore, it is necessary first to introduce the concept of random measurement error. It is assumed that the measurements have an error, which means that the value obtained from the measuring device is close to its actual value. According to Equation (9), if z^{meas} is the measured value of a quantity, z^{true} is its actual value, and η is the random error of measurement, then we have the following:

$$z^{\text{true}} = z^{\text{meas}} + \eta, \quad (9)$$

where η is a random number that shows the uncertainty in the measurement, and its density function is usually considered as a normal distribution with zero mean:

$$DF(\eta) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-\eta^2}{2\sigma^2}\right), \quad (10)$$

where σ is called the standard deviation, and σ^2 is the variance of the random number. Note that σ can express the amount of measurement error. If σ is large, the measurement is relatively imprecise (the measuring device is weak), and a small value of σ indicates a low error percentage (the measuring device has better quality).

4.2.2. Concepts of Maximum Similarity. Based on the maximum similarity method, an estimate of x (called x^{est}) should be found to maximize the probability of its occurrence. The probability density function z_1^{mean} is as Equation (11):

$$\text{PDF}(z_1^{\text{mean}}) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(z_1^{\text{mean}} - z_1^{\text{true}})^2}{2\sigma^2}\right). \quad (11)$$

In this method, the estimation of the unknown parameter based on the maximum similarity is always expressed as a value of the main parameter that minimizes the sum of squares of the difference between the measured value and the actual value expressed as a function of the desired unknown parameter. Therefore, according to Equation (12):

$$\min_x j(x) = \sum_{i=1}^{N_m} \frac{(z_i^{\text{meas}} - f_i(x))^2}{\sigma_i^2}, \quad (12)$$

where σ_i^2 is the measurement variance of the i th step, $j(x)$ is the objective function, z_i^{meas} is the measured quantity of the i th step, N_m is the number of independent measurements, and f_i is the function that is used to calculate the measurement value of the i th step.

4.2.3. Matrix Relation. If we write the function $f_i(x_1, x_2, \dots, x_{N_s})$ as Equation (13):

$$f_i(x) = f_i(x_1, x_2, \dots, x_{N_s}) = h_{i1}x_1 + h_{i2}x_2 + \dots + h_{iN_s}x_{N_s}. \quad (13)$$

5. Malicious Data

Malicious data is any inaccurate information inserted into the estimator. Identifying and, if feasible, eliminating them is one of the state estimator's primary objectives. Errors can arise from a variety of sources in measurements. The poor accuracy of communication equipment and measurements is typically the cause of random errors. The estimator filters such errors if sufficient data is supplied for the estimation. The state estimate approach determines how these filters work. Defective or damaged connections are among the variables that can lead to significant measurement errors in measuring equipment. Moreover, substantial variances and mistakes in recording values might result from communication system malfunctions and noise produced by unforeseen interferences. Furthermore, significant variances and inaccuracies in recording values might result from communication system failures as well as noise produced by unforeseen interferences. Furthermore, incorrect topology information—also referred to as malicious data in the state estimator—may falsify a state estimator. Before the estimating process begins, some of these data can be found and eliminated using standard testing. Malicious data can be detected by huge variations between the input and output currents of a substation node and by negative voltage measurements that are several orders of magnitude greater or smaller than expected values. Indeed, not all bad data types can be quickly identified using these techniques; as a result, state estimators must incorporate more complex features to recognize and detect any invalid data. However, the state estimate approach employed determines how to identify and fix faulty data. The appearance of faulty data can vary based on the location and quantity of measurements. Two broad categories can be used to categorize these data:

- (1) Single faulty data: This condition arises when there is only one measurement in the system that has a significant inaccuracy.
- (2) Multiple faulty data: This situation arises when there are multiple incorrect measurements in the system. It might happen in measurements where there is a strong or weak association between the parameters under investigation. Strongly correlated measurement errors have a significant impact on the estimated values of the measurements, which are seriously jeopardized if a system error occurs. However, in measurements with weak correlation, error effects are typically negligible.

Furthermore, malicious data may lead to coordinated errors; in such a scenario, the faults coincide, potentially exacerbating the problem and making its detection more challenging. Thus, multiple types are divided into three groups:

- (1) Noninteractive, numerous invalid data: In this scenario, the data are typically found in measurements with low correlation and do not affect one another.

- (2) Inconsistent multiple interactive faulty data: Measurements that exhibit a high correlation with inconsistent faulty data.
- (3) Corresponding faulty data: In cases where there is a strong correlation, inaccurate data tends to be consistently present.

6. Detection of FDIA with Algorithm

6.1. Algorithm Solution

6.1.1. Alternative Machine-Learning Techniques. A variety of machine-learning methods have distinct advantages when it comes to identifying and resolving FDIAs in the context of smart grid cyber-security. Three such techniques are principally investigated in this study: SVM, KNN, and decision trees.

Decision Trees: A decision tree is a structure that resembles a flowchart, in which each internal node stands for an attribute test, each branch for the test's result, and each leaf node for a class label. The routes connecting the root and the leaf signify the rules for classification. The simplicity, interpretability, and ease of this method implementation make it advantageous.

KNN: This nonparametric technique is applied to regression and classification. The majority label among the closest neighbors is chosen for FDIA detection when the distance between the test sample and certain K points in the feature space is computed. KNN is prized for its ability to handle multiclass scenarios well and for being resilient to noisy training data.

SVM: It can be applied to regression, classification, and other tasks by building a hyperplane or set of hyperplanes in a high-dimensional space. To provide a decision boundary that is ideally spaced from any data point, the best hyperplane that divides all classes with maximum margin must be found. SVM is especially helpful because of its excellent accuracy in binary classification jobs and its kernel tricks that allow it to handle nonlinear boundaries [50].

6.1.2. Methodology and Solution Algorithm. The method for identifying FDIA in smart grids follows the phases listed below, with an emphasis on classifying data into "manipulated" or "healthy" categories using AI-based techniques:

(1) Data Preparation

Data Acquisition: Measure the system under normal operating and compromised data integrity situations using measurements obtained from the 5-bus IEEE test system.

Labeling and Preprocessing: To guarantee consistency in the model input, standardize features to have a zero mean and a unit variance. Indicate by labeling the data if it comes from a manipulated or unmanipulated dataset.

Feature Selection: Focus on characteristics that are especially manipulable and use domain expertise and exploratory data analysis to identify pertinent features that make the most contributions to model correctness.

(2) Validation and Training of Models

Decision Tree: To find the ideal complexity that prevents overfitting while retaining enough granularity to identify

anomalies, train the decision tree model at different levels.

KNN: Using a validation set, assess the model's performance over a range of variables to determine the ideal number of neighbors (k).

SVM: To begin, train the SVM using a linear kernel. If required, then investigate nonlinear kernels. Use cross-validation to optimize the margin and penalty parameters.

Algorithm 1 represents the pseudo-code of our approach in this paper.

6.2. Designing an FDIA Vector in a Smart Power Grid. In Section 4, the computation of the observation vector z was briefly covered. If an FDIA targets a smart power grid, the system model can be described as follows:

$$z = Hx + e + a, \quad (14)$$

where a is the attack injected vector. The important feature of this data vector is its thinness because the attacker can only control a small number of measuring devices. When the i th measurement is under the attacker's control, $a = 0$.

In general, data attacks are divided into two categories: covert attacks and detectable attacks. In the covert attack, the attacker has full knowledge of the system structure and parameters, especially H . As a result, for the control center to be unsuccessful in detecting the attack, it enters desired data as $a = Hc$. Therefore, according to Equation (15), the control center mistakenly detects $x + c$ instead of estimating x :

$$z = Hx + e + a = Hx + Hc + e = H(x + c) + e. \quad (15)$$

According to [22], the detection of invalid data is done by the conventional method of residual analysis. In this method, to detect the presence of invalid data, the residual is compared with a threshold value. Accordingly, if the manipulated measurement is injected into the DC state estimate, the estimator will provide a wrong state variable because the residual of the postattack measurement will be as Equation (16):

$$\begin{aligned} r_{\text{bad}} &= z_{\text{bad}} - \hat{z}_{\text{bad}} = z_{\text{bad}} - H\hat{x}_{\text{bad}} = z + a - H(\hat{x} + c) \\ &= z + a - H\hat{x} = r. \end{aligned} \quad (16)$$

This type of attack is called undetectable because the measurement residual will not increase after injecting the wrong data into the system; therefore, conventional methods cannot detect data injection attacks.

6.3. Under Study Network Specifications. This section discusses the preparation of the test system to test the performance of the proposed method. In this research, the IEEE five-bus network, whose diagram is shown in Figure 4, has been selected as the test environment for the proposed method to detect FDIAs. The information on the lines and buses is presented in Table 2. In this network, $m = 11$ and

Input: Measurement data from the 5-bus IEEE network

Output: Classification results for each technique indicating “Healthy” or “Manipulated”

1: Load dataset containing measurements from the IEEE 5-bus system

2: Data Preprocessing:

- Standardize features to zero mean and unit variance
- Label the data as “Healthy” or “Manipulated”

3: Feature Selection:

- Apply filter methods like Chi-Squared, Mutual Information to select top N features
- Validate feature selection by assessing the impact on model performance through cross-validation

4: Split the data into training and testing subsets (e.g., 75% training, 25% testing)

5: Define Machine-Learning Models:

- SVM_Model = Train SVM using optimal hyperparameters found through grid search
- KNN_Model = Train KNN with k determined via cross-validation for best accuracy
- DT_Model = Train Decision Tree at an optimal depth, also determined via cross-validation

6: Train each model on the training dataset

7: Define a unified detection function:

function detect_fdia (test_data, models):

```

    results = {}
    for model_name, model in models.items():
        prediction = model.predict (test_data)
        results[model_name] = "Manipulated" if prediction == "Manipulated" else "Healthy"
    return results

```

8: Evaluate each model on the testing dataset:

- Compute performance metrics such as accuracy,
- Compare model performances to select the best performing technique

9: Apply the detection function on test data to classify each sample:

test_results = detect_fdia (test_data, { “SVM”: SVM_Model, “KNN”: KNN_Model, “DT”: DT_Model })

10: Output the classification results from test_results End Algorithm

ALGORITHM 1: Enhanced FDIA detection in smart grids using multiple ML techniques.

