A Smart Grid AMI Intrusion Detection Strategy Based on Extreme Learning Machine

Ke Zhang¹,²,*, Zhi Hu³, Yufei Zhan⁴, Xiaofen Wang¹ and Keyi Guo⁵

¹ School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China; xfwang@uestc.edu.cn
² Science and Technology on Eletronic Information Control Laboratory, Chengdu 610000, China
³ School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China; 201722010740@std.uestc.edu.cn
⁴ Glasgow College, University of Electronic Science and Technology of China, Chengdu 611731, China; 201720060327@std.uestc.edu.cn
⁵ Courant Institute of Mathematical Science, New York University, New York, NY 10003, USA; keyi.guo@nyu.edu
* Correspondence: kezhang@uestc.edu.cn

Received: 1 August 2020; Accepted: 16 September 2020; Published: 18 September 2020

Abstract: The smart grid is vulnerable to network attacks, thus requiring a high detection rate and fast detection speed for intrusion detection systems. With a fast training speed and a strong model generalization ability, the extreme learning machine (ELM) perfectly meets the needs of intrusion detection of the smart grid. In this paper, the ELM is applied to the field of smart grid intrusion detection. Aiming at the problem that the randomness of input weights and hidden layer bias in the ELM cannot guarantee the optimal performance of the ELM intrusion detection model, a genetic algorithm (GA)-ELM algorithm based on a genetic algorithm (GA) is proposed. GA is used to optimize the input weight and hidden layer bias of the ELM. Firstly, the input weight and hidden layer bias of the ELM are mapped to the chromosome vector of a GA, and the test error of the ELM model is set as the fitness function of the GA. Then, the parameters of the ELM intrusion detection model are optimized by genetic operation; the input weight and bias, corresponding to the minimum test error, are selected to improve the performance of the ELM model. Compared with the ELM and online sequential extreme learning machine (OS-ELM), the GA-ELM effectively improves the accuracy, detection rate and precision of intrusion detection and reduces the false positive rate and missing report rate.

Keywords: smart grid; advanced metering infrastructure (AMI); extreme learning machine (ELM); intrusion detection system (IDS)

1. Introduction

With the passage of time, the smart grid has made great progress in both technical and practical levels, and the ensuing smart grid security issues have also attracted more and more attention. If the smart grid is attacked, people’s lives will be seriously affected. For example, in 2009, the smart meter system of a US power grid company was attacked by hackers, resulting in enormous economic losses [1]. In 2010, Stuxnet targeted a number of energy facilities, hitting the industry hard [2]. In 2015, a power system in Ukraine was targeted by a Denial of Service (DoS) network attack, resulting in a large-scale blackout in the region [3]. It is obvious that smart grids bring convenience, as well as new challenges, to society; hence, targeted research must be carried out. Based on this, the United States, Europe and many other countries, as well as China, have successively carried out some researches in smart grid security and explored the field of smart grid intrusion detection.
The smart grid is composed of two parts: power distribution networks and communication networks. The communication network is responsible for communication between various devices [4]. Since the communication network is often connected to a computer, it brings many new security issues. Among them, the advanced metering infrastructure (AMI) is a very important component of the whole smart grid security; as a result, its network and physical security are extremely critical.

In recent years, the AMI has developed vigorously [5,6]. It is a system based on a network that connects power companies and customers, collects consumer’s data and other information and implements necessary control measures [7]. An AMI can provide info platforms and technical support for advanced applications, such as demand response management, real-time two-way interaction, distributed energy generation and storage. These actions rely heavily on the information infrastructure and communication networks, while those are also the most vulnerable. Various threats exist, such as energy theft and fraud, theft of sensitive information, service interruption for the purpose of extortion, sabotage, terrorism and hacktivism, etc. [8,9]. Further, the large-scale deployment of an AMI means that the attack surface that needs to be protected will increase significantly. This problem may be very dangerous. For example, after a malicious attacker obtains an encryption key, a large number of smart meters may be remotely disconnected, resulting in a major economy loss for the power company. Losses can even have a devastating impact on people’s lives and safety. Therefore, it is very urgent to ensure the confidentiality, availability, integrity and reliability of AMI information.

In the field of AMI security, a large number of documents have been published at home and abroad, which are based on cryptographic technologies, such as encryption [10,11], key distribution and management [12–14], authentication [15,16], privacy protection [17–19], security protocol [20,21], etc. However, in order to improve the security protection of AMI, the intrusion detection system (IDS) must be supplemented. Therefore, the research on smart grid intrusion detection technology is of great practical significance.

Therefore, it is necessary to conduct IDS research on the AMI. Since the AMI is connected to the computer, it is vulnerable to network attacks, but it is different from the traditional network environment. The AMI has some unique characteristics. For example, limited computing and processing resources require fast model training, while the intrusion detection rate is higher.

The contribution of this paper is that it combines the characteristics of the fast learning speed of the ELM and a strong model generalization ability to meet the needs of smart grid intrusion detection. On the other hand, the randomness of the input weights and hidden layer bias in the ELM do not guarantee the optimal performance of the extreme learning machine’s intrusion detection model. The genetic algorithm is an optimization algorithm based on the global search for the optimal solution, through selection, crossover, mutation, etc., to complete the optimization of the model. This paper introduces a genetic algorithm into the ELM and proposes a GA-based ELM smart grid AMI intrusion detection method to make it more suitable for the field of smart grid security.

The paper is organized as below, in sections: Sections 2 and 3 are about some important related works and research foundation. In Sections 4 and 5, the definitions of our proposed algorithm are given, then we discuss the details and advantages of the algorithm. In Section 6, the simulation results show that the proposed algorithm performs better compared with other algorithms, and then, Section 7 concludes the whole paper.

2. Related Works

The smart grid intrusion detection system can be divided into two types by the original data source: intrusion detection based on host information and network information. The former monitors and analyzes the file integrity, application logs and system calls, etc. The other one monitors and analyzes the data packet and flow characteristics in the communication network. However, according to the detection mechanism, there are three detection methods for IDS in the current power system AMI: abnormality-based detection, specification-based detection and signature-based detection [22]. Signature-based IDS uses a blacklist method, while the other two IDS use a whitelist method. Since signature-based intrusion detection
methods cannot detect some new attacks against AMI and require frequent updates, research mainly focuses on anomaly-based and standardized intrusion detection.

Anomaly-based IDS uses statistical measures to identify deviations from predefined normal behavior. In recent years, IDS based on machine learning (ML) and data mining technology has been a research focus in network security. A similar approach was adopted when developing IDS for AMI.

