electronics

Article

A Hierarchical Federated Learning-Based Intrusion Detection System for 5G Smart Grids

Xin Sun [1], Zhijun Tang [1], Mengxuan Du [2], Chaoping Deng [1], Wenbin Lin [1], Jinshan Chen [1], Qi Qi [1] and Haifeng Zheng [2,*]

[1] State Grid Fujian Electric Power Research Institute, Fuzhou 350007, China
[2] College of Physics and Information Engineering, Fuzhou University, Fuzhou 350108, China
* Correspondence: zhenghf@fzu.edu.cn

Abstract: As the core component of smart grids, advanced metering infrastructure (AMI) provides the communication and control functions to implement critical services, which makes its security crucial to power companies and customers. An intrusion detection system (IDS) can be applied to monitor abnormal information and trigger an alarm to protect AMI security. However, existing intrusion detection models exhibit a low performance and are commonly trained on cloud servers, which pose a major threat to user privacy and increase the detection delay. To solve these problems, we present a transformer-based intrusion detection model (Transformer-IDM) to improve the performance of intrusion detection. In addition, we integrate 5G technology into the AMI system and propose a hierarchical federated learning intrusion detection system (HFed-IDS) to collaboratively train Transformer-IDM to protect user privacy in the core networks. Finally, extensive experimental results using a real-world intrusion detection dataset demonstrate that the proposed approach is superior to other existing approaches in terms of detection accuracy and communication cost for an IDS.

Keywords: smart grid; federated learning; intrusion detection system

Citation: Sun, X.; Tang, Z.; Du, M.; Deng, C.; Lin, W.; Chen, J.; Qi, Q.; Zheng, H. A Hierarchical Federated Learning-Based Intrusion Detection System for 5G Smart Grids. *Electronics* **2022**, *11*, 2627. https://doi.org/10.3390/electronics11162627

Academic Editor: Ahmed F. Zobaa

Received: 18 July 2022
Accepted: 19 August 2022
Published: 22 August 2022

1. Introduction

With the development of new technologies, the demand for electric energy in industry, manufacturing and people's lives has increased sharply. Unfortunately, a traditional power network is unable to cope with the increasing service requirements and power energy consumption due to its simple structure. Therefore, the smart grid composed of a power network and communication network was proposed to solve this problem, in which energy and information can flow between customers and power companies to provide various services. As the core component of the smart grid, advanced metering infrastructure (AMI) [1–3] has been mainly studied in recent years. AMI consists of three key components: smart meters, bidirectional communication links, and a cloud server (data center) for data aggregation [4]. It uses a bidirectional communication network to collect the energy consumption data or other information for analysis and processing [5] and then implements control measures, such as the remote control of household appliances [6], etc. The services will gradually increase with the requirements of users in the future. However, it will pose a huge challenge to the communication network as the amount of transmitted data increases. Fortunately, with the continuous development of fifth-generation wireless communication technology (5G), it has the ability to provide a communication network with large bandwidth, high transmission speed and low communication delay, making the combination of smart grid and 5G become a necessary development direction in the future [7]. Integrating wireless technologies into the smart grid will make the deployment of smart meters more flexible and services or applications of AMI more diverse. However, the uncertain environment, wide distribution and bidirectional communication networks will make the AMI system susceptible to attacks [8]. Attacks such as man-in-the-middle

(MITM) [9], jamming [10], false data injection (FDI) [11], eavesdropping [12], denial of service (DoS) [13], and message replay [14] can damage the confidentiality and availability of the AMI system. For example, hacking into smart meters compromises users' privacy, and intrusion into cloud servers causes system destruction and energy price manipulation. All attacks will seriously affect people's lives.

To protect the communication security of AMI system, intrusion detection system (IDS) has been widely studied. It can dynamically detect suspicious or abnormal behavior and trigger the alarm in time [15,16]. Therefore, it is essential to design an efficient and fast detection IDS to meet the requirements of AMI. With the development of artificial intelligence, the IDS based on artificial intelligence has been widely adopted to improve the ability to detect the IDS. In such cases, the cloud server collects a large amount of user data to train an intrusion detection machine learning (ML) model, which is then used to monitor the AMI system at the cloud server side. However, users' privacy may be violated, and their lives may be affected in this process since the personal data can directly reveal their living habits. Moreover, the detection delay will increase if the attack is near the users' side.

To address the data privacy problem, an innovative distributed machine learning technique called "federated learning (FL)" has been proposed [17], which is illustrated in Figure 1. In federated learning, multiple users train a shared model collaboratively by exchanging model parameters with cloud servers without directly sending raw data. Such a distributed and secure architecture has been widely adopted in many areas and greatly motivates us to design a distributed and privacy-protected intrusion detection system for AMI. However, since all participants can directly upload their local model to the cloud server, which may cause a great burden to the server and consume a lot of communication resources as the number of participants grows.

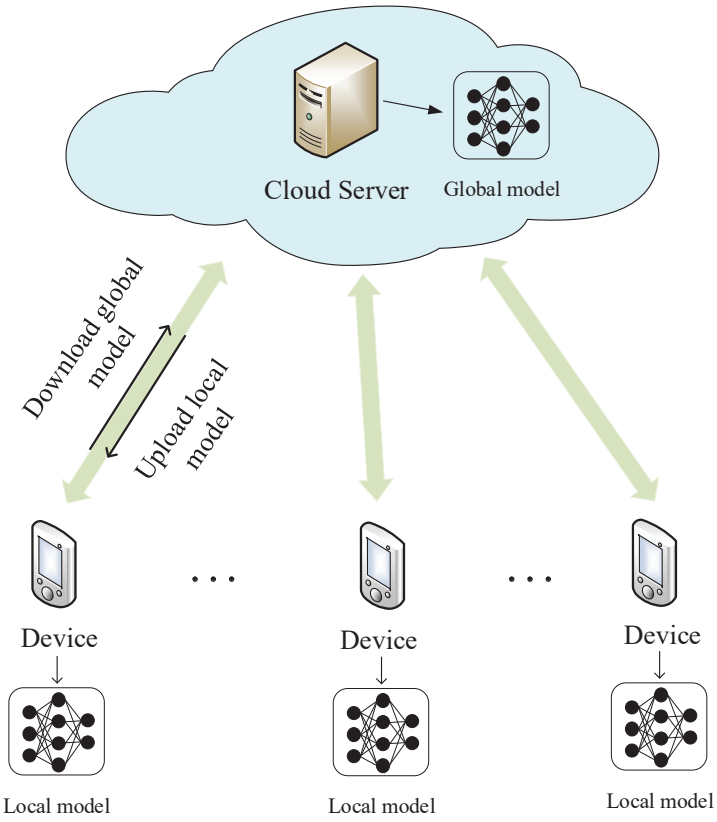

Figure 1. The framework of federated learning.