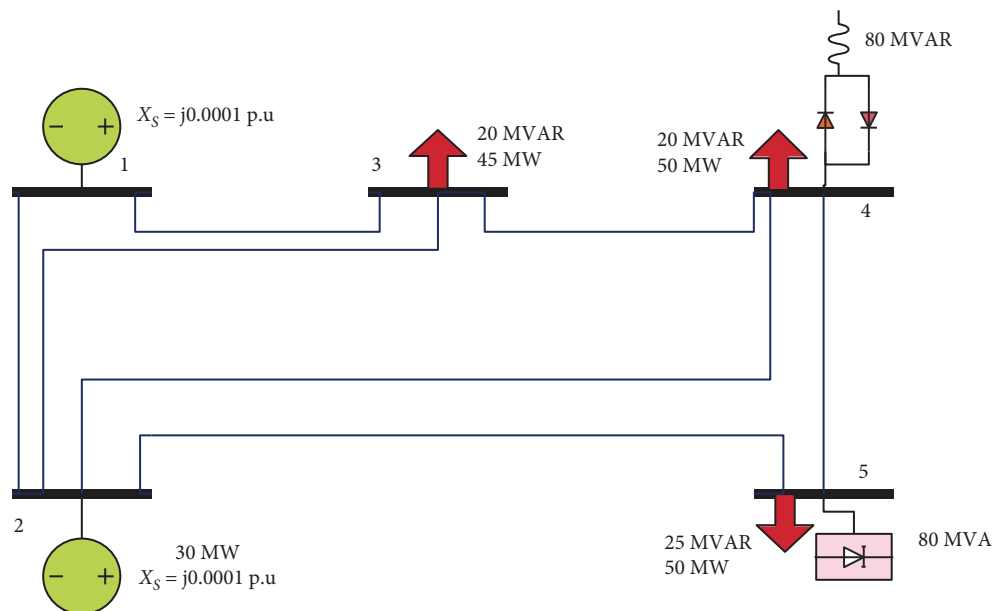


FIGURE 4: Five-bus IEEE network.

TABLE 2: Test network information.

a. The information of the tested network buses					
Bus PV	Assumed bus voltage	Generation		Load	
		MW	MVAR	MW	MVAR
1	1.06 + j0.0	0	0	0	0
2	1.0 + j0.0	40	30	20	10
3	1.0 + j0.0	0	0	0	15
4	1.0 + j0.0	0	0	0	15
5	1.0 + j0.0	0	0	0	15

b. The information of the tested network lines			
Bus PQ	Line impedance Z_{pq}		Line charging $\sqrt{Y_{pq}/2}$
	R (p.u)	X (p.u)	
1–2	0.02	0.06	0.0 + j0.015
1–3	0.08	0.24	0.0 + j0.025
2–3	0.06	0.25	0.0 + j0.020
2–4	0.06	0.18	0.0 + j0.020
2–5	0.04	0.12	0.0 + j0.015
3–4	0.01	0.03	0.0 + j0.010
4–5	0.08	0.24	0.0 + j0.025

c. MW limit for the branches in the tested network	
Line	MW limit (p.u)
1–2	0.8
1–3	0.3
2–3	0.2
2–4	0.2
2–5	0.6
3–4	0.1
4–5	0.1

$n = 5$, DC dispatching is considered, and its related relations are described. The network topology matrix (H) is calculated using the Matpower plugin in MATLAB software.

After calculating the matrix (H) based on $a = Hc$ in the previous section, the matrix of the attack vector is calculated according to the different values of the vector c , which is equal to $(c = [c_1 c_2 c_3 c_4]^T)$ in the network with five buses. Using the relation Equations (17) and (18), the number of specific states that can be designed from the values of attack vector c is 15 states. These states are shown in Table 3, and in Table 4 as an example, the numbers of these vectors in one lunch of the program are presented for each mode (in these two tables, K is the number of network measurements that the attacker is able to manipulate).

$$c(n, k) = \frac{n!}{k!(n-k)!}, \quad (17)$$

$$\binom{4}{1} + \binom{4}{2} + \binom{4}{3} + \binom{4}{4} = 4 + 6 + 4 + 1 = 15. \quad (18)$$

These modes are considered to check the power of attack detection for different attack vectors created by combining

different modes of manipulated information. Also, checking the impact of the information damage of one bus on the information damage of other buses is analyzed and investigated in a power network.

7. Results and Discussion

Two thousand measurement samples have been taken from a network with five busses to check the detection power of the proposed algorithms, half of which is considered valid data and the other half as manipulated data. Based on the number of measurements that are attacked each time the sample is taken, the attack vector is created according to the equations mentioned in the previous section and is added to the valid data. Also, to classify the obtained data, it is necessary to label both groups. Therefore, from these 2,000 samples, 1,000 of them, which are related to healthy data, are labeled as (one), and the other 1,000, which are related to the manipulated data, are assigned the label (two). It should be noted that 75% of this data was used for training and 25% for testing. In this part, based on the proposed classification algorithms, diagrams related to the analysis of attack detection power are drawn for different states of the attack vector, and then the performance of each algorithm is reviewed and evaluated.

7.1. Criteria for Evaluating the Accuracy of Algorithms. True positive, true negative, false positive, and false negative are

TABLE 3: Possible values for vector c .

Different modes for (k)	k	c_1	c_2	c_3	c_4	$c = [c_1 c_2 c_3 c_4]^T$	$a = Hc = [a_1 a_2 a_3 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11}]^T$
1	5	c_1	0	0	0	$c = [c_1 0 0 0]^T$	$[a_1 0 0 a_4 0 0 a_7 a_8 a_9 0 0]^T$
2	5	0	c_2	0	0	$c = [0 c_2 0 0]^T$	$[0 0 a_3 a_4 a_5 0 0 a_8 a_9 a_{10} 0]^T$
3	7	0	0	c_3	0	$c = [0 0 c_3 0]^T$	$[0 a_2 0 0 a_5 a_6 a_7 0 a_9 a_{10} a_{11}]^T$
4	5	0	0	0	c_4	$c = [0 0 0 c_4]^T$	$[0 0 a_3 0 0 a_6 a_7 0 0 a_{10} a_{11}]^T$
5	7	c_1	c_2	0	0	$c = [c_1 c_2 0 0]^T$	$[a_1 0 0 a_4 a_5 0 a_7 a_8 a_9 a_{10} 0]^T$
6	10	c_1	0	c_3	0	$c = [c_1 0 c_3 0]^T$	$[a_1 a_2 0 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11}]^T$
7	9	c_1	0	0	c_4	$c = [c_1 0 0 c_4]^T$	$[a_1 0 a_3 a_4 0 a_6 a_7 a_8 a_9 a_{10} a_{11}]^T$
8	9	0	c_2	c_3	0	$c = [0 c_2 c_3 0]^T$	$[0 a_2 0 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11}]^T$
9	9	0	c_2	0	c_4	$c = [0 c_2 0 c_4]^T$	$[0 0 a_3 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11}]^T$
10	8	0	0	c_3	c_4	$c = [0 0 c_3 c_4]^T$	$[0 a_2 a_3 0 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11}]^T$
11	10	c_1	c_2	c_3	0	$c = [c_1 c_2 c_3 0]^T$	$[a_1 a_2 0 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11}]^T$
12	10	0	c_2	c_3	c_4	$c = [0 c_2 c_3 c_4]^T$	$[0 a_2 a_3 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11}]^T$
13	11	c_1	0	c_3	c_4	$c = [c_1 0 c_3 c_4]^T$	$[a_1 a_2 a_3 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11}]^T$
14	10	c_1	c_2	0	c_4	$c = [c_1 c_2 0 c_4]^T$	$[a_1 0 a_3 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11}]^T$
15	11	c_1	c_2	c_3	c_4	$c = [c_1 c_2 c_3 c_4]^T$	$[a_1 a_2 a_3 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11}]^T$

represented, respectively, by TP, TN, FP, and FN in Equation (19). The accuracy of the Acc detection performance is taken into consideration if there are an equal number of attacked and healthy samples in the training data.

$$\text{Acc} = \frac{|\text{TP}| + |\text{TN}|}{|\text{TP}| + |\text{TN}| + |\text{FP}| + |\text{FN}|}. \quad (19)$$

Nevertheless, if the sampling is unstable, the number of attacked samples is significantly more or less than the number of healthy samples; in this case, Equation (20) is used to evaluate the performance of the detection accuracy.

$$F_1 = \frac{2|\text{TN}|}{2|\text{TN}| + |\text{FP}| + |\text{FN}|}. \quad (20)$$

There is always a potential attack vector that can introduce false information into the power grid covertly if $k \geq m - n + 1$. In this relationship, m and n stand for the number of measurements and state variables, respectively, and K is the number of measurements that the attacker can change. Conversely, the attack can be identified if the attacker can alter fewer measurements than $m - n + 1$. This paper and most other FDIA studies often employ the $p = k/m$ ratio to examine the attack detection performance. The diagnostic performance is assessed by various values of p to verify attack detection along various attack vectors in real-world scenarios.