Zhang and others described a distributed IDS for AMI and Supervisory Control and Data Acquisition (SCADA) systems [23]. The system relies on anomaly-based sensors deployed in the HAN (home area network), NAN (neighborhood area network), WAN (wide area network) and SCADA environments. IDS sensors are collected from communication streams. For security-related information, two ML algorithms (support vector machine (SVM) and clonal selection algorithm (CLONALG)/Airs2Parallel) are used to process the data to identify malicious behavior. These algorithms require good training to achieve good performance; however, attack samples in AMI are basically rare. In addition, traditional ML algorithms are difficult to implement in embedded systems. Krishna and others proposed an anomaly detection method that combines the PCA (principal component analysis) feature selection algorithm and the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) clustering algorithm to verify the integrity of smart meter measurements [24]. They used an open smart meter database, which was deployed from actual deployment and obtained to simulate normal electricity consumption behavior. Mohammadi and others proposed an anomaly and signature-based neighborhood network intrusion detection method in AMI [25]. This method considers various attacks against the physical layer, media access control (MAC) layer, transport layer and network layer and considers the specific needs of the NAN. Mustafa et al. [26] first introduced data mining technology to AMI’s IDS and studied various data flow mining algorithms that performed well on the KDDCUP99 dataset, and they further expanded the research work [27], considering more data flow mining algorithms and the feasibility of the analysis of different IDS on three different components of the AMI (smart meter, data concentrator and AMI data transponder). Aiming at the infrastructure of the AMI, Fadwa et al. proposed an intrusion detection system based on real-time distribution (DIDS) [28] using multi-layer implementation and data flow mining technology; using a method called Mini-Batch K-means, the unsupervised online cluster technology monitors the data flow in the advanced measurement system to determine whether there is abnormal traffic.

The specification-based IDS models the ideal behavior of the system through its functions and security strategies. Any sequence of operations performed outside the specification is considered a violation and a suspicious intrusion. Specification-based IDS can be seen as a “stricter” exception-based IDS.

For network intrusion detection, Berthier and others used a specification-based method to monitor the ANSI C12.22 protocol through dedicated sensors deployed in the NAN [29]. The uniqueness of this solution is that it uses a formal method to prove that the specification-based checker provides sufficient coverage related to the AMI security policy. Jokar and others proposed a hierarchical specification-based IDS for the HAN. An intrusion detection system based on the IEEE 802.15.4 standard can detect abnormal behavior at the physical layer and MAC layer [30]. Ali and others proposed a mutation-based IDS that made the attacker’s behavior unpredictable and, at the same time, ensured the certainty of the system [31]. They used the event logs collected from the intelligent collector to conduct AMI behavior, modeling, and then using, the invariant specifications generated by AMI behavior and variable configurations to verify these logs. In addition, they further extended their work [32,33], using a fourth-order Markov chain to model the event log and construct a specification written in linear time logic (LTL), and proposed a method of configuring randomization modules to resist and circumvent imitation attacks. Ruanand and others proposed a hybrid IDS framework that combined techniques based on specifications and signatures to handle known or unknown attacks [34]. The framework can also be deployed on devices with limited resources in the AMI. In [35], Gao and others proposed a scheme named IELM (the incremental extreme learning machine);
in this scheme, the features about security are selected from the network traffic, where a better detection accuracy can be obtained, on the NSL-KDD standard dataset [36].

For host intrusion detection, Tabizi and others proposed a model-based smart meter IDS construction technology [37] that uses LTL to represent the time relationship between monitored entities, which is suitable for smart meters with low processing capacity/limited memory capacity and other limitations. They implemented the IDS on an open-source smart meter platform, SegMeter, and verified its effectiveness. Liu and others used color Petri nets to describe the information flow between the units in the smart meter and proposed a threat model for the smart meter [38]. Aiming at the situation of low processing power and limited memory capacity, an attack against false data injection was proposed, a collaborative intrusion detection mechanism.

In summary, there is a lot of literature on intrusion detection for smart grids, and scholars have conducted many explorations, whether based on host information and network information or based on abnormality-based detection, specification-based detection and signature-based detection.

Under different classification methods, various researches have achieved certain effects, but none of them can completely solve the problems of IDS in smart grids. Some new research methods are constantly emerging, especially in today’s development of artificial intelligence technology; intrusion detection methods combining machine learning and deep learning [39–41] are constantly proposed. However, until today, a mature intrusion detection scheme has not been formed completely, and there is a lack of a benchmark for evaluation.

In addition, how to solve the security problem of AMI in smart grids is also a research focus. Therefore, the research on AMI strategy for smart grid security combined with artificial intelligence method ELM in this paper is of great theoretical and practical significance.

3. Research Foundation

In 2004, Professor Guang-bin Huang proposed an ELM, which is an effective single hidden layer feed-forward neural network with high generalization ability, as little manual intervention as possible and guarantees the premise of a certain learning accuracy; the speed of the algorithm is greatly accelerated [42]. Compared with traditional neural networks, it saves a lot of time and cost. This is because the algorithm does not need to repeatedly optimize the input weights and hidden layers from the input layer to the hidden layer during the training process. There is neuron offset, but the input weight and hidden layer offsets are obtained through random initialization. During training, we only need to determine the number of hidden layer neurons; we can directly get the output weight of the model to complete the model training.

Suppose a single hidden layer feed-forward neural network is provided with $N$ training samples $(x_i, t_i)$, where $x_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^T \in \mathbb{R}^n$ is the sample input vector and $t_i = [t_{i1}, t_{i2}, \ldots, t_{im}]^T \in \mathbb{R}^m$ is the target output vector, and where $n$ represents the number of features of the input sample and $m$ represents that the training sample has a total of $m$ classes. In addition, this single hidden layer neural network has $L$ hidden nodes; then, the network output can be expressed as:

$$\sum_{i=1}^{L} \beta_i g(\omega_i x_j + b_i) = o_j, \quad j = 1, 2, \ldots, N$$

where $g(\bullet)$ is the activation function, $\omega_i = [\omega_{i1}, \omega_{i2}, \ldots, \omega_{in}]^T$ is the input weight vector between the input layer and the hidden layer and $\beta_i = [\beta_{i1}, \beta_{i2}, \ldots, \beta_{im}]^T$ is the output weight vector between the hidden layer and the output layer; $b_i$ is the $i$th offset value of the first node of the hidden layer, where $\omega_i x_j$ represents the weight of the inner product of the value and the training sample value, and $o_j$ represents the actual output of the network model. The single hidden layer feedforward neural network model diagram is shown in Figure 1.
The goal of the single hidden layer feed-forward neural network is to minimize the error value of the output result, which is:

$$\sum_{j=1}^{N} \|o_j - t_j\| = 0 \quad (2)$$

where $t_j$ is the expected output.