This paper aims to design a large-scale, reliable, efficient, distributed and privacy-protected AMI system. We integrate the 5G wireless communication technology into the AMI system and propose a 5G-based AMI system. Then, we propose a transformer-based

intrusion detection model (Transformer-IDM) to improve the detection performance and adopt a hierarchical federated learning in the 5G-based AMI system to collaboratively train a shared Transformer-IDM without compromising user privacy and reduce communication costs. The main contributions of this paper are presented as follows:

- We proposed a 5G-based AMI system, where the smart meters deployed in different locations can access 5G base stations to transmit data to the cloud server. We also propose a hierarchical federated learning-based approach in the AMI system to enable smart meters to train a shared intrusion detection model collaboratively while maintaining privacy and reducing communication costs.
- A transformer-based intrusion detection model is proposed to improve the detection performance. The proposed model adopts feature extraction layers to extract numerical features and leverage transformer layers to capture the relationship between categorical features, aiming to identify intrusion information effectively.
- We conduct extensive simulations to evaluate the performance of the proposed transformer-based intrusion detection model and its performance in federated learning. The results demonstrate that the proposed model has a better performance in different situations.

The rest of this paper is organized as follows. Related works are described in Section 2. The system model and the problem formulation is presented in Section 3. The proposed transformer-based intrusion detection model is presented in Section 4, and the results on detection performance are shown in Section 5. Finally, the conclusion is drawn in Section 6.

2. Related Work

As mentioned before, the security of the AMI system is essential to the smart grid to provide reliable power supply and services. Consequently, integrating intrusion detection methods into an AMI system is an inevitable requirement for a secure smart grid.

Recently, with the development of artificial intelligence, various machine learning-based methods have been applied in intrusion detection for AMI systems. Alseiari et al. [18] proposed a real-time IDS, which adopts the k-means clustering method to detect anomalous data flow. In [19], an ensemble learning method based on the XGBoost algorithm is proposed to improve the attack detection and identification accuracy. A multi-support vector machine (SVM) based IDS is developed by Vijayanand et al. [20] to detect the attacks occurring in an AMI system, where each SVM classifier is utilized to detect one specific attack. Camana et al. [21] proposed an extremely randomized tree-algorithm-based detection method to overcome the problem that false injection data vulnerably compromise the detector during state estimation. However, traditional machine learning methods easily cause overfitting and generally exhibit low accuracy. Thus, deep learning is becoming more efficient in AMI intrusion detection due to its strong generalization ability. Zheng et al. [22] proposed a wide and deep convolutional neural network (CNN) model to detect electricity theft in smart grids. The proposed CNN model comprises the wide component and the deep CNN component, which helps to improve memorization and generalization. In [23], an intrusion detection model based on CNN is proposed, in which the detection accuracy of the model is improved by converting original input data into two-dimensional data. Thirimanne et al. [24] proposed a real-time IDS based on a deep neural network (DNN) model, which is hosted on a web server to provide real-time intrusion detection.

The one-hot encoding approach is generally applied to convert categorical features of a data sample into numerical features since the deep-learning-based methods require numerical features as an input. However, this causes the input features vector to become a high dimensional and sparse vector, which will impact the performance of intrusion detection. Meanwhile, the above methods are based on a centralized framework, where the model training and intrusion detection are performed at a cloud server. Under the centralized framework, users' (smart meters) locally collected data can be easily accessed by the cloud server, which increases the privacy risk of users. In addition, the centralized framework also increases the detection delay if the attack is near the user.

Different from the above works, we propose a federated learning-based distributed IDS in a 5G AMI system. Each user collaboratively trains an intrusion detection model by only exchanging model parameters with the cloud server to protect the users' privacy. Then, users utilize the obtained intrusion detection model to monitor the attacks locally and trigger the alarm in time. In addition, we also propose a transformer-based intrusion detection model (Transformer-IDM), in which the numerical and categorical features are processed respectively to improve detection performance.

3. System Model and Problem Formulation

The essential infrastructure of the smart grid is advanced metering infrastructure (AMI) [25], which comprises three major components: smart meters, two-way communication links, and a data center deployed at the power company. The data collected by smart meters will be transmitted to the data center, i.e., the cloud server, for analysis and processing. Then, the smart meters carry out the corresponding response based on the decision of the cloud server. With the increasing requirement for services, low transmission latency and highly reliable communication links are critical to the quality of service of the AMI system. In order to address the challenge, we first introduce the proposed 5G-based AMI system. Then, within such an AMI system, we propose a federated learning (FL)-based intrusion detection system (IDS).

3.1. The 5G-Based Advanced Metering Infrastructure System

The 5G wireless communication promises low end-to-end latency and high transmission bandwidth, and the network slicing technology enhances network resource efficiency and provides differentiated quality of service (QoS) guarantees under a shared physical infrastructure. Thus, 5G technology can satisfy the requirements of an AMI system in a smart grid for reliable, fast, and highly connected communications, making it suitable for the communications infrastructure of the AMI system. This paper considers an AMI system over a 5G network, consisting of household appliances, smart meters, 5G base stations, edge servers and a cloud server. A smart meter acts as an intelligent sensing device to communicate with household appliances through Home Area Networks (HANs) to collect data, in which HAN is a network responsible for the communication between household appliances and smart meters using Wi-Fi or other short-distance communication technologies. The data received by smart meters are uploaded to the base station and transmitted to the cloud server via a 5G Core network for processing. More importantly, we consider an edge server deployed at the base station to preprocess the data received from smart meters to reduce the communication costs in 5G Core. The proposed 5G-based AMI system is illustrated in Figure 2.

3.2. Federated Learning-Based Intrusion Detection System and Problem Formulation

Within the proposed 5G-based AMI system, we consider a federated learning-based intrusion detection system (FL-IDS), where the smart meters deployed in different locations can collaboratively learn an efficient global intrusion detection ML model via exchanging model parameters with a cloud server. As a result, the data collected by smarter meters do not require to be transmitted to the cloud server for centralized processing, which can protect the privacy of power consumers and reduce communication costs. Then, the trained ML model can be used to monitor the attacks at smart meters directly and trigger the alarm in time to protect the AMI system.

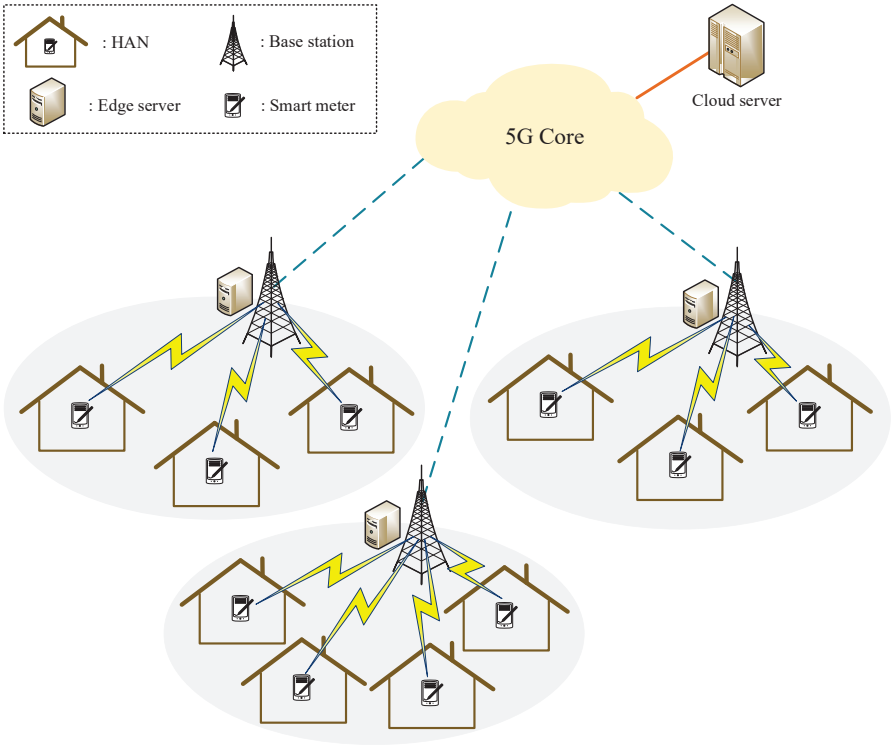

Figure 2. The 5G-based advanced metering infrastructure system.