7.2. Evaluation of KNN Detection Accuracy. In this part, the detection accuracy of the KNN algorithm for different values of k and all the different states of the attack vector (K) (listed in Table 3) is drawn in Figures 5, 6, 7, 8, 9, 10, and 11.

According to Figures 7, 8, 9, 10, and 11, with the increase of k (from mode 1 to mode 6), the detection accuracy of the KNN algorithm increases. From mode 7 to 10, with the increase of k , the detection accuracy is almost fixed.

Nevertheless, it can be seen that from mode 10, the detection accuracy gradually decreases with the increase of the value of k . With a general review of all modes, it can be seen that the lowest detection accuracy is in the first mode, and as it reaches the last graphs, the detection accuracy is improved, and the last mode has the best detection accuracy. Figure 12 is averaged for 16 states of the attack vector (in test mode).

To better understand this matter and find the appropriate k , in Figure 13, the accuracy of this algorithm in attack detection is drawn by considering $k = (7, 5, 3)$ as a constant in terms of p in the test mode. In other words, the same number of data is manipulated, the accuracy of detection of these states is averaged, and results are presented according to the accuracy of detection. According to Figure 13, it can be seen that when the number of manipulated data is less than $k = 7$, the algorithm has better accuracy. However, with the increase of p , the accuracy improves when $k = 3$, as shown in Table 5.

7.3. Evaluation of Decision-Tree Detection Accuracy. In this part, the detection accuracy of the decision tree algorithm for different depths in this algorithm for each of the different states of the attack vector (K) specified in Table 3 is drawn in Figures 12, 14, 15, 16, 17, and 18.

Figures 12, 14, 15, 16, 17, and 18 show that with the increase in depth and K , the algorithm's detection accuracy has improved in all cases.

In Figure 19, the detection accuracy of this algorithm for depth changes from 1 to 20 are plotted according to the average accuracy in the presence of different attack vectors in training and testing modes. Accordingly, it can be seen that the algorithm's accuracy improves with the depth of 1–9. found and then fixed.

Also, in Figure 20, the accuracy of FDI targeted attack detection for the decision tree algorithm at depths 5, 7, and 9 is drawn in terms of different p values. As can be seen, this algorithm has better accuracy in the depth of 9.

TABLE 4: Sample values of the attack vector in a single execution of the program.

Mode for (k)	k	c_1	c_2	c_3	c_4	$a = Hc = [a_1 a_2 a_3 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11}]^T$
1	5	0.267	0	0	0	$[-0.955 \ 0 \ 0 \ 2.488 \ 0 \ 0 \ -957 \ 3.446 \ -2.488 \ 0 \ 0]^T$
2	5	0	0.0916	0	0	$[0 \ 0 \ -8.491 \ 3.078 \ 0 \ 0 \ -8.491 \ 11.578 \ -3.078 \ 0]^T$
3	7	0	0	-0.1126	0	$[0 \ 3.7153 \ 0 \ 0 \ 3.8021 \ -3.8027 \ 3.7153 \ 0 \ 3.8024 \ -11.3201 \ 3.8027]^T$
4	5	0	0	0	-0.0430	$[0 \ 0 \ -6.734 \ 0 \ 0 \ -1.451 \ -6.735 \ 0 \ -1.452 \ 8.186]^T$
5	7	0.0158	-0.0653	0	0	$[-0.568 \ 0 \ 0 \ 7.593 \ -2.202 \ 0 \ -568 \ 8.096 \ -9.732 \ 2.202 \ 0]^T$
6	10	-0.0215	0	0.0172	0	$[0 \ 0.772 \ 0.5633 \ 0 \ -2.005 \ -0.577 \ 0.575 \ 0.207 \ -2.787 \ 1.431 \ 1.7172 \ -0.571]^T$
7	9	0.177	0	0	0.137	$[-6.368 \ 0 \ -21.635 \ 16.565 \ 0 \ 4.664 \ -28.01 \ -22.934 \ -16.567 \ -4.664 \ 26.297]^T$
8	9	0	-0.066	0.152	0	$[0 \ -4.991 \ 0 \ 6.248 \ -7.387 \ 5.108 \ -4.992 \ 6.248 \ -13.632 \ 17.483 \ -5.107]^T$
9	9	0	0.0362	0	-0.0031	$[0 \ 0 \ 0.491 \ -3.357 \ 1.222 \ 0.105 \ 0.492 \ -3.357 \ 4.578 \ -1.114 \ -0.598]^T$
10	8	0	0	0.0357	-0.0102	$[0 \ 1.173 \ 1.604 \ -1.202 \ 1.541 \ 0.4251 \ -1.204 \ 3.928 \ -3.148]^T$
11	10	-0.005	0.0744	0.0703	0	$[0.221 \ -2.3170 \ -7.470 \ 0.135 \ 2.373 \ -2.097 \ -7.693 \ 7.606 \ 4.552 \ -2.373]^T$
12	10	0	0.0709	0.0336	-0.0603	$[0 \ -1.104 \ 9.433 \ -6.561 \ 1.255 \ 3.163 \ 8.329 \ -6.561 \ 7.861 \ 3.012 \ -12.596]^T$
13	11	0.0358	0	0.080	0.025	$[-1.275 \ -2.680 \ -3.818 \ 3.321 \ -2.743 \ 1.920 \ -7.778 \ 4.594 \ -6.064 \ 7.348 \ 1.897]^T$
14	10	0.0518	0.0364	0	-0.0153	$[-1.840 \ 0 \ 2.371 \ 1.424 \ 1.222 \ 0.511 \ 0.528 \ 3.267 \ -0.202 \ -0.711 \ -2.880]^T$
15	11	0.0148	-0.038	0.0443	-0.0573	$[-0.521 \ -1.462 \ 8.962 \ 5.004 \ -2.820 \ 3.424 \ 6.978 \ 5.527 \ -7.824 \ 7.707 \ -12.387]^T$

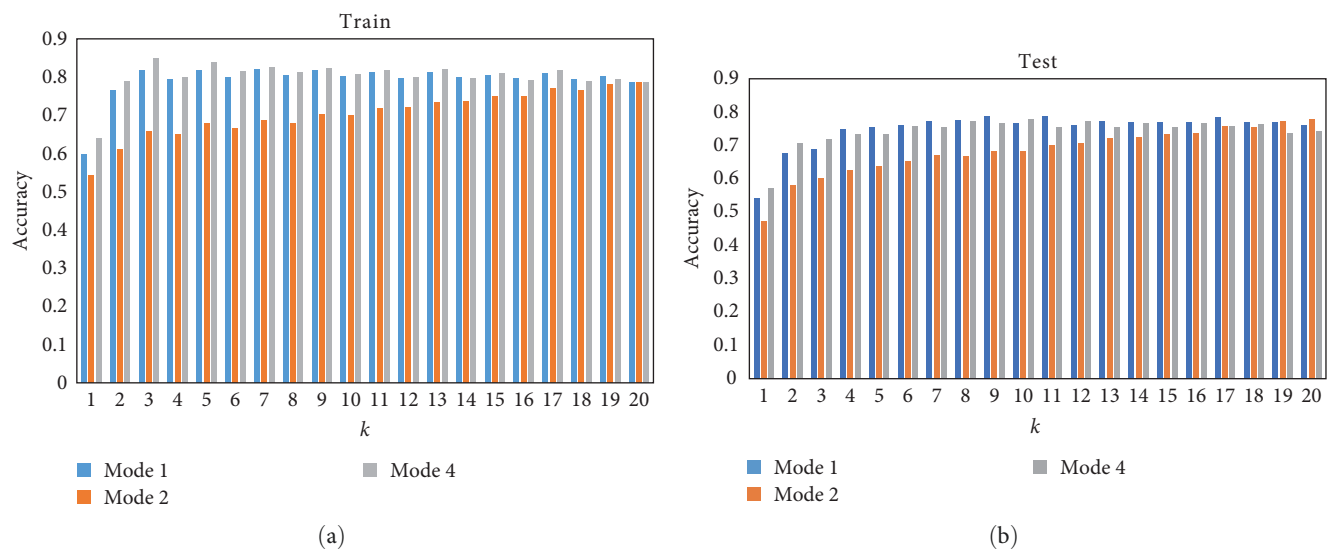


FIGURE 5: KNN detection accuracy for $K=5$: (a) train mode and (b) test mode.