According to the Formulas (1) and (2), $\beta_i$, $\omega_i$ and $b_i$ exist to make the following formula true:

$$\sum_{i=1}^{L} \beta_i g(\omega_i \cdot x_j + b_i) = t_j, j = 1, 2, \ldots, N \quad (3)$$

Based on the matrix, the expression could be simplified to:

$$H\beta = T \quad (4)$$

In the formula, $H$ is the output value matrix of the hidden layer, $\beta$ is the output weight matrix from the hidden layer to the output layer and $T$ is the expected output matrix. Further $H$, $\beta$ and $T$ are expressed as follows:

$$H = \begin{bmatrix} g(\omega_1 \cdot x_1 + b_1) & \cdots & g(\omega_L \cdot x_1 + b_L) \\ \vdots & \ddots & \vdots \\ g(\omega_1 \cdot x_N + b_1) & \cdots & g(\omega_L \cdot x_N + b_L) \end{bmatrix}_{N \times L} \quad (5)$$

![Figure 1. Single hidden layer feed-forward network.](image-url)
In most cases, $H\beta = T$ cannot be established. In order to train the model, a set of parameters is found: $\omega_i, \beta_i$ and $b_i$, so that the following equation is established:

$$\|H(\omega_i, b_i)\beta_i - T\| = \min_{\omega_i, \beta_i, b_i} \|H(\omega_i, b_i)\beta_i - T\|$$

(8)

When dealing with such problems, the traditional neural network algorithm will continuously optimize the parameters during the iteration process, resulting in a longer model training time. In the model training of the extreme learning machine, the input weights and the bias of the hidden layer are randomly initialized. Further, the output weight matrix. $H$ is determined, so the model becomes a linear system, $H\beta = T$, which can be obtained by least squares $\beta$. The formula is as follows:

$$\beta = H^+ T$$

(9)

In the formula, $H^+$ is the Moore-Penrose generalized inverse matrix of the output weight matrix $H$.

In summary, the learning process of the extreme learning machine is as follows:

1. Randomly initialize the input weights $\omega_i$ and the offset of the hidden layer $b_i$,
2. Calculate the output weight of the hidden layer $H$ and
3. Calculate the output weight from the hidden layer to the output layer $\beta$.

Genetic algorithm is a global optimization algorithm for finding the optimal solution [43]. It is realized by simulating the Darwinian biological evolution theory and the genetic mechanism of nature. In the genetic algorithm, the search space about the representative problem is mapped to a genetic space, and the possible solutions in the genetic space are encoded to form a vector—that is, the chromosome in biological evolution—and each gene in the chromosome corresponds to each element of the vector through duplication, crossover, mutation and other ways to continuously evolve the “chromosome” represented by the decode, so as to find the optimal solution to the representative problem. The genetic algorithm has the characteristics of high parallelism and self-adaptation, the principle and operation are simple and it has been widely used in many optimization fields. The algorithm steps are as follows:

1. Population initialization

   In the process of initializing the population, including determining the size of the population $M$, the initial population $Y(0)$ in the genetic algorithm is composed of $M$ randomly generated chromosomes, the largest genetic algebra $N$, crossover probability $P_x$ and mutation probability $P_m$ in inheritance.
(2) Individual fitness calculation

Let the \( t \) generation population be \( Y(t) \), and further, set \( f(y) \) as the population, where \( y \in M = \{y_1, \ldots, y_M\} \) and \( y_i = [x_{1i}, \ldots, x_{mi}] \) and \( m \) represent the number of genes in each chromosome; then, the calculation formula of \( f(y) \) is as follows:

\[
f(y) = \begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix} = \begin{bmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{M1} & \cdots & x_{Mm} \end{bmatrix}
\]

(10)

Let \( \text{fit}(y) \) be the fitness function, where \( y \in \{y_1, \ldots, y_M\} \) if it is an optimized neural network algorithm; then, the calculation formula of the fitness function of each chromosome is as follows:

\[
\text{fit}(y_i) = \frac{1}{n} \sum_{j=1}^{n} (O_j - T_j)^2
\]

(11)

In (11), \( O_j \) is the predicted value output of the \( j \) chromosome, and \( T_j \) is the actual value output of the model; \( n \) represents the total number of input data.

(3) Individual evolution

Individual evolution—that is, in the model training process—imitates the biological evolution principle of nature; this process includes the following three steps:

- Selection operation: Selection operation is based on the fitness of individuals in the population and randomly selects \( L \) parent pairs from the \( t \) generation population of \( Y(t) \), where \( L \geq M \).
- Crossover operation: Crossover operation is the key algorithm in GA, which determines the global convergence. Among them, the key criterion is that the offspring chromosome should inherit the excellent characteristics of the parent chromosome and, at the same time, ensure the feasibility of the offspring itself and further, randomly select \( L/2 \) chromosomes of the parents. When the probability of the chromosomes is less than the crossover probability \( P_x \), the crossover is made by randomly formulating one or more points, so as to obtain two progeny chromosomes and, finally, obtain \( L \) intermediate individuals by crossover.
- Mutation operation: \( L \) intermediary individuals are obtained through crossover operations. Then, the mutation operation is performed according to the mutation probability \( P_m \), which is achieved by changing alleles and, finally, determined to form \( L \) candidate individuals.

(4) Child selection

Select \( M \) chromosomes from \( L \) candidate individuals according to fitness and form a new generation of population \( Y(t + 1) \) by the fitness ratio method. If the probability of a certain body being selected is \( P_x \), then the calculation formula of \( P_x \) is:

\[
P_x = \frac{\text{fit}(y_i)}{\sum \text{fit}(y_i)}
\]

(12)

In the formula, \( \text{fit}(y_i) \) is the fitness value of the \( i \) candidate, and \( \sum \text{fit}(y_i) \) is the sum of the fitness values of all candidates in the population, and \( y_i \) represents the \( i \) candidate in the population.

The specific selection steps are as follows:

- Calculate the fitness value of each candidate individually,
- Calculate the sum of the fitness values of all candidates in the population and record the intermediate cumulative value \( S \)-mid,
- randomly generate a number between \( 0 < n < \text{sum} \),
• Select the corresponding first chromosome of $S$-mid to enter a new population,
• Repeat steps (c) and (d) until a population of chromosomes is formed and
• Stop evolution.

