Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 9322368, 15 pages
https://doi.org/10.1155/2021/9322368

Research Article

An Improved *K*-Means Clustering Intrusion Detection Algorithm for Wireless Networks Based on Federated Learning

Bin Xie,[1,2] **Xinyu Dong**,[1,2] **and Changguang Wang**[1,2]

[1]*College of Computer and Cyber Security, Hebei Normal University, Shijiazhuang 050024, China*
[2]*Key Lab of Network & Information Security of Hebei Province, Hebei Normal University, Shijiazhuang 050024, China*

Correspondence should be addressed to Changguang Wang; wangcg@hebtu.edu.cn

Received 28 June 2021; Accepted 19 July 2021; Published 10 August 2021

Academic Editor: Ximeng Liu

The existing wireless network intrusion detection algorithms based on supervised learning confront many challenges, such as high false detection rate, difficulty in finding unknown attack behaviors, and high cost in obtaining labeled training data sets. This paper presents an improved *k*-means clustering algorithm for detecting intrusions on wireless networks based on Federated Learning. The proposed algorithm allows multiple participants to train a global model without sharing their private data and can expand the amount of data in the training model and protect the local data of each participant. Furthermore, the cosine distance of multiple perspectives is introduced in the algorithm to measure the similarity between network data objects in the improved *k*-means clustering process, making the clustering results more reasonable and the judgment of network data behavior more accurate. The AWID, an open wireless network attack data set, is selected as the experimental data set. Its dimensionality reduces by the method of principal component analysis (PCA). Experimental results show that the improved *k*-means clustering intrusion detection algorithm based on Federated Learning has better performance in detection rate, false detection rate, and detection of unknown attack types.

1. Introduction

With the rapid development of wireless LAN and mobile communication technologies, the WiFi network has become an indispensable part of people's daily work and life, bringing great convenience. Meanwhile, it is always threatening people's property and information security because of the security threats. Therefore, it is of great significance to study network information security and related technologies. At present, as an important direction of network security research, the network intrusion detection method has attracted the interest of many researchers [1].

Network intrusion detection, as an important dynamic security technology, is the most widely used and effective active network security defence method at present, which makes up for the deficiency of static security technology. Intrusion detection technology is mainly divided into two categories: misuse intrusion detection and anomaly intrusion detection [2]. With the aid of the established database of known intrusion behavior characteristics, misuse intrusion detection technology can use the database to real-time monitor the network data flow in pattern matching and thus determine whether the network behavior and its variant behavior are abnormal. When the data traffic characteristics and features in the database intersect any of the detection rules, it can be concluded that an invasion has occurred. Misuse intrusion detection technology relies on the characteristic library of known intrusion behaviors, which can detect the known intrusion behavior quickly and accurately, thus, determine which type it belongs to. Unfortunately, it cannot detect the network intrusion behavior of an unknown attack type. By establishing the normal behavior characteristic database, the abnormal intrusion detection technology can solve the above problem. When the behavior characteristics of network data do not conform to the rules of the normal behavior characteristic database, the behavior is determined as network intrusion behavior. This technique can detect the intrusion behavior of an unknown attack type, but its false detection rate and missed detection are high [3]. With the increasing diversification and complexity of network intrusion behaviors, the network intrusion detection system based on anomaly detection technology can better adapt to the changeable network environment, which makes it more popular at present.

In the network intrusion detection system with supervised anomaly detection, a large amount of normal behavior data needs to be marked in the practical application process to establish the normal behavior characteristic library. However, it is very difficult and costly to obtain pure and accurate data sets of normal behavior in the real network environment [4]. To solve this problem, the unsupervised anomaly detection method was proposed, which does not rely on labeled data or requires manual or other methods to mark and classify the training data set [5]. Accordingly, the lack of training data set for the detection model is alleviated to a certain extent. However, the major challenges faced by it are still lacking a large number of effective training data sets. How to train the detection models with the network traffic data from different data sources and protect their privacy is an urgent problem to be solved.