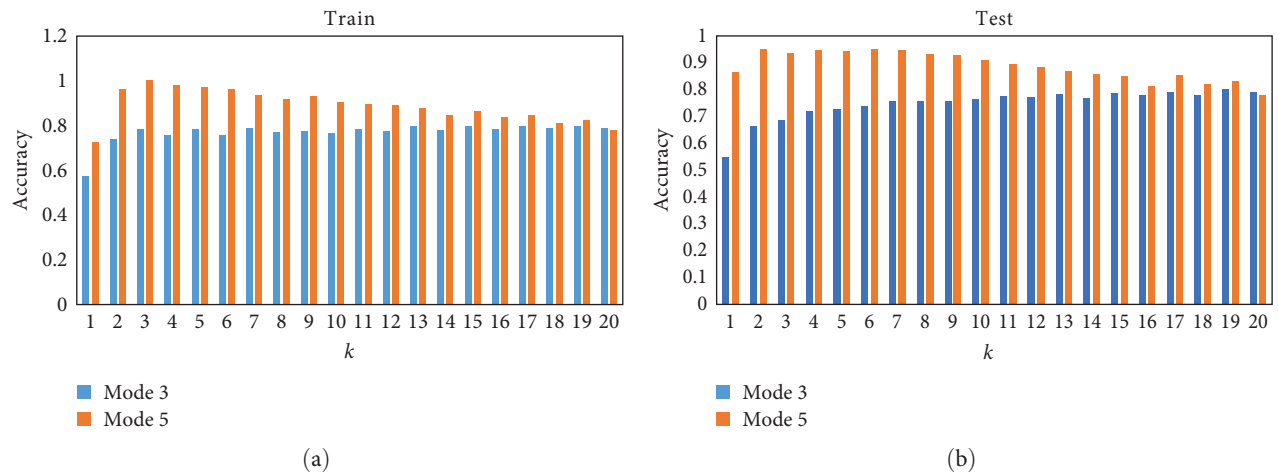


FIGURE 6: Nearest neighbor algorithm detection accuracy for $K=7$: (a) train mode and (b) test mode.

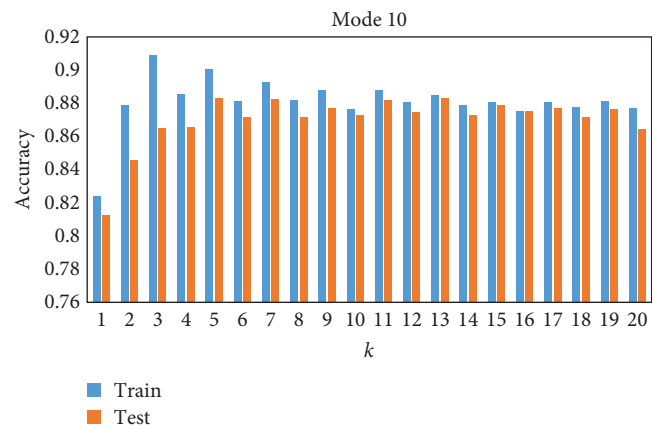


FIGURE 7: Nearest neighbor algorithm detection accuracy for $K=8$.

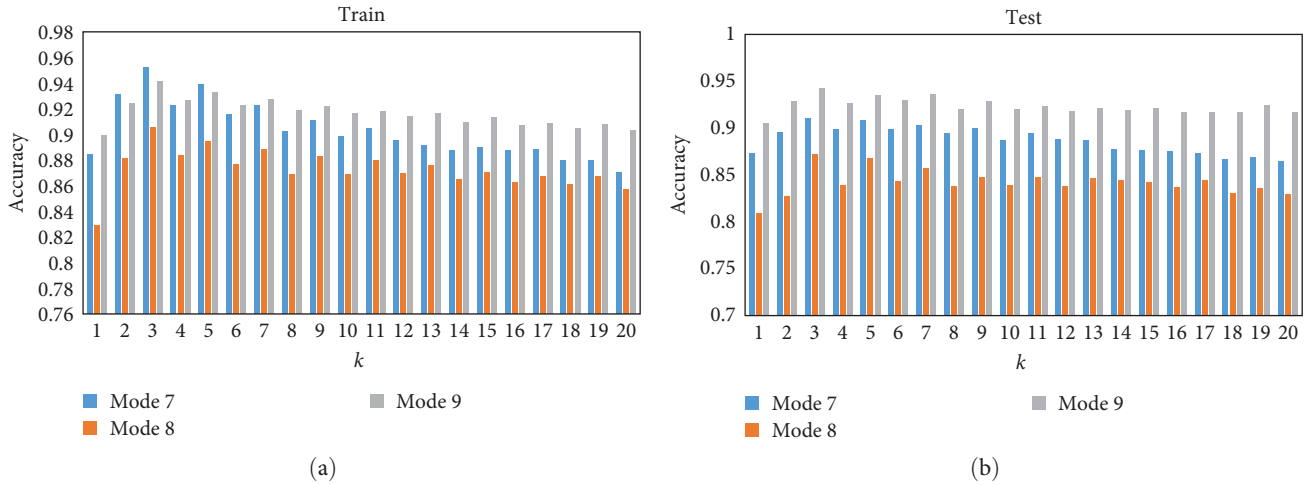


FIGURE 8: Nearest neighbor algorithm detection accuracy for $K=9$: (a) train mode and (b) test mode.

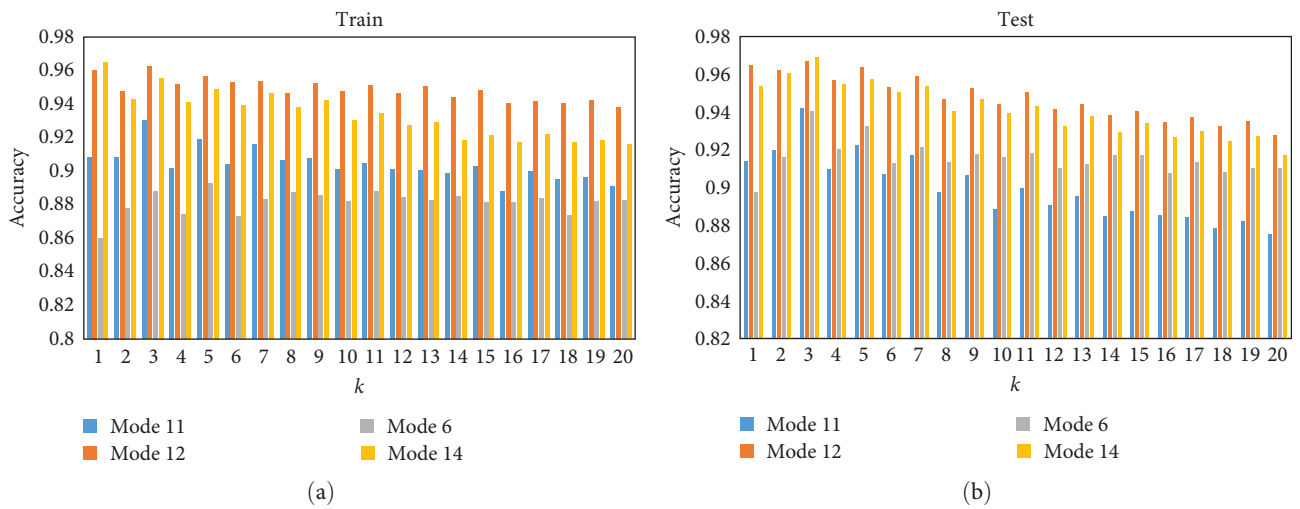


FIGURE 9: Nearest neighbor algorithm detection accuracy for $K=10$: (a) train mode and (b) test mode.

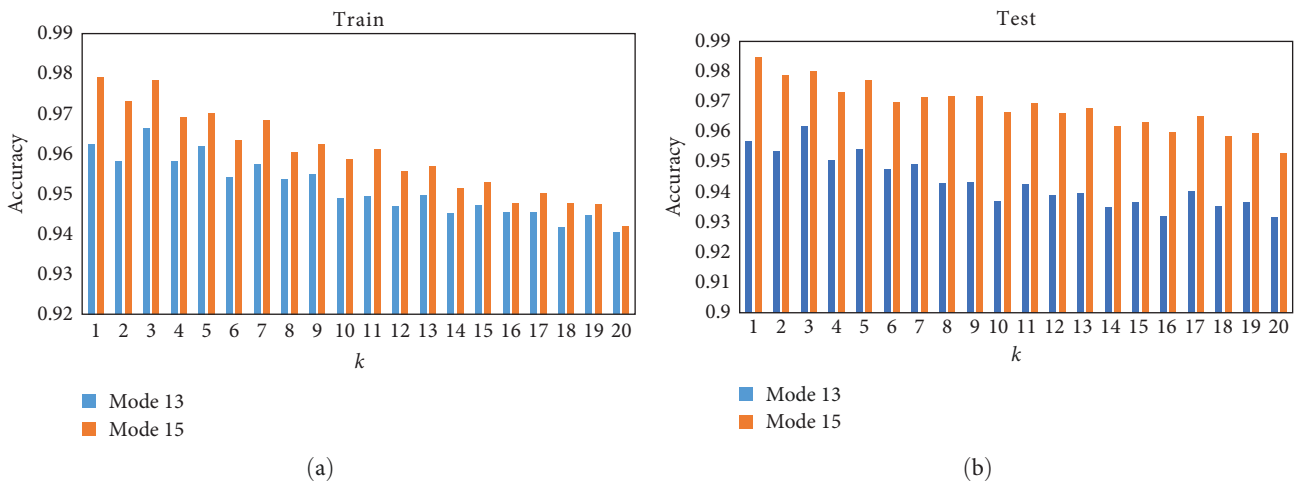


FIGURE 10: Nearest neighbor algorithm detection accuracy for $K=11$: (a) train mode and (b) test mode.