When the genetic algorithm iterates to the maximum genetic algebra $N$, the optimization of the genetic algorithm ends. At this time, the individual with the highest fitness value is selected as the global optimal solution for this model optimization, namely:

$$\text{best} = \max \{ \text{fit}(y_i) \}$$  \hspace{1cm} (13)

4. Data Selection and Technical Evaluation Indicators

4.1. Data Selection

In a smart grid, because the advanced measurement system is connected to the internet and is susceptible to network intrusion, the advanced measurement system can be specifically divided into a three-layer network model: The first-level network is composed of smart meters and controllable appliances, which is susceptible to DoS and probing attacks; the second-level network is composed of the network before the smart meter and the concentrator, which is susceptible to user-to-root (U2R) attacks; the third-level network is composed of the network between the concentrator and the data processing center and is susceptible to remote-to-login (R2L) attacks [4]. In response to these four types of network attacks, this paper selects the KDDCUP99 dataset [44] for verification experiments, which can fully represent the IDS situation that AMI may encounter.

KDDCUP99 is composed of labeled training data and unlabeled test data. The training set data consists of five million connection records, and the test set data consists of two million records. Each record has 41 feature attributes and a class identifier. The characteristic attributes include the basic characteristics of the transmission control protocol (TCP) connection (nine types in total), content characteristics of the TCP connection (13 types in total), time-based network traffic statistical characteristics (nine types in total) and host-based network traffic statistical characteristics (10 species in total). Class identifiers include normal and abnormal. Among them, abnormal types can be divided into four attack types, which are further subdivided into 39 attacks. The training set contains 22 attacks, and the test set contains the rest (17 attacks).

The four types of abnormalities include DoS, R2L, U2R and probing:

1. DoS attack: Denial of Service attack. Classification identification: back, and, Neptune, pod, smur and, teardrop.
2. R2L attack: Unauthorized access from a remote host. Classification identification: ftp_write, guess_passwd, imap, multihop, phf, spy, warezclient and warezmaster.
3. U2R attack: Local supergroup users have unauthorized access to remote computers. Classification identification: buffer_overflow, loadmodule, perl and rootkit.
4. Probing attack: Attack by scanning ports. Classification identification: ipsweep, nmap, portsweep and satan.

In addition, KDDCUP99 provides a 10% training set that contains most categories of all training sets. In this paper, the 10% of the set is used as our experimental data. Furthermore, 10% of the training set is still relatively large; there were still 494,021 datasets. It requires further sampling. In this experiment, 10,000 pieces of data were divided as the training set, and 10,000 pieces of data were divided as the test set. In the 10% KDDCUP training set, probing, U2R and R2L have less data. In order to enhance the model’s detection of these three types, this article will use all the records of probing, U2R and R2L and split half of each type of data as training data, and the other half of the data are used as test data. For the two types of data, normal and DoS, refer to [45]; sampling is completed at a ratio of 1:3. For the training data set and test data sets of this paper, see Table 1.
Table 1. Quantity comparison of the experimental data. DoS: Denial of Service.

| Type of Invasion | 10% Data Set | Training Set | Test Set |
|------------------|--------------|--------------|---------|
|                  | Quantity     | Proportion/\%| Quantity | Proportion/\%| Quantity | Proportion/\%|
| Normal           | 97,278       | 19.69        | 5518    | 55.18        | 5518    | 55.18        |
| DoS              | 391,458      | 79.24        | 1840    | 18.4         | 1840    | 18.4         |
| Probing          | 4107         | 0.83         | 2053    | 20.53        | 2053    | 20.53        |
| R2L              | 1126         | 0.23         | 563     | 5.63         | 563     | 5.63         |
| U2R              | 52           | 0.01         | 26      | 0.26         | 26      | 0.26         |
| Total            | 494,021      | 100.00       | 10,000  | 100.00       | 10,000  | 100.00       |

4.2. Technical Evaluation Indicators

Six indicators are mainly used to evaluate the smart grid intrusion detection algorithm—namely, test error, correct rate, false negative rate, detection rate, false alarm rate and detection accuracy [45,46]. In the experiment, true positive (TP) refers to the number of normal types that triggered smart grid attack detection and was correctly identified as a normal type, false positive (FP) refers to triggered smart grid attack detection and the number of false alarms that are mistakenly recognized as the correct type, true negative (TN) refers to the number of abnormal alarms that triggered the smart grid attack detection and were correctly identified as abnormal alarms and false negative (FN) refers to the correct type of numbers that triggered the smart grid attack detection and were mistakenly identified as an abnormal alarm. The test error refers to the difference between the actual output type and the expected output type, $TO_j$ is the $j$th expected output value, $TR_j$ is the $j$th actual output value and the accuracy rate refers to the proportion of all types that are correctly detected to account for all types. The false negative rate refers to the percentage of the abnormal types that are mistakenly recognized as normal types in all abnormal types, the detection rate refers to the proportion of correctly detected types (T) in the total number of types (S), the false alarm rate refers to the proportion of normal types that are mistakenly recognized as abnormal types in all normal types and the detection accuracy refers to the proportion of correctly identified normal types in all those identified as normal types. The definition of the indicator is as follows:

\[
\text{Test error} = \sqrt{\sum_{j=1}^{N} (TO_j - TR_j)^2}
\]  \hspace{1cm} (14)

\[
\text{Correct rate} = \frac{(TP + TN)}{(TP + FP + FN + TN)}
\]  \hspace{1cm} (15)

\[
\text{False negative rate} = \frac{FP}{(FP + TN)}
\]  \hspace{1cm} (16)

\[
\text{Detection rate} = \frac{T}{S}
\]  \hspace{1cm} (17)

\[
\text{False alarm rate} = \frac{FN}{(FN + TP)}
\]  \hspace{1cm} (18)

\[
\text{Detection accuracy} = \frac{TP}{(TP + FP)}
\]  \hspace{1cm} (19)