Federated Learning [6], first proposed by Bernd, is an effective way to solve the multisource data cotraining model, whose purpose is to carry out collaborative training without sharing private data. The model is not trained intensively with the aggregated multisource data but trained cooperatively with them by only transmitting their relevant encrypted parameters. With the wide application of Federated Learning in intrusion detection systems, researchers have proposed a series of effective detection algorithms based on it. For example, Wang et al. [7] proposed a method of using a DCNN network to extract features under the federated learning mechanism and finally using the Softmax classifier model to carry out intrusion detection. Zhao et al. [8] proposed a network intrusion detection classification model (CNN-FL) that integrates Federated Learning and convolutional neural network (CNN), and it used multisource data to cotrain the same model to improve the classification accuracy of the classifier. Wei et al. [9] proposed a cross-platform malicious user detection method for social networks based on vertical federated learning, which combined multiparty data for modeling and analysis and finally realized more accurate detection of malicious users.

In order to further improve the detection rate of wireless network intrusion detection system, reduce the false detection rate, flexibly discover the attack behavior of unknown attack types, and efficiently reduce the training time of the model, this paper proposes an improved k-means clustering intrusion detection algorithm for wireless network based on Federated Learning. This algorithm no longer takes Euclidean distance as the measurement method between data objects of wireless network but uses cosine distance [10] which is more suitable for high-dimensional network data to describe the similarity between objects and then measures the similarity between any two data objects from multiple perspectives, making the measurement results more reasonable and accurate. At the same time, this algorithm improves k-means clustering by combining three-way decision ideas, realizes the dynamic adjustment of k values by setting the threshold α, and also realizes the delayed decision of uncertain network data by using the q neighborhood of data objects, thus, further improving the detection rate of the intrusion detection system and reducing the false detection rate.

In this paper, we choose the open data set AWID [11] of the wireless network to do the experiments and the method of principal component analysis (PCA) [12] to reduce the dimensionality of experimental data, which significantly decreases the data feature scale and improves the performance of the algorithm. Experimental results show that, compared with the traditional wireless network intrusion detection algorithm, the proposed algorithm in this paper not only realizes the purpose of expanding the amount of training data but also improves the performance of the detection rate, the false detection rate, and the discovery of unknown attack types under the condition of ensuring the data privacy security.

2. Intrusion Detection Model Based on Federated Learning

As an artificial intelligence algorithm, Federated Learning is designed to ensure the security of private data while sharing and using local private data. It solves the problem that multiple computing nodes train the global model together without exchanging the original data. In Federated Learning, the global model is trained among a large number of participants in a distributed means. To avoid the server accessing local data, the participants only train the model locally and update the global model only by passing model parameters to the server. In this paper, it is applied to the wireless network intrusion detection system under the condition of lacking a large amount of effective training data. Instead, it makes full use of the local network data to assist in training the detection model. Accordingly, the overall performance of the detection model can be improved, and the privacy data leakage problem can be avoided in the process of data transmission [13]. The intrusion detection model based on Federated Learning is shown in Figure 1, where it is assumed that there are H participants with the same goal to jointly train the global classifier. In each iteration, the server passes the global model M to each participant, and the participant trains it separately through local wireless network data. After the local training is completed, each participant passes the model parameters back to the server, and the server updates the global parameters by averaging the model parameters of each participant and sends the new parameters to each participant. The core function adopted by the classifier model in this paper is the improved k-means clustering function, and the model parameter passed in the iteration process is the k-means clustering threshold α. The updating process of the global model is shown in

$$M\alpha_{t+1} = \frac{1}{H}\sum_{H=1}^{H} M\alpha_t^H, \tag{1}$$

where $M\alpha_{t+1}$ is the model parameter at the moment of $t+1$

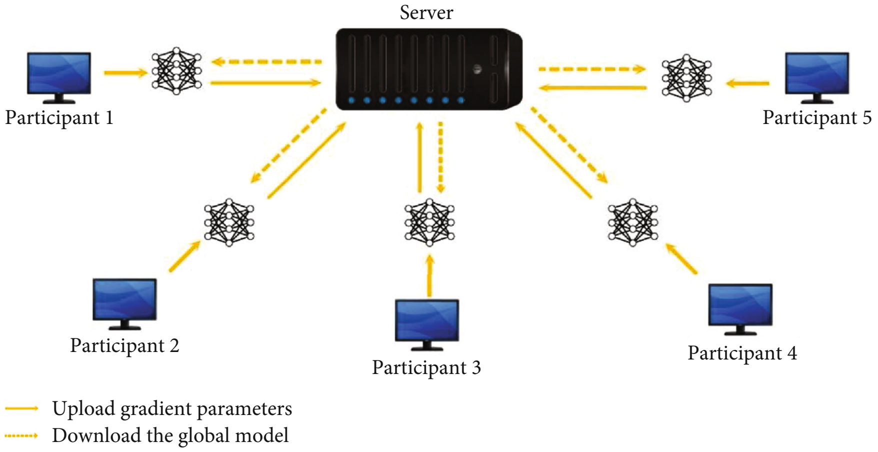


Figure 1: The intrusion detection model based on Federated Learning.

iteration, and $M\alpha_t^H$ is the model parameters uploaded by the H-th participant after t iterations.