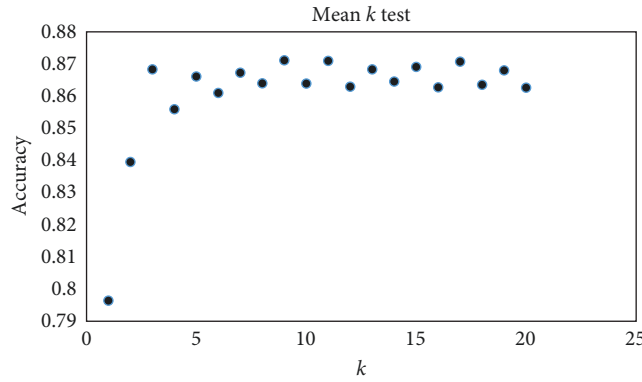
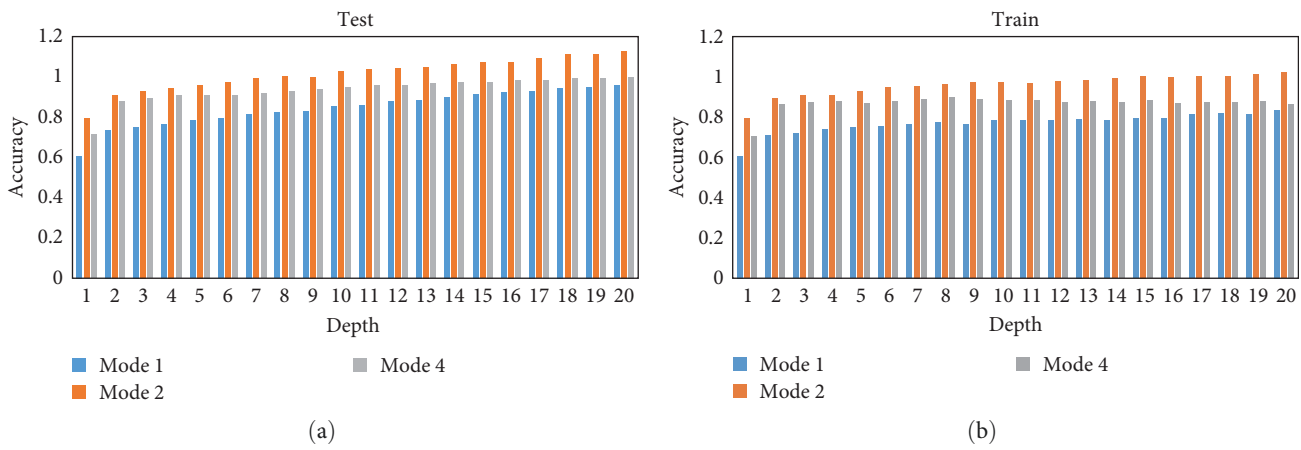
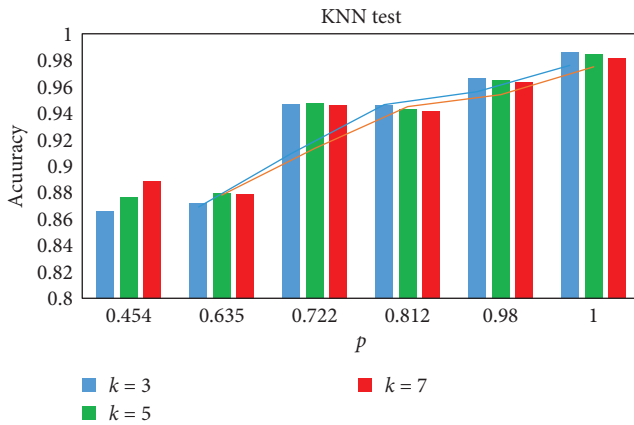


FIGURE 11: Detection accuracy of KNN algorithm for 15 states of attack vector in test mode.

FIGURE 12: Accuracy of the decision tree algorithm for $K=5$: (a) test mode and (b) train mode.FIGURE 13: Detection accuracy of KNN algorithm in terms of p .

7.4. Assessing the Accuracy of Detection of FDIAs Using Benchmark Algorithms

7.4.1. Detection of Bad Data Injection Attack with SVM Algorithm. With the aim of separating a data set into two classes, linear classification methods try to do the task by proposing a hypersurface (a linear equation). This method can classify collected data into two groups (valid and

TABLE 5: Detection accuracy of KNN algorithm in terms of p, K .

K, p	K	$K=3$	$K=5$	$K=7$
$K=5, p=.45$		0.855	0.870	0.855
$K=7, p=.63$		0.855	0.865	0.865
$K=8, p=.72$		0.956	0.954	0.951
$K=9, p=.81$		0.942	0.936	0.936
$K=10, p=.9$		0.964	0.962	0.961
$K=11, p=1$		0.984	0.983	0.980

manipulated data). The SVM is one of the linear classification methods that greatly finds the best hypersurface that separates the data of two classes with the maximum distance.

The detection accuracy of this algorithm for each mode of the attack vector and K (given in Table 3) is presented in Table 6, and the results are illustrated in Figures 21 and 22.

According to the diagram, the detection accuracy of this algorithm is between 0.50 and 0.80, which is much lower than the proposed algorithms.

7.4.2. Linear Discriminant Analysis (LDA) Algorithm. To separate classes of objects, this statistical technique uses a linear combination of attributes, and it is commonly used

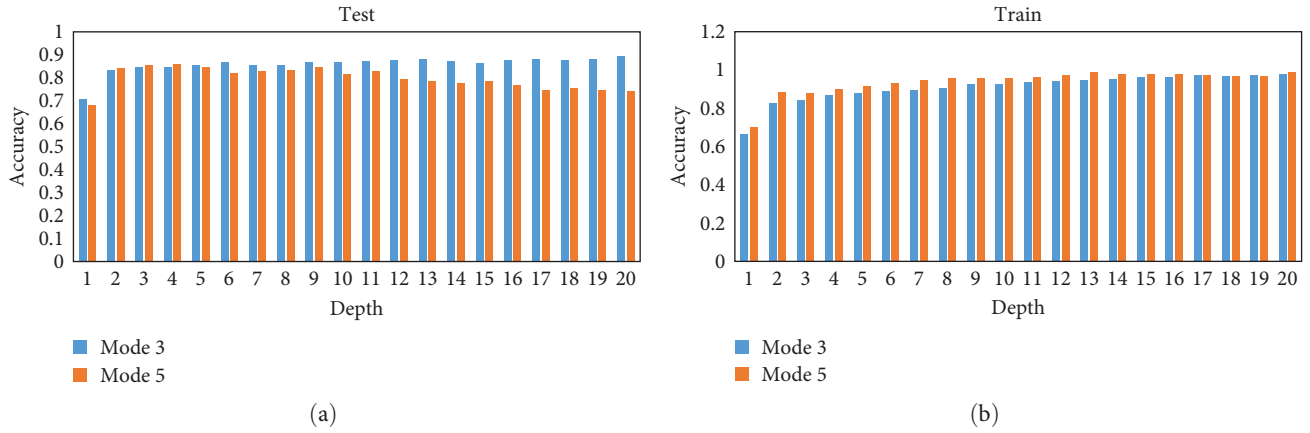


FIGURE 14: Accuracy of the decision tree algorithm for $K=7$: (a) test mode and (b) train mode.

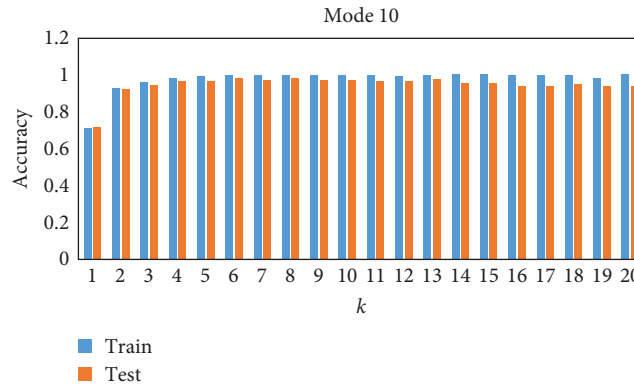


FIGURE 15: Accuracy of the decision tree algorithm for $K=8$.