5. ELM Intrusion Detection Algorithm Based on Optimization

For the traditional ELM, the input weight from the input layer to the hidden layer and the offset of the hidden layer neurons are assigned by random initialization, and the input weight is obtained by the least squares method. During training, only the number of hidden layer neurons needs to be determined. The output weight of the model can be directly obtained, and the training of the
model can be completed, which greatly improves the training speed of the model. Smart grid intrusion detection requires fast training and detection speed. The over-learning machine has a unique advantage, but due to the randomness of input weights and hidden layer bias, cannot solve the problem of gradient descent and other problems, and it cannot guarantee that the performance of the over-learning machine intrusion detection model is optimal. However, the network security of the smart grid is of great importance. It requires training, and the detection is fast, so that it can respond to the security situation of the smart grid in a timely manner. At the same time, it is required to make a correct response to avoid the waste of human and financial resources, so the performance of the model’s intrusion detection is crucial. Genetic algorithm is a classic optimization algorithm based on the natural selection mechanism and genetics principles. It can excellently optimize the model through operations such as selection, crossover and mutation. In order to take advantage of the rapid training of the over-learning machine and, also, to further improve the classification performance of the over-learning machine model when it is faced with smart grid big data, it is fast and has high accuracy. Based on this, we proposed a genetic algorithm-based ELM intrusion detection algorithm, GA-ELM. The genetic algorithm was used to optimize the input weight and hidden layer bias of the extreme learning machine in order to obtain the optimal intrusion detection model of the extreme learning machine.

The steps of the intrusion detection algorithm based on GA for the ELM are as follows:

(1) Population initialization

Let the size of the population be $S$—that is, there are $S$ chromosomes, where each chromosome $x_i$ includes $A\times B$ input weight from the input layer to the hidden layer and $B$ offset of the hidden layer. Further, the initial population is the first-generation population of this genetic algorithm, as follows:

$$X = \begin{bmatrix}
x_1 \\
\vdots \\
x_m \\
\end{bmatrix} = \begin{bmatrix}
a_{11} & \cdots & a_{1A} & b_{11} & \cdots & b_{1B} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
a_{m1} & \cdots & a_{mA} & b_{m1} & \cdots & b_{mB} \\
\end{bmatrix}$$

In the formula, $a_{ij}$ is the input weight from the input layer to the hidden layer, and $b_{ik}$ is the offset of the hidden layer neurons, where $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, A$ and $k = 1, 2, \ldots, B$; the weight and offset of the first-generation population are obtained by random initialization.

(2) Fitness setting and calculation

In the GA-ELM, the genetic algorithm optimizes the input weight of the extreme learning machine $\omega_i$ and the offset of the hidden layer neurons $b_j$. Each chromosome $x_i$ of the population is composed of $\omega_i$ and $b_j$, and the expression of $x_i$ is:

$$x_i = \omega_i + b_j; \omega_i = a_{i1}, \ldots a_{iA}; b_j = b_{j1}, \ldots, b_{jB}$$

According to the theory of the extreme learning machine, the least squares method can be used to obtain the output weight matrix from the hidden layer to the output layer:

$$\beta = H^+ T$$

Setting $O_j$ as the actual output value of the $j$ chromosome and target output value $T_j$, the fitness function of the intrusion detection algorithm of the ELM based on genetic learning is defined as follows:

$$fit = \sqrt{\sum_{j=1}^{n} (O_j - T_j)^2}$$
(3) Parameter optimization

Calculate the fitness of each chromosome in the population, and select, cross and mutate the population according to the fitness to update the new generation of the population. When the number of optimizations reaches the maximum genetic algebra, the GA to the input weights of the ELM and the offset optimization of the hidden layer are ended, and the optimal parameters are selected as the output weights from the input layer of the extreme learning machine to the hidden layer and the offset of the hidden layer neurons.

According to the above steps, the flow chart of the intrusion detection algorithm based on the genetic algorithm for the extreme learning machine is shown in Figure 2.

![Flow chart of the genetic algorithm-extreme learning machine (GA-ELM).](image)

After receiving the data from the smart grid communication network, the genetic algorithm-based extreme learning machine intrusion detection system initializes the extreme learning machine intrusion detection model. Then, the genetic algorithm is used to input weights and a hidden layer of neurons in the ELM. Parameter optimization is performed to obtain the optimal intrusion detection model. Finally, the intrusion is detected by the trained model to form corresponding indicators, determine whether to
trigger the alarm system and report the detection results to the data processing center. This algorithm not only guarantees the requirements of the smart grid for model training speed but, also, further improves the detection accuracy through the genetic algorithm.

6. Experiment

This experiment is tested on the KDDCUP99. In order to verify the effectiveness and superiority of the GA-ELM, it is compared with the intrusion detection algorithm based on the extreme learning machine and the online sequential (OS)-ELM [47] for comparison.

6.1. Intrusion Detection Technology Based on the GA-ELM

Based on the GA-ELM intrusion detection algorithm steps, the first set of the experimental parameters of the GA-ELM algorithm, by referring to experience and testing multiple parameters, we selected a set of relative optimal parameters for setting and set the population size to 20. The maximum genetic algebra is 100, the mutation probability is 0.01 and the crossover probability is 0.7. After a lot of experiments, when the number of the hidden layer of neurons is 480, the performance of the GA-ELM intrusion detection model is better, and the excitation function of the hidden layer of neurons adopts the Sigmoid function. Further, the evolution process of intrusion detection based on the GA-ELM is shown in Figure 3. It can be drawn from the figure that the GA-ELM algorithm performed 100 iterations. At the 0th iteration, it is the original ELM. The test error gradually decreases with the increase of the number of iterations, indicating that the GA-ELM algorithm is effective, and the input weight of the original ELM and the offset of the hidden layer neurons are effectively optimized. The confusion matrix is shown in Figure 4. For the convenience of comparison, the confusion matrix based on the intrusion detection technology of the ELM is given. The number of hidden neurons is also 480, as shown in Figure 5. According to the confusion matrix, we can understand the specific situation of the GA-ELM algorithm for normal, DoS, probing, R2L and U2R detection, from which we can initially see that GA-ELM algorithm-based intrusion detection has normal, DoS, probing, R2L and U2R detection; the detection rate is higher than the intrusion detection based on the ELM. When faced with a small sample intrusion of R2L and U2R, compared with the ELM, the GA-ELM increased the detection rate of R2L by 12.8% and the detection rate of U2R by 23.1%. The effect is remarkable. Furthermore, through the confusion matrix, technical indicators such as the correct rate, false alarm rate, detection rate and detection accuracy can be calculated, which will be further demonstrated in Section 6.

Figure 3. Genetic evolution of the GA-ELM.
Figure 4. Confusion matrix of the GA-ELM. DoS: denial of service.