3. Improved K-Means Clustering Algorithm

3.1. Traditional K-Means Clustering Algorithm. The traditional k-means clustering algorithm takes the Euclidean distance between data objects as the basis to measure whether the connection between data objects is close and considers that the closer the feature attributes are, the smaller the distance between data objects [14, 15]. It follows two assumptions of clustering: (1) the number of normal data objects in the whole test data set is far greater than that of abnormal data objects; (2) there are obvious differences between normal data objects and abnormal data objects. The traditional k-means clustering algorithm is as follows.

It is a typical two-way clustering method based on the idea of two-way decision, that is, to determine whether a data object belongs to a certain class cluster or not. With the increasing diversity and complexity of wireless network intrusion, the two-way clustering algorithm has some shortcomings when dealing with network data sets. On the one hand, clustering results cannot fully reflect the properties of all data objects themselves. For example, in Figure 2, according to the traditional two-way clustering method, data objects $X1$ and $X2$ are grouped into two clusters $C1$ and $C2$, respectively. There are significant differences between data points $X1$, $X2$, and data objects in the corresponding clusters. Therefore, the adoption of two-way clustering method will inevitably reduce the intrusion detection rate while increase the false detection rate. On the other hand, the traditional k -means clustering algorithm adopts a fixed value of k, the number of categories of all data behavior, which is difficult to predict exactly in advance. Therefore, the fixed k will affect the accuracy of classification and judgment of wireless network data.

3.2. The Improved K-Means Clustering Algorithm by Combining Three-Way Decisions

3.2.1. Idea of the Three-Way Clustering. To solve the problems existing in the application of traditional clustering algorithms in the intrusion detection systems, many scholars have improved the two-way clustering algorithm by introducing the three-way decision idea into the clustering algorithm and then proposed the three-way clustering method. The core idea is to extend the decision items into positive domain decision, negative domain decision, and boundary domain decision [16, 17]. If you have a full grasp of and a comprehensive understanding of things, you can directly make a judgment of acceptance or rejection; otherwise, further investigation is manifested as a delay in decision making [18, 19]. Taking Figure 3 as an example, using the traditional two-way clustering method, data objects $X1$ and $X2$ can only be classified $C1$ and $C2$, respectively. But there are significant differences between data objects $X1$ and $X2$ and those in classes $C1$ and $C2$. The clustering results are shown in Figure 3. The three-way clustering method is used for clustering, and the results are shown in Figure 4. $X1$ and $X2$ are divided into the boundary region of $C1$ and $C2$, respectively, which can be used as uncertain data objects for further processing. Compared with the traditional two-way clustering method, it has obvious advantages in structure and can further cluster and judge outlier data points according to their particularity.

The three-way clustering method is an extension of the traditional two-way clustering method, which is a solution to the reasonable classification of uncertain data objects. If it is difficult to determine the category of the data object immediately, it can assign to the boundary area. The data objects whose category can be determined accurately assign to the core area.

For data set $X = \{x_1, x_2, \cdots, x_n\}$, assume that $C = \{C_1, C_2, \cdots, C_k\}$ is the result of using the two-way clustering method to cluster X. Each category C_i is improved according to the three-way decision idea, and it is represented as $C_i = C_i^P \cup C_i^B$ by two sets C_i^P and C_i^B, where they are

$$C_i^P \neq \phi, i = 1, 2, \cdots, k, \tag{2}$$

```
Input: data sample set $X = \{x_1, x_2, \cdots, x_n\}$ and the number of clustering $k$.
Initialize: Randomly select $k$ data objects to form the first cluster center point set $U = \{u_1, u_2, .., u_k\}$ from the data set $X = \{x_1, x_2, \cdots, x_n\}$.
        Repeat:
        Calculate the distance $d(x_i, u_j)$ between all $x_i$ in $X$ and $k$ cluster center points $u_j$.
        Update the cluster $C_j = \{x_i | d(x_i, u_j) \leq d(x_i, u_l), j \neq l\}$.
        Recalculate the new mean in each cluster $u_j = 1/|C_j| \sum_{x_i \in C_j} x_i$.
        Update the cluster center point set $U = \{u_1, u_2, .., u_k\}$.
        Calculate the objective function $J = \sum_{j=1}^{k} \sum_{x_i \in C_j} d(x_i, u_j)^2$.
Until: $J$ convergence.
Output: $k$ clusters.
```