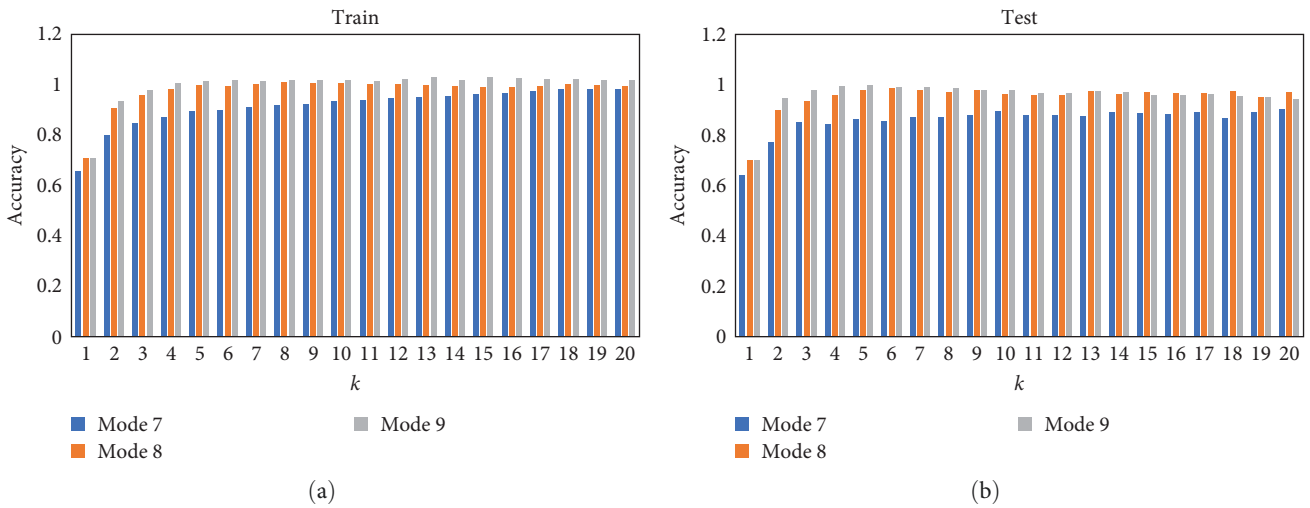


FIGURE 16: Accuracy of the decision tree algorithm for $K=9$: (a) train mode and (b) test mode.

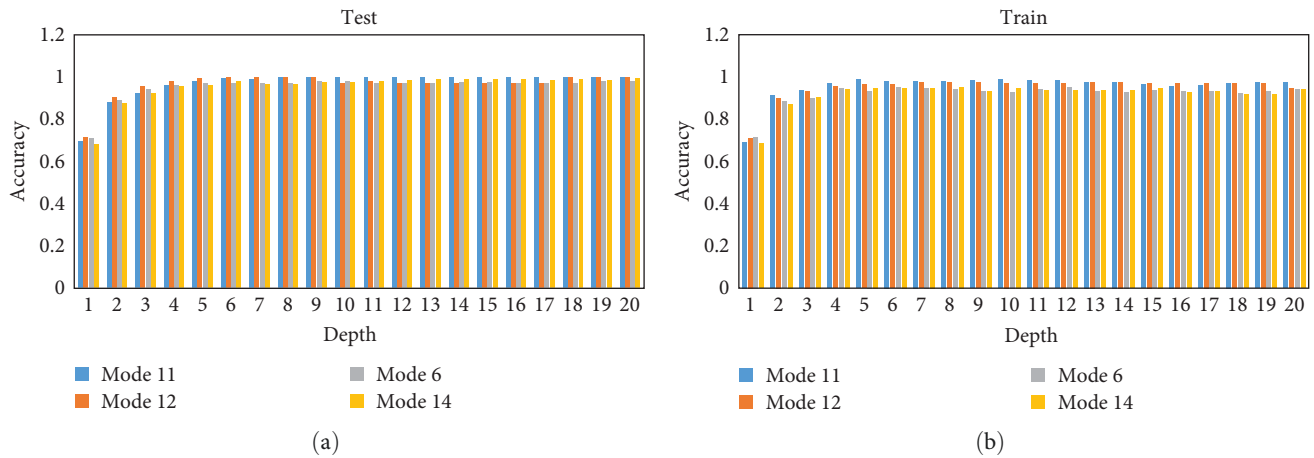


FIGURE 17: Accuracy of the decision tree algorithm for $K=10$: (a) test mode and (b) train mode.

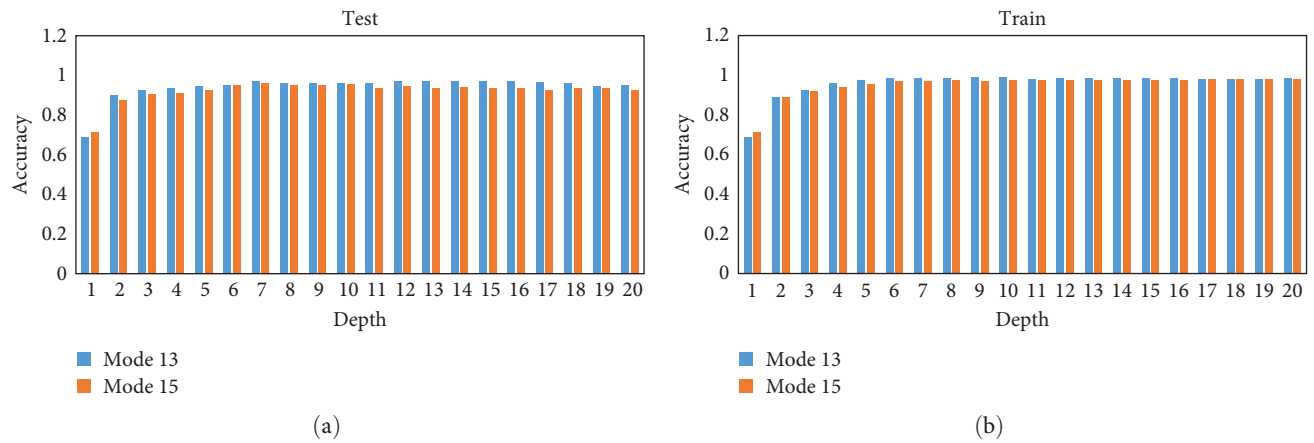


FIGURE 18: Accuracy of the decision tree algorithm for $K=11$: (a) test mode and (b) train mode.

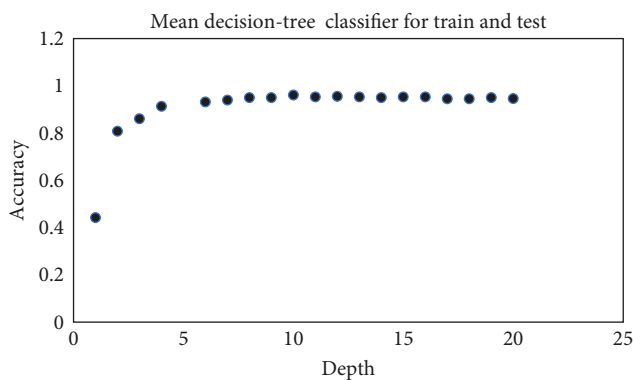


FIGURE 19: The average accuracy of the decision tree algorithm for different depths and attack vectors.

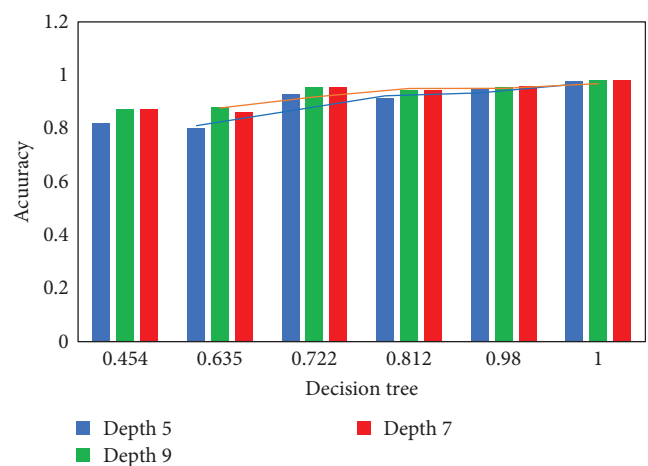


FIGURE 20: FDI targeted attack detection accuracy with decision tree algorithm in terms of p .

in machine-learning and pattern recognition. A linear combination of multiple features is expressed by an independent variable, which is strongly related to regression analysis and variance analysis. A class label serves as this independent variable in LDA. LDA and principal component analysis are closely related techniques that search for a linear

combination of variables that best describes the data. To achieve the highest interclass and lowest intraclass dispersion, the LDA approach depicts the vector data in a space with smaller dimensions. Table 7 displays the LDA detection

TABLE 6: SVM algorithm detection accuracy in terms of K .

SVM testing	SVM training	K
0.57	0.57	5
0.52	0.43	5
0.52	0.55	7
0.55	0.59	5
0.60	0.59	7
0.61	0.62	10
0.58	0.59	9
0.57	0.57	9
0.57	0.57	9
0.79	0.79	8
0.58	0.57	10
0.58	0.56	10
0.63	0.64	11
0.65	0.64	10
0.81	0.79	11

TABLE 7: Detection accuracy of LDA algorithm in different modes of attack vector.

Mode	LDA testing	LDA training
Mode 1	0.45	0.52
Mode 2	0.43	0.53
Mode 3	0.46	0.53
Mode 4	0.56	0.58
Mode 5	0.44	0.53
Mode 6	0.47	0.52
Mode 7	0.49	0.53
Mode 8	0.49	0.54
Mode 9	0.42	0.54
Mode 10	0.39	0.50
Mode 11	0.46	0.48
Mode 12	0.48	0.51
Mode 13	0.41	0.52
Mode 14	0.42	0.48
Mode 15	0.42	0.47

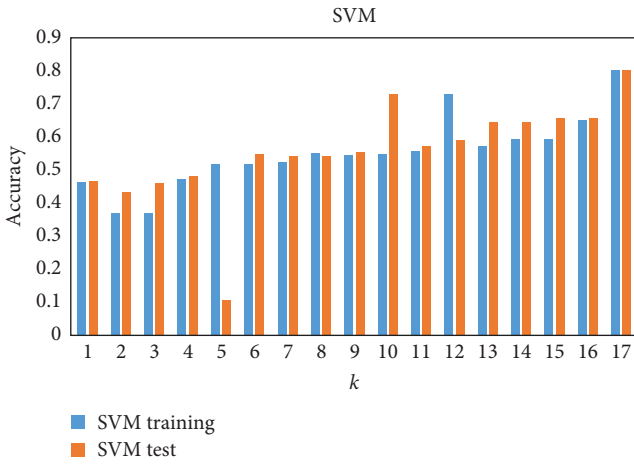


FIGURE 21: SVM algorithm detection accuracy for different attack vectors.