We assume that the system consists of U 5G base stations $\{B_1, \ldots, B_U\}$ and N smart meters $\{S_1, \ldots, S_N\}$ to form U clusters in an area (in general, $N >> U$), where a cluster consists of one base station and several accessed smart meters. The clusters set is represented by $\{C_1, \ldots, C_U\}$, where $C_j (j \in \{1, \ldots, U\})$ is the participant set of B_j. In FL, smart meters S_i $(i \in (1, \ldots, N))$ can use their local data D_i to train a shared intrusion detection ML models $\boldsymbol{w}$ collaboratively. The goal of FL is to obtain an optimal global model $\boldsymbol{w}^*$, which satisfies:

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w} \in \mathbb{R}^d} \sum_{i=1}^{N} \frac{|D_i|}{|D|} F_i(\boldsymbol{w}) \tag{1}$$

where $|D_i|$ is the local dataset size of smart meter S_i, $|D| = \sum_{i=1}^{N} |D_i|$ is the total size of all local datasets, $F_i(\boldsymbol{w}) = \frac{1}{|D_i|} \sum_{(\boldsymbol{x}_n, y_n) \in D_i} l(\boldsymbol{w}; (\boldsymbol{x}_n, y_n))$ is the total loss function of smart meter S_i and $l(\boldsymbol{w}; (\boldsymbol{x}_n, y_n))$ is the loss based on one data sample $(\boldsymbol{x}_n, y_n)$. To solve the above optimization problem, the federated averaging (FedAvg) algorithm [26] can be adopted.

FedAvg algorithm mainly consists of local training and global aggregation. For the training process, participants commonly use their local dataset to perform gradient updates on their local models. Such a training process at smart meter S_i can be expressed as:

$$\boldsymbol{w}_i^{(t+1/2)} = \boldsymbol{w}_i^{(t)} - \eta \nabla F_i\left(\boldsymbol{w}_i^{(t)}\right), \tag{2}$$

where $\boldsymbol{w}_i^{(t)} = \boldsymbol{w}^{(t)}$ is the received global model at iteration t, which can also be regarded as the local model of smart meter S_i, $\nabla F_i\left(\boldsymbol{w}_i^{(t)}\right) = \frac{1}{|\mathcal{D}_i|} \sum_{(x_n, y_n) \in \mathcal{D}_i} \nabla l\left(\boldsymbol{w}_i^{(t)}; (x_n, y_n)\right)$ is the gradient of $\boldsymbol{w}_i^{(t)}$ and η is the learning rate. After the local gradient update, the trained local model $\boldsymbol{w}_i^{(t+1/2)}$ is collected by the cloud server to establish a global model by model aggregation. The aggregation strategy can be shown as follows:

$$\boldsymbol{w}^{(t+1)} = \sum_{i=1}^{N} \frac{|D_i|}{|D|} \boldsymbol{w}_i^{(t+1/2)}, \tag{3}$$

where the aggregation weight is related to the size of the local dataset, and the established global model $\boldsymbol{w}^{(t+1)}$ is distributed to all participants for the next iteration.

After multiple iterations, the optimal global model will finally be obtained. However, the traditional ML models, such as DNN models, CNN models, etc., are not efficient at serving as an intrusion detection ML model. Since the intrusion detection data usually have many categorical features, one-hot encoding approach should be employed to convert these categorical features into numerical features that can be fed into the traditional ML model. Nevertheless, this will increase the number of input features and lose the correlation information between categorical features. In addition, the communication costs between the base station and cloud server will increase with the number of participating smart meters, which brings a huge burden on the 5G Core. Therefore, we aim to design a novel neural network model to improve intrusion detection performance and adopt a hierarchical FL, which can reduce the communication costs in the core network.

4. Proposed Transformer-Based Intrusion Detection Model

Considering the dataset includes categorical features and numerical features, we design a neural network that utilizes the transformer [27] to perform contextual embeddings for categorical features and leverage the feature extraction network to extract the feature of numerical features. More details of the proposed transformer-based intrusion detection model (Transformer-IDM) are presented as follows.

4.1. The Transformer-IDM Model

The proposed Transformer-IDM consists of a feature extraction layer, a column embedding layer, a stack of N Transformer layers and a multi-layer perceptron. The architecture of Transformer-IDM is illustrated in Figure 3. In Transformer-IDM, the feature extraction layer consists of a convolutional layer and a pooling layer, and each transformer layer is composed of a multi-head self-attention layer and a feed-forward layer.

We assume that $(\boldsymbol{x}, y)$ is one data sample in the dataset, where $\boldsymbol{x} = \{\boldsymbol{x}^{num}, \boldsymbol{x}^{cate}\}$ is input features, which consist of numerical features $\boldsymbol{x}^{num} \in \mathbb{R}^{1\times L}$ and categorical features $\boldsymbol{x}^{cate}$, and y is the label of a data sample. Firstly, we use multiple feature extraction layers to process the numerical features $\boldsymbol{x}^{num}$, where each feature extraction layer contains two operations: 1D convolution and max-pooling. After a stack of feature extraction, the original numerical features $\boldsymbol{x}^{num}$ can be transformed to powerful compressed features $\boldsymbol{c}^{num} \in \mathbb{R}^{1\times l}$, where $l < L$. This process can be expressed as $\boldsymbol{c}^{num} = F_e(\boldsymbol{x}^{num})$, where $F_e(\cdot)$ is the feature extraction operation.

Next, for categorical features, let $\boldsymbol{x}^{cate} = \{x_1^{cate}, \cdots, x_r^{cate}\}$, where $x_i^{cate} (i \in \{1, \cdots, r\})$ is a categorical feature. We use the column embedding to embed each x_i^{cate} into m dimension; specifically, this process can be expressed as $E(\boldsymbol{x}^{cate}) = \{\boldsymbol{e}_1(x_1^{cate}), \cdots, \boldsymbol{e}_r(x_r^{cate})\}$, where $\boldsymbol{e}_i(x_i^{cate}) \in \mathbb{R}^{1\times m}$ $(i \in \{1, \cdots, r\})$ is the embedding of the x_i^{cate} and $E(\boldsymbol{x}^{cate})$ is the set of embeddings for all the categorical features. Then, the embedded categorical features $E(\boldsymbol{x}^{cate})$ are transformed to the input of transformer layer $\boldsymbol{I} = (\boldsymbol{e}_1(x_1^{cate}), \cdots, \boldsymbol{e}_r(x_r^{cate}))^T \in \mathbb{R}^{r\times m}$ and fed into a stack of transformer layers, where the output of the first transformer layer is inputted to the next transformer layer, and so forth. A transformer layer includes a multi-head self-attention layer followed by a position-wise feed-forward layer, with element-wise addition and layer-normalization being performed after each layer. There are three learnable parameters $W_Q \in \mathbb{R}^{m\times d_k}$, $W_K \in \mathbb{R}^{m\times d_k}$ and $W_V \in \mathbb{R}^{m\times d_v}$ in the multi-head self-attention layer. The input $\boldsymbol{I}$ is projected onto three matrices $K \in \mathbb{R}^{r\times d_k}$, $Q \in \mathbb{R}^{r\times d_k}$ and $V \in \mathbb{R}^{r\times d_v}$ as:

$$Q = IW_Q,\ K = IW_K,\ V = IW_V, \tag{4}$$

where m is the number of embedded features input to the transformer, and d_k and d_v denote the hidden dimension of multi-head attention, respectively. Next, we compute the matrix of the output of multi-head self-attention as:

$$\text{Attention}(K, Q, V) = A \cdot V, \tag{5}$$

where $A = \text{softmax}\left(\left(QK^T\right)/\sqrt{d_k}\right) \in \mathbb{R}^{r\times r}$ is the attention matrix that captures the similarity between input embeddings. Then the output of the attention head is projected back to the input dimension r through a feed-forward network. Consequently, through successive transformer layers, the embedded $E(\boldsymbol{x}^{cate})$ will be transformed into contextual embeddings, namely, the relationship between the categorical features can be greatly captured in the output of the last transformer layer.

Finally, the outputs from the feature extraction layer and transformer layer are concatenated to form a vector with $(l + d \times m)$ dimensions. This vector is input to a multi-layer perceptron so as to obtain the detection results.

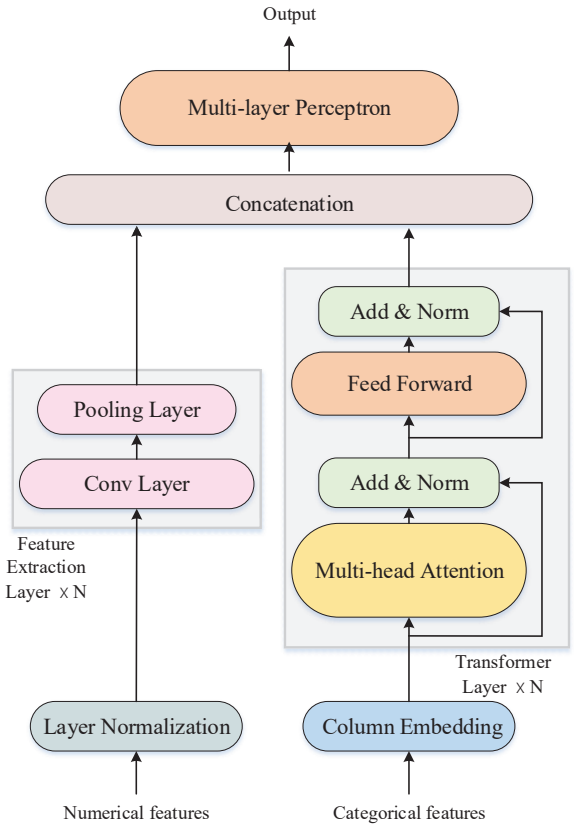

Figure 3. The architecture of Transformer-IDM.

4.2. Hierarchical Federated Learning-Based Intrusion Detection System

In the proposed hierarchical federated learning-based intrusion detection system (HFed-IDS), participating smart meters in different clusters can use their local data to collaboratively train a shared Transformer-IDM with the help of edge servers to reduce the traffic in the core network. The process is presented in Algorithm 1. At the beginning of HFed-IDS, participating smart meters receive a global model from the cloud server and perform H local gradient update epochs. Then, the trained model of the participant is uploaded to the base station. Different from the traditional FL, after the base station receives the model of all participants in the cluster, it directly aggregates the received models at the edge server to reduce the number of models that need to be transmitted to the cloud server. The aggregation process at the edge server of base station B_j can be formulated as:

$$\boldsymbol{W}_{B_j}^{(t+1/2)} = \sum_{i\in C_j} \frac{|D_i|}{\sum_{i\in C_j}|D_i|} \boldsymbol{w}_i^{(t+1/2)}, \tag{6}$$

where $\boldsymbol{W}_{B_j}^{(t+1/2)}$ is the established cluster model at the edge server of base station B_j, and C_j is the participant set of B_j. Then, each cluster model $\boldsymbol{W}_{B_j}^{(t+1/2)} (j \in \{1, \ldots, U\})$ is

transmitted to the cloud server for global aggregation, which can significantly reduce the communication costs compared to transmitting all participants' models to the cloud server for aggregation. The global aggregation can be expressed as:

$$\boldsymbol{w}^{(t+1)} = \sum_{j=1}^{U} \frac{\sum_{i \in C_j} |D_i|}{|D|} \boldsymbol{W}_{B_j}^{(t+1/2)}, \tag{7}$$

where the aggregation weight of cluster model $\boldsymbol{W}_{B_j}^{(t+1/2)}$ is related to the total dataset size in cluster C_j and $\boldsymbol{w}^{(t+1)}$ is the global model. Note that the obtained global model in (7) and (3) is the same since the aggregation weight in (6) and (7) is related to the size of the dataset. Thus, the proposed HFed-IDS not only has the same performance as the FedAvg algorithm but also can reduce communication costs. Finally, the obtained novel global model is sent back to all participants for the next iteration. Note that the raw data are not transmitted to the cloud server in HFed-IDS, so the user's privacy can be protected. In addition, the participating smart meters can finally obtain an efficient intrusion detection model, which makes it able to perform attack detection locally. Moreover, although the proposed HFed-IDS requires some edge servers to implement intermediate aggregation, it will be easily implemented as the edge server will be commonly deployed at the base station for edge computing applications in 5G networks [28].

Algorithm 1: HFed-IDS

Input: T : the total number of global communication iterations, η : the learning rate, H : the number of local gradient update epochs.
Output: $\boldsymbol{w}^{(T)}$

Server: initialize the global model $\boldsymbol{w}^{(0)}$, $\boldsymbol{w}^{(0)} = \boldsymbol{w}_1^{(0)} = \cdots = \boldsymbol{w}_N^{(0)}$ and send it to all participants; **for** $t = 0, \cdots, T$ **do**
  for $LocE = 0, \cdots, H$ **do**
    Each participant i in parallel performs $\boldsymbol{w}_{i,LocE}^{(t)} = \boldsymbol{w}_{i,LocE-1}^{(t)} - \eta \nabla F_i\left(\boldsymbol{w}_{i,LocE-1}^{(t)}\right)$;
  end
  $\boldsymbol{w}_i^{(t+1/2)} = \boldsymbol{w}_{i,H}^{(t)}$;
  All participants send $\boldsymbol{w}_i^{(t+1/2)}$ to the connected **base station**;
  for $j = 1, \cdots, U$ **do**
    $\boldsymbol{W}_{B_j}^{(t+1/2)} = \sum_{i \in C_j} \frac{|D_i|}{\sum_{i \in C_j} |D_i|} \boldsymbol{w}_i^{(t+1/2)}$
  end
  Send all cluster model $\boldsymbol{W}_{B_j}^{(t+1/2)}$ to **Server**;
  Server: establish global model: $\boldsymbol{w}^{(t+1)} = \sum_{i=1}^{N} \frac{|D_i|}{|D|} \boldsymbol{w}_i^{(t+1/2)}$;
  Server: send global model $\boldsymbol{w}^{(t+1)}$ to all participants, $\boldsymbol{w}^{(t+1)} = \boldsymbol{w}_1^{(t+1)} = \cdots = \boldsymbol{w}_N^{(t+1)}$;
end

5. Performance Evaluation

5.1. Dataset

In this section, we evaluate the performance of the proposed algorithm with NSL-KDD datasets [29]. The NSL-KDD dataset is an improved version of the KDD Cup 99 dataset [30] in which redundant data are removed to make the distribution of the dataset more balanced and reasonable. Moreover, this dataset contains a large number of attacks that cover all possible AMI attack types. As a result, it is one of the most widely used datasets for intrusion detection in smart grids.