Figure 5. Confusion matrix of the ELM.
6.2. Intrusion Detection Technology Based on the OS-ELM

According to the principle of the ELM, only the number of hidden-layer neurons needs to be determined during training, but in the training stage, it needs to use all the data for training and testing. While the dataset capacity is large in the field of smart grids, if each training needs to obtain all the original data, it will take a long time and is inefficient. Based on this, the team of Professor Guang-bin Huang proposed an online sequential ELM. The OS-ELM can continuously update the output weight with the arrival of new data, instead of retraining based on the entire dataset, which greatly improved the generalization of the ELM ability. Hence, we applied OS-ELM to the field of smart grid intrusion detection as a comparison experiment between the GA-ELM-based intrusion detection algorithm and ELM-based intrusion detection algorithm. In the experiment, the numbers of hidden-layer neurons of the OS-ELM and GA-ELM are set to 480, keeping the same with the ELM. In addition, the initial training set of the OS-ELM is 5000, updated 1000 data each time, until 10,000 data updates are completed, and the model training is completed. The confusion matrix is shown in Figure 6.

![Confusion Matrix](image)

**Figure 6.** Confusion matrix of the online sequential (OS)-ELM.

It can be seen from the confusion matrix that the detection rate of intrusion detection based on the OS-ELM algorithm for normal, DoS, probing, R2L and U2R is lower than that based on the GA-ELM algorithm. In U2R, compared with the OS-ELM, the GA-ELM algorithm increased the detection rate of R2L by 7.8% and the detection rate of U2R by 11.6%, which has a significant effect. Furthermore, through the confusion matrix, technical indicators such as the correct rate, false alarm rate, detection rate and detection accuracy can be calculated, which will be further demonstrated in Section 6.3.
6.3. Comparison of Detection Indexes

First, the variation of the test errors of the three algorithms GA-ELM, OS-ELM and ELM, with the increase of the number of hidden layer neurons, is investigated. In addition, according to the confusion matrix in the previous two sections, the specific conditions of the classification of the three algorithms GA-ELM, OS-ELM and ELM in smart grid intrusion detection can be obtained intuitively and, based on this, the correct rate of technical indicators such as the false alarm rate, detection rate and detection accuracy. For the above three algorithms, this section will compare their error detection number, detection rate and other detection indicators.

6.3.1. Influence of the Number of Hidden Layer Neurons on the Test Error

While considering the performance of the model, the number of hidden-layer neurons in the three algorithms GA-ELM, OS-ELM and ELM is set to be between 120 and 480 in view of the effect of the number of hidden-layer neurons on the test error. The step length is 40, with a total of 10 experiments; the test results are shown in Figure 7.

![Figure 7. Effect of the number of hidden neurons on the test error.](image.png)

It can be seen from Figure 7 that the test error of the three algorithms gradually decreases with the growth of the number of hidden-layer neurons. Further, the test error of the GA-ELM algorithm is the lowest, indicating that, after the genetic algorithm, the input of the ELM network model, the weights and the offset of the hidden layer neurons are optimized. Under the same test error, the number of hidden-layer neurons of the GA-ELM is much less than that of the OS-ELM and the ELM. It shows that the GA-ELM’s network structure is more compact than the OS-ELM and ELM. In addition, the test errors of the OS-ELM and the ELM are not much different, but the OS-ELM can update the model faster. In the smart grid, it is better than the ELM algorithm—more applicable. In summary, the test error of GA-ELM is lower, the network structure is more compact and it has greater universality for the smart grid.
6.3.2. Evaluation of Other Indicators

According to the confusion matrix, technical indicators such as the correct rate, false alarm rate, detection rate and detection accuracy can be calculated. For the above three algorithms, this section will compare their error detection number, detection rate and other detections. The indicators are shown in Tables 2–4, respectively.

| Identification Type | Error Detection Number/Total Number of Samples |
|---------------------|-----------------------------------------------|
|                     | GA-ELM | OS-ELM | ELM    |
| Normal              | 46/5518| 72/5518| 121/5518|
| DoS                 | 23/1840| 28/1840| 27/1840 |
| Probing             | 59/2053| 71/2053| 80/2053 |
| R2L                 | 31/563 | 75/563 | 103/563 |
| U2R                 | 18/26  | 21/26  | 24/26  |

Table 2. Comparison of the error detection numbers of the three methods. ELM: extreme learning machine, GA-ELM: genetic algorithm ELM and OS-ELM: online sequential ELM.

| Identification Type | Detection Rate/% |
|---------------------|------------------|
|                     | GA-ELM | OS-ELM | ELM    |
| Normal              | 99.17  | 98.70  | 97.81  |
| DoS                 | 98.75  | 98.48  | 98.53  |
| Probing             | 97.13  | 96.54  | 96.10  |
| R2L                 | 94.49  | 86.68  | 81.71  |
| U2R                 | 30.77  | 19.23  | 7.69   |

Table 3. Comparison of the detection rates of the three methods.

| Identification Type | Three Methods |
|---------------------|---------------|
|                     | GA-ELM | OS-ELM | ELM    |
| Correct rate/%      | 98.23  | 97.33  | 96.45  |
| False alarm rate/%  | 0.83   | 1.30   | 2.19   |
| False-negative rate/%| 1.36  | 2.59   | 3.24   |
| Accuracy/%          | 98.90  | 97.91  | 97.38  |

Table 4. Comparison of the detection indexes of the three methods.

It can be seen from Tables 2–4 that the GA-ELM-based intrusion detection algorithm is lower than the OS-ELM-based intrusion detection algorithm and the ELM-based intrusion detection number and false alarm rate. The intrusion detection algorithm is higher than the other two in terms of accuracy. It can be seen that the genetic learning-based extreme learning machine algorithm can effectively optimize the input weight of the ELM and the offset of hidden-layer neurons. By improving the performance of the model, it maintains the training speed of the extreme learning machine while better augmenting the accuracy in a smart grid intrusion detection.

7. Conclusions

In this paper, a thorough analysis of various applications of the extreme learning machine to the field of smart grid AMI intrusion detection was provided, along with a discussion around the impact of over-learning the input weight and randomness of the hidden layer bias. Featuring its prominent optimization ability about classification performance, the genetic algorithm was proposed as a solution to raise model accuracy based on the fast training and detecting speed already secured by the over-learning machine. By mapping the input weight in the hidden layer bias map to chromosome vectors and utilizing the test errors as fitness functions in the genetic algorithm, this GA-based algorithm was able to optimize the parameters based on genetic traits, hence enhancing the performance of
an over-learning model. The best performance could be achieved by setting up input weight and hidden-layer neurons corresponding to the minimized testing error. This paper also presented detailed examinations and comparisons on three different algorithms, including the OS-ELM, ELM and GA-ELM. It was shown that the GA-ELM most effectively augmented the performance of the ELM network, creating a more compact network structure while further improving the accuracy, precision and detection rate of smart grid intrusion detection. This algorithm greatly diminished the false alarm rate and omission alarm rate, which represents a new way of combining algorithm usage in the field of smart grid intrusion detection.