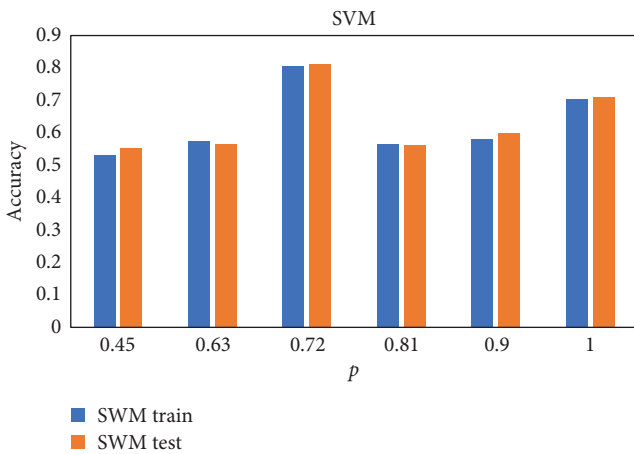


FIGURE 22: SVM algorithm detection accuracy in terms of p .

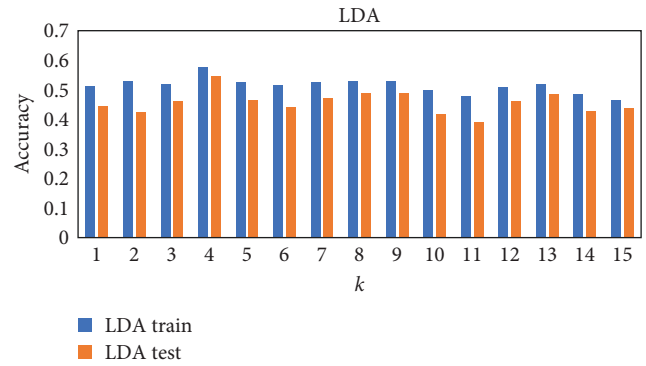


FIGURE 23: Detection accuracy of LDA algorithm in different states of the attack vector.

accuracy of the fake data injection assault for each attack vector state as well as K , which is specified in Table 3. These values are used to generate the graph in Figure 23.

According to the drawn graphs, it can be seen that the detection accuracy of this algorithm is between 0.35 and 0.60, and this accuracy is much lower than the proposed algorithms.

7.4.3. LR Algorithm. LR is a statistical technique that categorizes data according to the input field values. For problems with two distinct categories, this strategy can be used for binomial models; for problems with more than two categories, it can be applied to polynomial models. Figure 24 shows that LR connects the values of the input variable to the likelihoods of every potential category for the output variable by constructing a system of equations. The model can be used to estimate the probabilities of fresh data sets once it has been created. Each data set has a membership probability for each potential output category; the predicted output value for a given state is determined by selecting the target category with the highest likelihood.

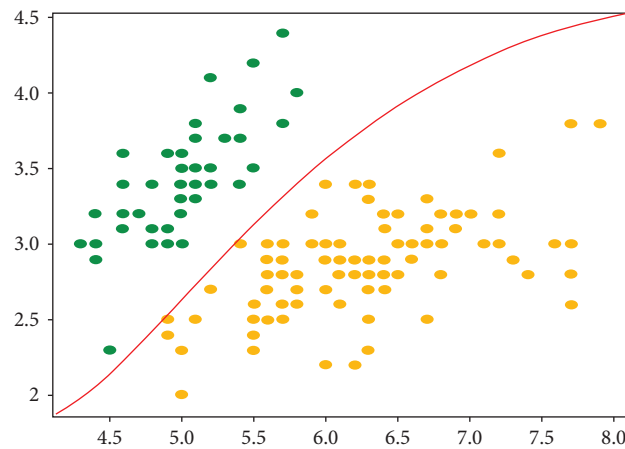


FIGURE 24: Classification using logistic regression algorithm.

TABLE 8: Detection accuracy of LR algorithm in different modes of attack vector.

Mode	Logistic regression classifier testing	Logistic regression classifier training
Mode 1	0.44	0.52
Mode 2	0.45	0.52
Mode 3	0.47	0.54
Mode 4	0.56	0.60
Mode 5	0.48	0.53
Mode 6	0.44	0.50
Mode 7	0.49	0.54
Mode 8	0.55	0.55
Mode 9	0.55	0.55
Mode 10	0.40	0.46
Mode 11	0.37	0.45
Mode 12	0.43	0.49
Mode 13	0.43	0.49
Mode 14	0.38	0.46
Mode 15	0.34	0.40

LR models typically have very high accuracy. These models predict probability for all variable categories and can operate with symbolic or numerical input variables. Table 8 presents the detection accuracy of the LR method for each attack vector mode and K . Figure 25 is then constructed based on these data.

As a result, it is evident that this algorithm’s detection accuracy is substantially lower than the other approaches, ranging from 0.30 to 0.60.

8. Evaluation and Comparison

The previous section discussed modeling a smart power grid, attack vector design, and the proposed method in detail. The required measurements are extracted in this section using IEEE standard data. The network topology matrix is

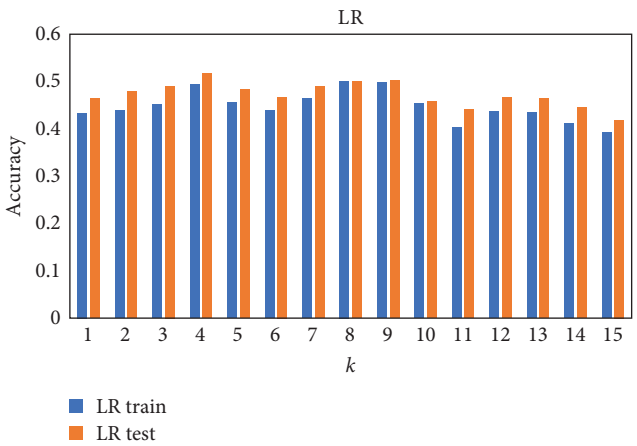


FIGURE 25: Detection accuracy of LR algorithm in different states of the attack vector.

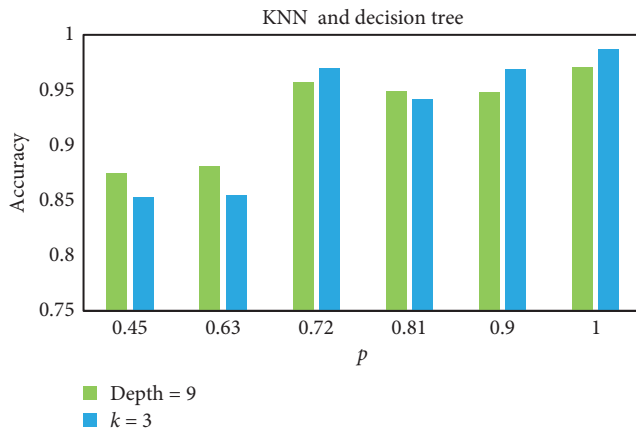


FIGURE 26: FDI targeted attack detection accuracy with two KNN algorithms and decision tree in terms of p .

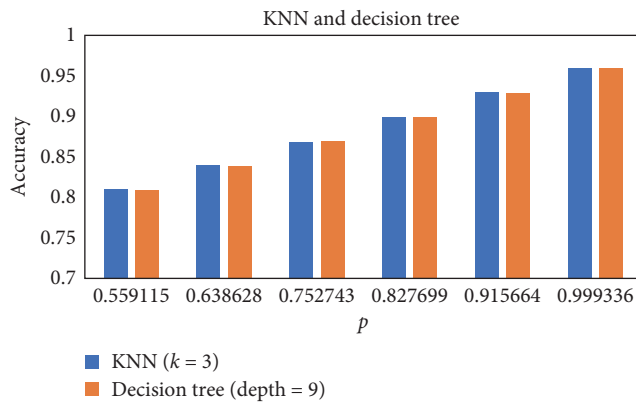


FIGURE 27: The accuracy of FDI targeted attack detection with two KNN algorithm and decision tree in terms of p .

calculated based on the desired network in MATLAB software, and the attack vector is injected into the measurements. Then, the FDIA is investigated and identified using the proposed methods. By comparing the two proposed algorithms with other commonly used methods in the classification field evaluated in this article, it was seen that the accuracy of these two algorithms is much more favorable than other methods. According to the obtained results, the diagram of the KNN algorithm with $k = 3$ and the decision tree algorithm with a depth of 9 are the best answers. As can be seen, the decision tree algorithm equals 0.45, 0.63, and the nearest neighbor algorithm equals 0.72, 0.9, 1 as the highest accuracy.

Figures 26 and 27 show the diagram of the KNN algorithm with $k = 3$ and the decision tree algorithm with a depth of 9. As can be seen, the decision tree algorithm with a depth of 9 in p is equal to 0.45 and 0.64, and the nearest neighbor algorithm with $k = 3$ in p is equal to 1.9 and 72.