In the NSL-KDD dataset, one piece of the data sample has 41 features and 1 label, in which features can be divided into two types, i.e., categorical features and numerical features. As a result, each data sample has three categorical features and 38 numerical features, as shown in Table 1. In addition, there are several subtypes of attacks, i.e., labels of data samples, in the NSL-KDD dataset, which can be classified into five main categories: normal, denial of service (DoS), user to root (U2R), remote to local (R2L), and probing scanning (Probe). Table 2 shows the types of attacks and their main categories. The details of these attack categories are described in Table 3. In order to make a fair comparison, 50% of the training data was randomly selected from the NSL-KDD dataset as the model training set, and the remaining data were used as the testing set. Table 4 shows the distribution of the dataset. We can find that the quantity of R2L and U2R is relatively small in the training and testing set, which poses significant challenges to the learning and detection abilities of the model.

Table 1. The type of features.

Type of Features	Features
Categorical features	'protocol_type', 'service', 'flag'
Numerical features	'duration', 'src_bytes', 'dst_bytes', 'land', 'logged_in', 'is_host_login', 'is_guest_login', 'wrong_fragment', 'urgent', 'hot', 'num_failed_logins', 'num_compromised', 'root_shell', 'su_attempted', 'num_root', 'num_file_creations', 'num_shells', 'num_access_files', 'num_outbound_cmds', 'count', 'srv_count', 'serror_rate', 'srv_serror_rate', 'rerror_rate', 'srv_rerror_rate', 'same_srv_rate', 'diff_srv_rate', 'srv_diff_host_rate', 'dst_host_count', 'dst_host_srv_count', 'dst_host_same_srv_rate', 'dst_host_diff_srv_rate', 'dst_host_same_src_port_rate', 'dst_host_srv_diff_host_rate', 'dst_host_serror_rate', 'dst_host_srv_serror_rate', 'dst_host_rerror_rate', 'dst_host_srv_rerror_rate'

Table 2. The main categories and their contained attacks.

Dos	Probe	R2L	U2R	Normal
back land Neptune pod smurf teardrop	ipsweep nmap portsweep satan	spy warezclient ftpwrite guesspasswd imap multihop phf warezmaster	bufferoverflow loadmodule perl rootkit	normal

Table 3. The description of attacks.

Attack Categories	Description
Denial of service (DoS)	This attack occupies too many computing or memory resources so that the machine cannot handle legitimate requests and access.
Probing scanning (Probe)	This attack gathers information about potential vulnerabilities of the target system that can be used to launch attacks lately.
Remote to local (R2L)	Attacker does not have access to the victim's machine, and hence tries to gain local access as a user of that machine.
User to root (U2R)	Using this attack, attackers access the system as a normal user and exploit some vulnerability to gain root access to the system.

Table 4. The distribution of the dataset.

Main Categories	NSL KDD	Training Set	Testing Set
Normal	67,343	33,732	33,611
Dos	45,927	22,821	23,106
Probe	11,656	5933	5723
R2L	995	473	522
U2R	52	27	25
Total	125,973	62,986	62,987

5.2. Dataset Preprocessing

Each data sample contains both categorical and numerical features. However, since the neural network can not process categorical features, the dataset must be preprocessed. For conventional neural networks, such as CNN and DNN, the categorical features are usually converted to a numerical vector by one-hot processing. For example, the feature 'protocol_type' has three attributes, namely the transmission control protocol (TCP), user datagram protocol (UDP), and internet control message protocol (ICMP). After one-hot processing, they can be respectively represented by three vectors (0, 0, 1), (0, 1, 0), and (1, 0, 0), and the dimensions of vectors are 1×3. Unfortunately, 'service' contains 70 attributes. Therefore, the dimensions of vectors obtained after one-hot processing are too large to represent attributes effectively. However, in the proposed approach, one-hot processing is not adopted. Instead, the categorical features are all embedded into m dimensions embeddings, which can reduce the input features. In addition, since the numerical features with large values may mislead the learning model if it is used directly, the numerical features must be normalized to eliminate the influence of substantial feature differences. The following method can be used to normalize the numerical features:

$$x_i = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}, \tag{8}$$

where x_i is the original value, and $x_{\min}$ and $x_{\max}$ represent the minimum and maximum value of the feature, respectively.

5.3. Evaluation Metrics

In IDS, the evaluation metrics include accuracy (ACC), precision (P), detection rate (DR) and F-measure (F). ACC is the most important metric for evaluating the performance

of the IDS network and is described as the percentage of the number of correctly classified samples over total samples, which is shown in the following equation:

$$ACC = \frac{TP + TN}{TP + FP + TN + FN}, \tag{9}$$

where true positive (TP) refers to the number of correctly identified abnormal samples, true negative (TN) represents the number of correctly identified normal samples, false positive (FP) represents the number of incorrectly identified abnormal samples, and false negative (FN) represents the number of incorrectly identified normal samples. In addition, P is defined as the percentage of the number of correctly identified abnormal samples over the total number of abnormal samples predicted by the model and can be calculated as:

$$P = \frac{TP}{TP + FP}. \tag{10}$$

DR is defined as the ratio of the number of correctly identified abnormal samples over the total true number of abnormal samples and can be calculated as:

$$DR = \frac{TP}{TP + FN}. \tag{11}$$

F is defined as the average harmonic mean of P and DR, and its specific calculation can be calculated as:

$$F = \frac{2 * P * DR}{P + DR}. \tag{12}$$

5.4. Setup

In this simulation, the proposed Transformer-IDM contains two feature exaction layers and two transformer layers. The number of attention heads in the multi-head attention layer is eight, and the multi-layer perceptron before the output and the feed-forward network are fully connected layers. The embedding dimensions of the categorical feature are eight, the optimizer is Adam, and the learning rate was set to 0.0005. According to the above settings, we compare the performance of different models in which a multi-classification problem is considered for models. Python 3.7 and TensorFlow 2.8.0 were used to build different models on a computer equipped with an Intel Core i7-8700 CPU, 8 GB memory and a single NVIDIA GeForce RTX2080Ti GPU.

5.5. Numerical Results

First, we compare the performance of the proposed Transformer-IDM with a support vector machine (SVM) [31], linear regression (LR), K-nearest neighbors (KNN) [32], multinomial Naive Bayes (MultinomialNB) [33], DNN-3 [34], DNN-16 [24] and GRU+MLP [35] without federated learning. Namely, we train the model using the total training dataset and evaluate its performance using the total testing dataset without model exchanging.