In the follow-up work, when conditions are ripe, we will further collect real smart grid network intrusion data and research smart grid security strategies.

Author Contributions: Z.H. and K.Z. established an experiment platform, provided experimental data after conducting numerous experiments and then composed the first draft of this paper; K.Z. and Z.H. proposed the algorithm in this paper; Y.Z. helped finalize the paper and X.W. and K.G. revised the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Sichuan Science and Technology Program (No. 2019YFG0405) and the Project of Science and Technology on Eletronic Information Control Laboratory and Natural Science Foundation of China (No. U1833122).

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

References
1. FBI-Smart Meter Hacks Likely to Spread. Available online: http://krebsonsecurity.com/2012/04/fbi-smart-meter-hacks-likely-to-spread/ (accessed on 22 May 2020).
2. Knapp, D.E.; Langill, J.T. Industrial Cyber Security History and Trends. In Industrial Network Security, 2nd ed.; Elsevier Inc.: Amsterdam, The Netherlands, 2015; Chapter 3; pp. 41–53.
3. Tang, Y.; Chen, Q.; Li, M.Y. Overview of network attack research in the environment of power information physical fusion system. Power Syst. Autom. 2016, 40, 59–69. (In Chinese with English Abstract)
4. Vega Vega, R.A.; Chamoso-Santos, P.; González Briones, A.; Casteleiro-Roca, J.-L.; Jove, E.; Herrero, A.; Corchado, E.; Calvo-Rolle, J.L.; Matsu, K.; Meizoso-Lopez, M.d.C.; et al. Intrusion Detection with unsupervised techniques for Network Management Protocols over Smart Grids. Appl. Sci. 2020, 10, 2276. [CrossRef]
5. Ipakchi, A.; Albueyeh, F. Grid of the Future. IEEE Power Energy Mag. 2013, 11, 20. [CrossRef]
6. Zhang, J.C.; Chen, Z.Y. The impact of AMI on the future power system. Autom. Electr. Power Syst. 2010, 34, 20–23.
7. Modaghegh, H.; Abasgholi, B.; Gilani, N.; Arian, M.; Soleimani, V. Advanced Metering Infrastructure System Architecture. In Proceedings of the Asia-Pacific Power and Energy Engineering Conference, Wuhan, China, 25–28 March 2011; pp. 1–6.
8. Zhou, L.; Chen, S. A Survey of Research on Smart Grid Security. In Network Computing and Information Security 2012; Springer: Berlin/Heidelberg, Germany, 2012; Volume 345, pp. 395–405.
9. Shein, R. Security Measures for Advanced Metering Infrastructure Components. In Proceedings of the IEEE Power & Energy Engineering Conference, Chengdu, China, 28–31 March 2010; pp. 1–3.
10. Zhang, X.D.; Tong, W.M. Research on Fusion Algorithm of Elliptic Curve Cryptography in Advanced Metering Infrastructure Communication. In Proceedings of the International Symposium on Computational Intelligence and Design, Hangzhou, China, 13–14 December 2014; pp. 395–398.
11. Soykan, E.U.; Ersoz, S.D.; Soykan, G. Identity based signcryption for advanced metering infrastructure. In Proceedings of the IEEE Smart Grid Congress & Fair, Istanbul, Turkey, 29–30 April 2015; pp. 1–5.
12. Kamto, J.; Qian, L.; Fuller, J. Key Distribution and Management for Power Aggregation and Accountability in Advance Metering Infrastructure. In Proceedings of the IEEE Smart Grid Communications, Tainan, China, 5–8 November 2012; pp. 360–365.
13. Wan, Z.; Wang, G.; Yang, Y. SKM: Scalable Key Management for Advanced Metering Infrastructure in Smart Grids. IEEE Trans. Ind. Electron. 2014, 61, 7055–7066. [CrossRef]
14. Woong, G.; Jin, K. Two-Dimensional Key Table-Based Group Key Distribution in Advanced Metering Infrastructure. *J. Appl. Math.* 2014, 2014, 149649.

15. Jeon, J.W.; Lim, S.H.; Yi, O.Y. A Wireless Network Structure and AKA (Authentication and Key Agreement) Protocol of Advanced Metering Infrastructure on the Smart Grid based on Binary CDMA. *J. Korea Inst. Inf. Sec. Cryptol.* 2010, 20, 111–124.

16. Yan, Y.; Qian, Y.; Sharif, H. A Secure and Reliable In-Network Collaborative Communication Scheme for Advanced Metering Infrastructure in Smart Grid. In Proceedings of the IEEE Wireless Communications & Networking Conference, Cancun, Quintana Roo, Mexico, 28–31 March 2011; pp. 909–914.

17. Deng, P.; Yang, L. A Secure and Privacy-Preserving Communication Scheme for Advanced Metering Infrastructure. In Proceedings of the IEEE PES Innovative Smart Grid Technologies Conference, Washington, DC, USA, 24–27 February 2012; pp. 1–5.

18. Chen, S.; Xu, K.; Li, Z. A Privacy-Aware Communication Scheme in Advanced Metering Infrastructure (AMI) Systems. In Proceedings of the IEEE Wireless Communications & Networking Conference, Shanghai, China, 7–10 April 2013; pp. 1860–1863.

19. Saputro, N.; Akkaya, K. On preserving user privacy in Smart Grid advanced metering infrastructure applications. *Secur. Commun. Netw.* 2014, 7, 206–220. [CrossRef]

20. Yan, Y.; Hu, R.Q.; Das, S.K. An efficient security protocol for advanced metering infrastructure in smart grid. *IEEE Netw.* 2013, 27, 64–71. [CrossRef]

21. Yan, Y.; Hu, R.Q.; Das, S.K. A Security Protocol for Advanced Metering Infrastructure in Smart Grid. In Proceedings of the IEEE Global Communications Conference, Austin, TX, USA, 8–12 December 2014; pp. 649–654.