According to the plotted graphs, it is quite clear that the detection accuracy increased with increasing value for both algorithms. This shows that the detectors are able to detect attacks, which cause more severe disturbances in the system.

9. Conclusion

The innovation of smart power grids brings huge challenges and initiatives in the industry, where the issue of security is the biggest concern of this technology. In smart power grids, there are two types of data such as physical measurements and state variables. This innovation is collected every 2–4 s and used to estimate the network status. From the investigations of the modeling of a smart electricity network and the design of the attack vector with the proposed method and based on the desired network, the network topology matrix was calculated in MATLAB software, and the attack vector was injected into the measurements. Then, using the proposed methods, it was investigated and identified the FDIA. By comparing the two proposed algorithms with other widely used methods in the field of classification that were evaluated in this article, it was seen that the accuracy of these two algorithms is much more favorable than other methods. According to the best depth for the decision tree algorithm and k for the KNN algorithm, it is drawn with $k = 3$ and the decision tree algorithm with a depth of 9. As can be seen, the decision tree algorithm with a depth of 9 in p is equal to 0.45 and 0.64, and the nearest neighbor algorithm with $k = 3$ in p is equal to 0.72, 1.9, and 0.64.

Data Availability

The data that support the findings of this study are available on request from the corresponding author.

Conflicts of Interest

The authors have declared that no conflicts of interest exist.

References

- [1] T. Samavat and M. Zand, "Artificial intelligence techniques for smart power systems," in *IoT and Analytics in Renewable Energy Systems*, vol. 1, pp. 107–123, CRC Press.
- [2] L. Tightiz and H. Yang, "A comprehensive review on IoT protocols' features in smart grid communication," *Energies*, vol. 13, no. 11, Article ID 2762, 2020.
- [3] P. Sanjeevikumar, T. Samavat, and M. Khoobani, "Machine learning-based hybrid demand-side controller for renewable energy management," in *Sustainable Developments by Artificial Intelligence and Machine Learning for Renewable Energies*, pp. 291–307, Elsevier, 2022.
- [4] L. Tightiz, L. M. Dang, and J. Yoo, "Novel deep deterministic policy gradient technique for automated micro-grid energy management in rural and islanded areas," *Alexandria Engineering Journal*, vol. 82, pp. 145–153, 2023.
- [5] A. A. Dashtaki, M. Khaki, M. Zand et al., "A day ahead electrical appliance planning of residential units in a smart home network using ITS-BF algorithm," *International Transactions on Electrical Energy Systems*, vol. 2022, Article ID 2549887, 13 pages, 2022.
- [6] D. Kim, S. Jeon, J. Shin, and J. T. Seo, "Design the IoT botnet defense process for cybersecurity in smart city," *Intelligent Automation & Soft Computing*, vol. 37, no. 3, pp. 2979–2997, 2023.
- [7] S. Padmanaban, M. A. Nasab, T. Samavat, M. A. Nasab, and M. Zand, "Securing smart power grids against cyber-attacks,"

- in *InIoT and Analytics in Renewable Energy Systems*, vol. 1, pp. 17–36, CRC Press, 2023.
- [8] M. Zand, “Securing smart power grids against cyber-attacks,” in *IoT and Analytics in Renewable Energy Systems*, vol. 1, pp. 17–36, CRC Press.
 - [9] E. Hashemi, “Cyber security in smart energy networks,” in *IoT and Analytics in Renewable Energy Systems*, pp. 309–325, CRC Press.
 - [10] S. Rastgoo and S. Padmanaban, “Using an intelligent control method for electric vehicle charging in microgrids,” *World Electric Vehicle Journal*, vol. 13, no. 12, Article ID 222, 2022.
 - [11] M. A. Nasab, W. K. Al-Shibli, M. Zand, B. Ehsan-Maleki, and S. Padmanaban, “Charging management of electric vehicles with the presence of renewable resources,” *Renewable Energy Focus*, vol. 48, Article ID 100536, 2024.
 - [12] S. Aoufi, A. Derhab, and M. Guerroumi, “Survey of false data injection in smart power grid: attacks, countermeasures and challenges,” *Journal of Information Security and Applications*, vol. 54, Article ID 102518, 2020.
 - [13] S. H. Mohammed, A. Al-Jumaily, M. S. J. Singh et al., “A review on the evaluation of feature selection using machine learning for cyber-attack detection in smart grid,” *IEEE Access*, vol. 12, pp. 44023–44042, 2024.
 - [14] A. A. Habib, M. K. Hasan, A. Alkhayyat, S. Islam, R. Sharma, and L. M. Alkwai, “False data injection attack in smart grid cyber physical system: issues, challenges, and future direction,” *Computers and Electrical Engineering*, vol. 107, Article ID 108638, 2023.
 - [15] Q. Ma, Z. Xu, W. Wang et al., “Dynamic load-altering attack detection based on adaptive fading Kalman filter in power systems,” *Global Energy Interconnection*, vol. 4, no. 2, pp. 184–192, 2021.
 - [16] M. Ghiasi, M. Dehghani, T. Niknam, A. Kavousi-Fard, P. Siano, and H. H. Alhelou, “Cyber-attack detection and cyber-security enhancement in smart DC-microgrid based on blockchain technology and Hilbert Huang transform,” *IEEE Access*, vol. 9, pp. 29429–29440, 2021.
 - [17] M. Irfan, A. Sadighian, A. Tanveer, S. J. Al-Naimi, and G. Oligeri, “A survey on detection and localisation of false data injection attacks in smart grids,” *IET Cyber-Physical Systems: Theory & Applications*, 2024.
 - [18] A. Ali, M. A. Khan, and H. Choi, “Supervised machine learning-based prediction of hydrogen storage classes utilizing dibenzyltoluene as an organic carrier,” *Molecules*, vol. 29, 2024.
 - [19] M. Ghiasi, T. Niknam, Z. Wang, M. Mehrandezh, M. Dehghani, and N. Ghadimi, “A comprehensive review of cyber-attacks and defense mechanisms for improving security in smart grid energy systems: past, present and future,” *Electric Power Systems Research*, vol. 215, Article ID 108975, 2023.
 - [20] K. D. Lu, L. Zhou, and Z. G. Wu, “Representation-learning-based CNN for intelligent attack localization and recovery of cyber-physical power systems,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, 2023.
 - [21] W. Jiang, W. Yang, J. Zhou, W. Ding, Y. Luo, and Y. Liu, “Reinforcement learning-based detection for state estimation under false data injection,” *IEEE Access*, vol. 9, pp. 66498–66508, 2021.
 - [22] F. Ünal, A. Almalaq, S. Ekici, and P. Glauner, “Big data-driven detection of false data injection attacks in smart meters,” *IEEE Access*, vol. 9, pp. 144313–144326, 2021.
 - [23] M. Cui, J. Wang, and B. Chen, “Flexible machine learning-based cyberattack detection using spatiotemporal patterns for distribution systems,” *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1805–1808, 2020.
 - [24] S. Mohammadi, F. Eliassen, Y. Zhang, and H. A. Jacobsen, “Detecting false data injection attacks in peer to peer energy trading using machine learning,” *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 5, pp. 3417–3431, 2022.
 - [25] V. Gomathi, N. Hemavathi, R. Valarmathi, K. Madhurima, C. Varshmitha, and L. Haarika, “Naive bayes based intrusion detection scheme for false data injection attack detection in smart grid,” in *International Conference on Emerging Research in Computational Science (ICERCS)*, pp. 1–6, Coimbatore, India, 2023.
 - [26] M. J. Abudin, S. Thokchom, R. T. Naayagi, and G. Panda, “Detecting false data injection attacks using machine learning-based approaches for smart grid networks,” *Applied Sciences*, vol. 14, no. 11, Article ID 4764, 2024.
 - [27] A. Y. Kermani, A. Abdollahi, and M. Rashidinejad, “Cyber-secure energy and flexibility scheduling of interconnected local energy networks with introducing an XGBoost-assisted false data detection and correction method,” *International Journal of Electrical Power & Energy Systems*, vol. 155, Article ID 109683, 2024.
 - [28] B. Chen, Q. H. Wu, M. Li, and K. Xiahou, “Detection of false data injection attacks on power systems using graph edge-conditioned convolutional networks,” *Protection and Control of Modern Power Systems*, vol. 8, no. 1, pp. 1–2, 2023.
 - [29] K.-D. Lu, Z.-G. Wu, and T. Huang, “Differential evolution-based three stage dynamic cyber-attack of cyber-physical power systems,” *IEEE/ASME Transactions on Mechatronics*, vol. 28, pp. 1137–1148, 2022.
 - [30] M. Dehghani, T. Niknam, M. Ghiasi, N. Bayati, and M. Savaghebi, “Cyber-attack detection in DC microgrids based on deep machine learning and wavelet singular values approach,” *Electronics*, vol. 10, no. 16, Article ID 1914, 2021.
 - [31] P. Zambare and Y. Liu, “Understanding cybersecurity challenges and detection algorithms for false data injection attacks in smart grids,” in *Internet of Things. Advances in Information and Communication Technology. IFIP IoT 2023. IFIP Advances in Information and Communication Technology*, eds. D. Puthal, S. Mohanty, and B. Y. Choi, vol. 684, Springer, Cham, 2024.