In order to compare the performance of these models, we use the metrics P, DR, F and ACC to evaluate the performance of different models. The metric P reflects the confidence of attack detection, DR represents the ability of the model to identify attacks and F combines P with DR, which is also a useful metric to reflect the ability of models. Firstly, we calculate the metrics P, DR, and F of each attack category for the proposed Transform-IDM and the other models, and the results are shown in Table 5. Note that the results for neural-network-based models are obtained after 100 training epochs, and the same Adam optimizer is adopted to train these neural-network-based models for a fair comparison. Then, in Table 6, we compare all evaluation metrics of different models to illustrate the overall performance of different models.

From Table 5, we find that the Transformer-IDM has a higher P, DR and F for each attack category, which indicates that the proposed Transformer-IDM has better performance as an intrusion detection model. Moreover, we also notice that the evaluation metrics of

R2L and U2R for all models are low. The reason is that the quantity of R2L and U2R in the training set is relatively small, which will impact models to learn the features of R2L and U2R, resulting in a decreased performance. In Table 6, we can also observe that the proposed Transformer-IDM has a higher P, DR, F and ACC, which also demonstrates the superiority of Transformer-IDM as an intrusion detection model.

Table 5. The values of evaluation metrics of each attack category for different models.

Models	Evaluation Metrics (%)											
	P				***DR***				***F***			
	DoS	**Probing**	**R2L**	**U2R**	**DoS**	**Probing**	**R2L**	**U2R**	**DoS**	**Probing**	**R2L**	**U2R**
SVM	99.04	96.63	80.33	64.28	98.21	95.05	55.56	35.00	98.63	95.83	65.69	46.15
LR	99.08	97.28	79.35	72.73	97.89	95.70	75.09	32.00	98.48	96.48	77.16	44.44
KNN	98.77	98.68	87.16	61.54	99.45	96.69	67.62	32.00	99.11	97.67	76.16	42.11
MultinomialNB	97.72	56.31	26.63	16.36	84.24	91.32	31.23	36.00	90.48	96.67	28.75	22.50
DNN-3	99.73	98.06	79.33	15.79	98.38	97.45	72.80	12.00	99.03	97.76	75.92	13.63
GRU+MLP	99.30	97.53	78.45	11.76	98.05	96.54	54.41	8.00	98.67	97.03	64.25	9.52
DNN-16	99.55	98.61	81.79	18.75	99.46	98.74	55.94	12.00	99.51	98.67	66.44	14.63
Transformer-IDM	99.85	98.76	87.21	78.57	99.91	98.85	87.55	44.00	99.88	98.81	87.38	56.41

Table 6. The values of evaluation metrics.

Models	***P*** **(%)**	***DR*** **(%)**	***F*** **(%)**	***ACC*** **(%)**
SVM	97.76	97.8	97.76	97.81
LR	97.94	97.95	97.94	97.95
KNN	98.76	98.79	98.76	98.79
MultinomialNB	91.09	88.65	89.31	88.65
DNN-3	98.48	98.49	98.48	98.5
GRU+MLP	97.98	98.04	97.99	98.05
DNN-16	98.86	98.91	98.86	98.92
Transformer-IDM	99.49	99.49	99.49	99.48

Next, we evaluate the performance of neural-network-based models, i.e., DNN-3, GRU+MLP, DNN-16 and Transformer-IDM, with federated learning. Specifically, participants in federated learning use their local dataset to train a shared global model through exchanging model parameters with a cloud server. In this simulation, we assume that there are 10 smart meters participating in federated learning, in which smart meters are divided into three clusters shown in Figure 2. The training set is uniformly and randomly divided into 10 sub-datasets with the same size, and these sub-datasets are assigned to each participant as a local dataset to train the model. In order to evaluate the performance of the obtained global model, we still adopt the total testing set to test the performance. In addition, in our federated learning setting, each participant will send the trained model to the cloud server for aggregation after performing 10 local gradient update epochs.

In Figure 4, we plot the test accuracy of different global models in 100 global iterations, namely, 100 communication interactions between participants and the cloud server. It illustrates that when the proposed Transformer-IDM is used as the shared global model in federated learning, the test accuracy of the global model rises rapidly and can achieve the highest test accuracy after 100 global iterations. Table 7 shows the values of P, DR, and F of each attack category for different global models obtained after 100 global iterations. We can find that the Transformer-IDM obtained in federated learning (Fed-Transformer-IDM) also has a higher P, DR and F for each attack category. In Table 8, we compare all evaluation metrics of different global models obtained after 100 global iterations, in which Fed-Transformer-IDM still has a higher P, DR, F and ACC. According to the above results, it can be concluded that the proposed Transformer-IDM has the ability to act as an intrusion detection model in federated learning. In addition, in Table 8, we also present

the total parameters of different global models. Among them, DNN-3 and DNN-16 have 3 and 16 hidden layers, respectively, which causes DNN-16 to have the largest parameters. In GRU+MLP, since the GRU layer increases the complexity of the model, it has more model parameters than DNN-3. The proposed Transformer-IDM has the fewest model parameters since it has the tiny transformer layer and feature exacted layer, which can reduce communication costs in federated learning.

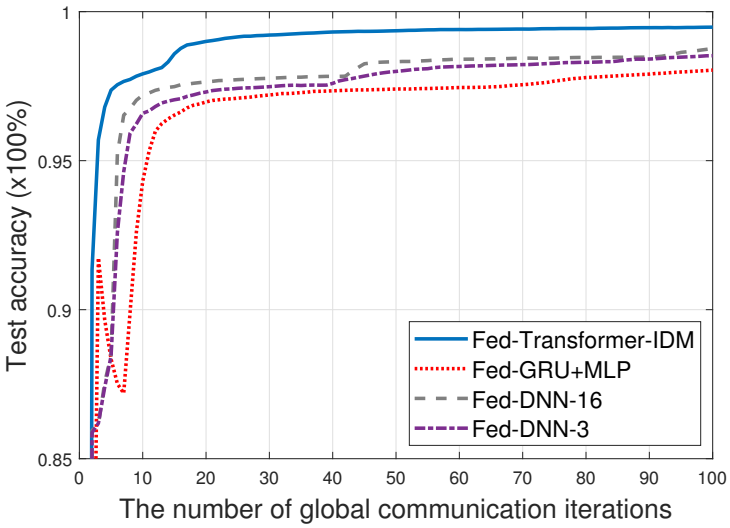

Figure 4. The test accuracy comparison at each global iteration.

Table 7. The values of evaluation metrics of each attack category for different global models.

Models	Evaluation Metrics (%)											
	P				*DR*				*F*			
	DoS	**Probing**	**R2L**	**U2R**	**DoS**	**Probing**	**R2L**	**U2R**	**DoS**	**Probing**	**R2L**	**U2R**
Fed-DNN-3	99.74	97.99	81.34	21.05	98.33	97.39	74.33	16.00	99.03	97.69	77.68	18.18
Fed-GRU+MLP	99.39	97.51	79.82	20.00	97.94	96.37	52.29	16.00	98.66	96.93	63.19	17.78
Fed-DNN-16	99.75	98.57	80.25	25.00	98.70	98.62	72.41	16.00	99.22	98.59	76.13	19.51
Fed-Transformer-IDM	99.87	98.86	87.36	80.00	99.91	98.64	87.36	48.00	99.89	98.75	87.36	60.00

Table 8. The values of evaluation metrics for different global models.