22. Berthier, R.; Sanders, W.H.; Khurana, H. Intrusion Detection for Advanced Metering Infrastructures: Requirements and Architectural Directions. In Proceedings of the First IEEE International Conference on Smart Grid Communications, Gaithersburg, MD, USA, 4–6 October 2010; pp. 350–355.

23. Zhang, Y.; Wang, L.; Sum, W. Distributed IDS in a multi-layer network architecture of smart grids. *IEEE Trans. Smart Grid* 2011, 2, 796–808. [CrossRef]

24. Krishna, V.B.; Weaver, G.A.; Sanders, W.H. PCA-Based Method for Detecting Integrity Attacks on Advanced Metering Infrastructure. In *Quantitative Evaluation of Systems*; Springer International Publishing: Cham, Switzerland, 2015.

25. Mohammadi, B.; Nasim, M.; Jelena, M.; Vojislav, B. A framework for intrusion detection system in advanced metering infrastructure. *Secur. Commun. Netw.* 2014, 7, 205. [CrossRef]

26. Faisal, M.A.; Aung, Z.; Williams, J.R. Securing Advanced Metering Infrastructure Using Intrusion Detection System with Data Stream Mining. In Proceedings of the Pacific Asia conference on Intelligence and Security Informatics, Kuala Lumpur, Malaysia, 29–30 May 2012; pp. 96–111.

27. Faisal, M.A.; Aung, Z.; Williams, J.R. Data-Stream-Based Intrusion Detection System for Advanced Metering Infrastructure in Smart Grid: A Feasibility Study. *Syst. J.* 2015, 9, 31–44. [CrossRef]

28. 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 International Conference on Smart Grid & Clean Energy Technologies, Offenburg, Germany, 20–23 October 2015; pp. 148–153.

29. Berthier, R.; Sanders, W.H. Specification-Based Intrusion Detection for Advanced Metering Infrastructures. In Proceedings of the IEEE Pacific Rim International Symposium on Dependable Computing, Pasadena, CA, USA, 12–14 December 2011; pp. 184–193.

30. Jokar, P.; Nicanfar, H.; Leung, V.C.M. Specification-Based Intrusion Detection for Home Area Networks in Smart Grids. In Proceedings of the IEEE International Conference on Smart Grid Communications, Brussels, Belgium, 17–20 October 2011; pp. 208–213.

31. Ali, M.Q.; Alshaer, E. Probabilistic Model Checking for AMI Intrusion Detection. In Proceedings of the IEEE International Conference on Smart Grid Communications, Vancouver, BC, Canada, 21–24 October 2013; pp. 468–473.

32. Ali, M.Q.; Alshaer, E. Configuration-Based IDS for Advanced Metering Infrastructure. In Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security, Berlin, Germany, 4–8 November 2013; pp. 451–462.

33. Ali, M.Q.; Alshaer, E. Randomization-Based Intrusion Detection System for Advanced Metering Infrastructure. *ACM Trans. Inf. Syst. Sec.* 2015, 18, 7. [CrossRef]
34. Ruan, H.M.; Yeap, G.W.; Lei, C.L. Hybrid intrusion detection framework for advanced metering infrastructure. *Front. Artif. Intell. Appl.* **2015**, *2015*, 894–903.
35. Gao, J.L.; Chai, S.C.; Zhang, B.H.; Xia, Y.Q. Research on Network Intrusion Detection Based on Incremental Extreme Learning Machine and Adaptive Principal Component Analysis. *Energies* **2019**, *12*, 1223. [CrossRef]
36. Hakim, L.; Fatma, R.; Novriandi. Influence Analysis of Feature Selection to Network Intrusion Detection System Performance Using NSL-KDD Dataset. In Proceedings of the International Conference on Computer Science, Information Technology, and Electrical Engineering, Jember, Indonesia, 16–17 October 2019; pp. 217–220.
37. Tabrizi, F.M.; Pattabiraman, K. A Model-Based Intrusion Detection System for Smart Meters. In Proceedings of the IEEE International Symposium on High-assurance Systems Engineering, Miami Beach, FL, USA, 9–11 January 2014; pp. 17–24.
38. Liu, X.; Zhu, P.; Zhang, Y. A Collaborative Intrusion Detection Mechanism Against False Data Injection Attack in Advanced Metering Infrastructure. *IEEE Trans. Smart Grid* **2015**, *6*, 2435–2443. [CrossRef]
39. Dong, Y.; Wang, R.; He, J. Real-Time Network Intrusion Detection System Based on Deep Learning. In Proceedings of the IEEE 10th International Conference on Software Engineering and Service Science, Beijing, China, 18–20 October 2019; pp. 1–4.
40. Yu, S.E.; Cha, J.M.; Lee, T.Y.; Kim, J.; Mun, D.W. Features Recognition from Piping and Instrumentation Diagrams in Image Format Using a Deep Learning Network. *Energies* **2019**, *12*, 4425. [CrossRef]
41. Ishaque, M.; Hudec, L. Feature Extraction using Deep Learning for Intrusion Detection System. In Proceedings of the International Conference on Computer Applications & Information Security, Riyadh, Saudi Arabia, 19–21 March 2019; pp. 1–5.
42. Huang, G.B.; Zhu, Q.Y.; Siew, C.K. Extreme learning machine: Theory and applications. *Neurocomputing* **2006**, *70*, 489–501. [CrossRef]
43. Zhang, K.; Duan, C.; Jia, H.T. Genetic simulated annealing-based coverage-enhancing algorithm for multimedia directional sensor networks. *Int. J. Comm. Syst.* **2015**, *28*, 1598–1609. [CrossRef]
44. Sapre, S.; Ahmadi, P.; Islam, K. A Robust Comparison of the KDDCup99 and NSL-KDD IoT Network Intrusion Detection Datasets Through Various Machine Learning Algorithms. *arXiv* 2019, arXiv:1912.13204.
45. Huang, S.H.; Chen, W.Z.; Li, J. Network intrusion detection technology based on PCA and elm. *J. Jilin Univ.* **2017**, *5*, 576–583, (In Chinese with English Abstract).
46. Zhan, X.; Yuan, H.B.; Wang, X.D. Research on Block Chain Network Intrusion Detection System. In Proceedings of the International Conference on Computer Network, Electronic and Automation, Xi’an, China, 27–29 September 2019; pp. 191–196.
47. Liang, N.Y.; Huang, G.B.; Saratchandra, P. A Fast and Accurate Online Sequential Learning Algorithm for Feedforward Networks. *IEEE Trans. Neural Netw.* **2006**, *17*, 1411–1423. [CrossRef] [PubMed]