Models	*P* (%)	*DR* (%)	*F* (%)	*ACC* (%)	Total_Params
Fed-DNN-3	98.50	98.51	98.5	98.51	26,935
Fed-GRU+MLP	97.98	98.03	97.97	98.03	105,557
Fed-DNN-16	98.76	98.78	98.76	98.78	636,037
Fed-Transformer-IDM	99.49	99.49	99.49	99.49	20,503

In order to compare the size of transmitted data of different models in federated learning, we plot the total size of transmission data in the uplink (from smart meters to a base station) versus test accuracy in Figure 5, in which a quantization operation is adopted where each transmitted model parameter is encoded and quantized into 32 bits. The results demonstrated that the proposed Transformer-IDM not only has a higher test accuracy but also requires fewer transmission costs than the other models. We also compare the size of transmitted data from the base station to the cloud server in HFed-IDS and the traditional FL. Note that, in traditional FL, the local model of each smart meter is directly transmitted to the cloud server for global aggregation without the aid of the edge server. In addition, we set the same number of participating smart meters and data distributions as in the above simulation for HFed-IDS and traditional FL. Figure 6 plots the total size of uploaded data of different FL algorithms in the core network versus the iteration rounds, in which the proposed Transformer-IDM is used as the global model in different FL algorithms. We can observe that the proposed HFed-IDS can reduce the size of transmitted data in the core

network since the local model of the smart meter is firstly aggregated at the edge server to reduce the total number of models transmitted to the cloud server.

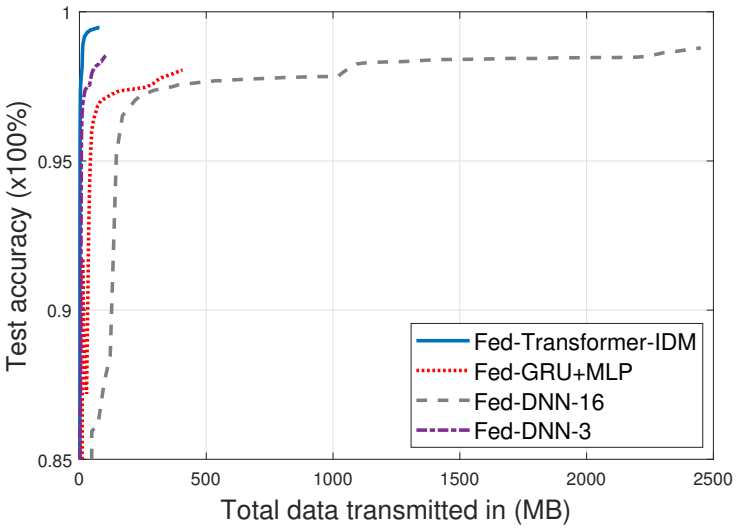

Figure 5. The total size of transmission data in the uplink versus the test accuracy of different global models.

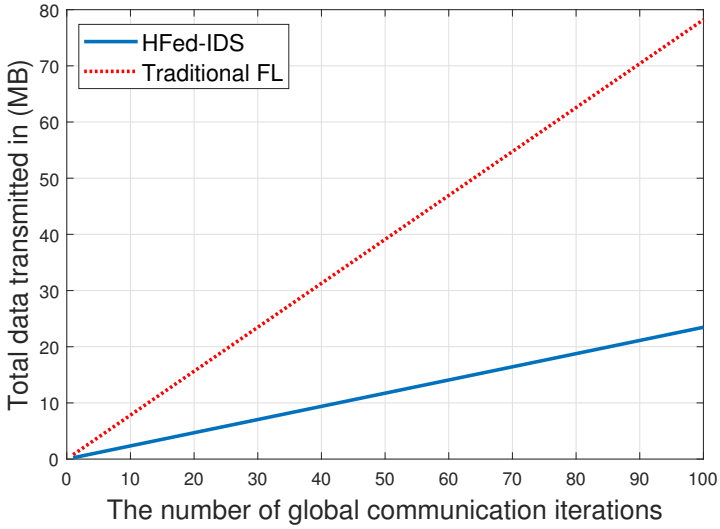

Figure 6. The size of transmitted data in the core network at each iteration.

6. Conclusions and Future Works

In this paper, we applied 5G communication technology to smart grids to provide a fast and reliable AMI system. In order to protect the security of the AMI system, we proposed a transformer-based intrusion detection model (Transformer-IDM), which leverages the transformer layer and feature exaction layer to process categorical features and numerical features, respectively, to improve the detection performance. Then, within the 5G-based AMI system, we proposed a hierarchical federated learning-based intrusion detection system (HFed-IDS). The participants collaboratively train a Transformer-IDM by exchanging the model parameters with the cloud server without compromising privacy. More importantly, in HFed-IDS, the upload model from smart meters will be aggregated at the edge server to reduce the size of transmitted data in the core network. After the federated learning, smart meters can use the model obtained by participating in federal learning to carry out intrusion detection at the user side to achieve a low-delayed detection. Finally, we use the real-world intrusion detection dataset NSL-KDD to evaluate the proposed model. The experimental results demonstrated that the proposed Transformer-IDM

performs better and can be used as an intrusion detection model. It is worth noting that the resource heterogeneity in federated learning, such as the varying storage, computational, and communication capabilities of each device, may affect the performance of the global model. Thus, addressing these problems will be a key focus of our future works. Furthermore, datasets covering more attack types can be used to evaluate the performance of the proposed method in our future work.

Author Contributions: Conceptualization, X.S.; data curation, Z.T.; investigation, J.C.; methodology, C.D.; project administration, W.L.; validation, Q.Q.; writing—original draft, M.D.; writing—review and editing, H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by State Grid Fujian Electric Power Limited Company under Grant (No. 521304210003).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sun, C.C.; Cardenas, D.J.S.; Hahn, A.; Liu, C.C. Intrusion detection for cybersecurity of smart meters. *IEEE Trans. Smart Grid* **2020**, *12*, 612–622. [CrossRef]
2. Zanetti, M.; Jamhour, E.; Pellenz, M.; Penna, M.; Zambenedetti, V.; Chueiri, I. A tunable fraud detection system for advanced metering infrastructure using short-lived patterns. *IEEE Trans. Smart Grid* **2017**, *10*, 830–840. [CrossRef]
3. Radoglou-Grammatikis, P.I.; Sarigiannidis, P.G. Securing the smart grid: A comprehensive compilation of intrusion detection and prevention systems. *IEEE Access* **2019**, *7*, 46595–46620. [CrossRef]
4. Ghorbanian, M.; Dolatabadi, S.H.; Masjedi, M.; Siano, P. Communication in smart grids: A comprehensive review on the existing and future communication and information infrastructures. *IEEE Syst. J.* **2019**, *13*, 4001–4014. [CrossRef]
5. Alahakoon, D.; Yu, X. Smart electricity meter data intelligence for future energy systems: A survey. *IEEE Trans. Ind. Inform.* **2015**, *12*, 425–436. [CrossRef]
6. Das, H.; Saikia, L. GSM enabled smart energy meter and automation of home appliances. In Proceedings of the 2015 International Conference on Energy, Power and Environment: Towards Sustainable Growth (ICEPE), Shillong, India, 12–13 June 2015; pp. 1–5.
7. Dragičević, T.; Siano, P.; Prabaharan, S. Future generation 5G wireless networks for smart grid: A comprehensive review. *Energies* **2019**, *12*, 2140.
8. Pedramnia, K.; Rahmani, M. Survey of DoS Attacks on LTE infrastructure used in AMI System and Countermeasures. In Proceedings of the 2018 Smart Grid Conference (SGC), Sanandaj, Iran, 28–29 November 2018; pp. 1–6.
9. Wlazlo, P.; Sahu, A.; Mao, Z.; Huang, H.; Goulart, A.; Davis, K.; Zonouz, S. Man-in-the-middle attacks and defense in a power system cyber-physical testbed. *arXiv* **2021**, arXiv:2102.11455.
10. Algin, R.; Tan, H.O.; Akkaya, K. Mitigating selective jamming attacks in smart meter data collection using moving target defense. In Proceedings of the 13th ACM Symposium on QoS and Security for Wireless and Mobile Networks, Miami, FL, USA, 21–25 November 2017; pp. 1–8.
11. Tufail, S.; Batool, S.; Sarwat, A.I. False data injection impact analysis in ai-based smart grid. In Proceedings of the Southeast Con 2021, Atlanta, GA, USA, 10–13 March 2021; pp. 1–7.
12. Chaudhry, J.; Qidwai, U.; Miraz, M.H. Securing big data from eavesdropping attacks in scada/ics network data streams through impulsive statistical fingerprinting. In Proceedings of the International Conference for Emerging Technologies in Computing, London, UK, 19–20 August 2019; pp. 77–89.
13. Liu, S.; Liu, X.P.; El Saddik, A. Denial-of-service (DoS) attacks on load frequency control in smart grids. In Proceedings of the 2013 IEEE PES Innovative Smart Grid Technologies Conference (ISGT), Washington, DC, USA, 24–27 February 2013; pp. 1–6.
14. Zhao, J.; Wang, J.; Yin, L. Detection and control against replay attacks in smart grid. In Proceedings of the 2016 12th International Conference on Computational Intelligence and Security (CIS), Wuxi, China, 16–19 December 2016; pp. 624–627.
15. Liu, H.; Lang, B. Machine learning and deep learning methods for intrusion detection systems: A survey. *Appl. Sci.* **2019**, *9*, 4396. [CrossRef]
16. Li, Y.; Xue, W.; Wu, T.; Wang, H.; Zhou, B.; Aziz, S.; He, Y. Intrusion detection of cyber physical energy system based on multivariate ensemble classification. *Energy* **2021**, *218*, 119505. [CrossRef]
17. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Ft. Lauderdale, FL, USA, 20–22 April 2017; pp. 1273–1282.
18. Alseiari, F.A.A.; Aung, Z. Real-time anomaly-based distributed intrusion detection systems for advanced Metering Infrastructure utilizing stream data mining. In Proceedings of the 2015 International Conference on Smart Grid and Clean Energy Technologies (ICSGCE), Offenburg, Germany, 20–23 October 2015; pp. 148–153.

19. Hu, C.; Yan, J.; Wang, C. Advanced cyber-physical attack classification with extreme gradient boosting for smart transmission grids. In Proceedings of the 2019 IEEE Power & Energy Society General Meeting (PESGM), Atlanta, GA, USA, 4–8 August 2019; pp. 1–5.
20. Vijayanand, R.; Devaraj, D.; Kannapiran, B. Support vector machine based intrusion detection system with reduced input features for advanced metering infrastructure of smart grid. In Proceedings of the 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 6–7 January 2017; pp. 1–7.
21. Acosta, M.R.C.; Ahmed, S.; Garcia, C.E.; Koo, I. Extremely randomized trees-based scheme for stealthy cyber-attack detection in smart grid networks. *IEEE Access* **2020**, *8*, 19921–19933. [CrossRef]
22. Zheng, Z.; Yang, Y.; Niu, X.; Dai, H.N.; Zhou, Y. Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids. *IEEE Trans. Ind. Inform.* **2017**, *14*, 1606–1615. [CrossRef]
23. Liu, G.; Zhang, J. CNID: Research of network intrusion detection based on convolutional neural network. *Discret. Dyn. Nat. Soc.* **2020**, *2020*, 4705982. [CrossRef]
24. Thirimanne, S.P.; Jayawardana, L.; Yasakethu, L.; Liyanaarachchi, P.; Hewage, C. Deep Neural Network Based Real-Time Intrusion Detection System. *SN Comput. Sci.* **2022**, *3*, 145. [CrossRef]
25. Liu, Y.; Yang, X.; Wen, W.; Xia, M. Smarter Grid in the 5G Era: Integrating Power Internet of Things with Cyber Physical System. *Front. Commun. Netw.* **2021**, *2*, 23. [CrossRef]
26. Konečnỳ, J.; McMahan, H.B.; Yu, F.X.; Richtárik, P.; Suresh, A.T.; Bacon, D. Federated learning: Strategies for improving communication efficiency. *arXiv* **2016**, arXiv:1610.05492.
27. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, 6000–6010.
28. Mao, Y.; You, C.; Zhang, J.; Huang, K.; Letaief, K.B. A Survey on Mobile Edge Computing: The Communication Perspective. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 2322–2358. [CrossRef]
29. NSL-KDD Dataset. Available online: http://nsl.cs.unb.ca/nsl-kdd/ (accessed on 1 June 2022).
30. Tavallaee, M.; Bagheri, E.; Lu, W.; Ghorbani, A.A. A detailed analysis of the KDD CUP 99 data set. In Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, ON, Canada, 8–10 July 2009; pp. 1–6.
31. Tian, Y.; Mirzabagheri, M.; Bamakan, S.M.H.; Wang, H.; Qu, Q. Ramp loss one-class support vector machine; a robust and effective approach to anomaly detection problems. *Neurocomputing* **2018**, *310*, 223–235. [CrossRef]
32. Serpen, G.; Aghaei, E. Host-based misuse intrusion detection using PCA feature extraction and kNN classification algorithms. *Intell. Data Anal.* **2018**, *22*, 1101–1114. [CrossRef]
33. Panda, M.; Abraham, A.; Patra, M.R. Discriminative multinomial naive bayes for network intrusion detection. In Proceedings of the 2010 Sixth International Conference on Information Assurance and Security, Atlanta, GA, USA, 23–25 August 2010.
34. Mirzaee, P.H.; Shojafar, M.; Pooranian, Z.; Asefy, P.; Cruickshank, H.; Tafazolli, R. FIDS: A Federated Intrusion Detection System for 5G Smart Metering Network. In Proceedings of the 2021 17th International Conference on Mobility, Sensing and Networking (MSN), Exeter, UK, 13–15 December 2021.
35. Xu, C.; Shen, J.; Du, X.; Zhang, F. An intrusion detection system using a deep neural network with gated recurrent units. *IEEE Access* **2018**, *6*, 48697–48707. [CrossRef]