Algorithm 1:

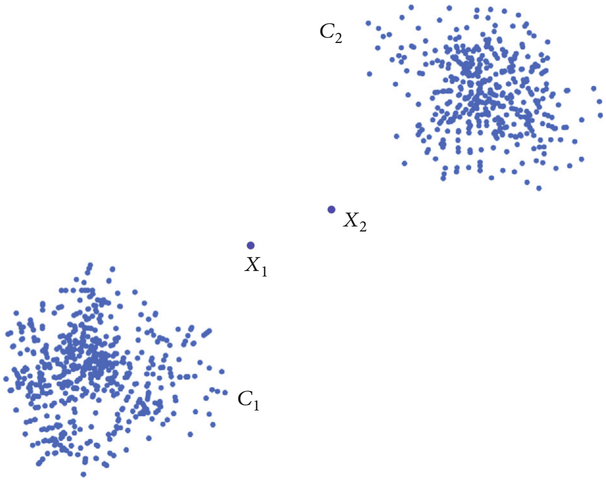


Figure 2: The diagram of data set.

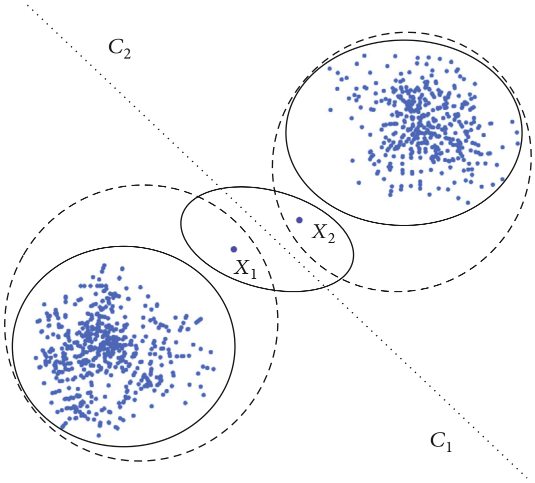


Figure 4: The result of three-way clustering.

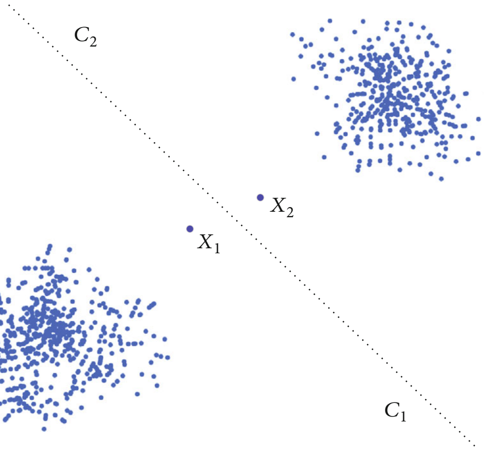


Figure 3: The result of two-way clustering.

$$\overset{i=1}{\overset{k}{U}} \left(C_i^P \cup C_i^B\right) = X, \tag{3}$$

where C_i^P, C_i^B are called the core region and the boundary region of the class, respectively. The data objects in the core region are determined to belong to a class C_i, while the data objects in the boundary region may belong to a class C_i.

In the process of intrusion detection, the core region data are directly classified as intrusion data or normal data, and the boundary region data are deferred to decide to reduce misjudgment.

3.2.2. Three-Way Dynamic Threshold K-Means Clustering Algorithm. In general, the same kind of behavior data in the network intrusion detection system has a high similarity [20], so the vast majority of boundary region C_i^B in the result set based on the three-way clustering algorithm can be basically determined as behavior data that is not of the same kind with the core region C_i^P. On the basis of the traditional three-way clustering algorithm based on q neighborhood, the following improved k-means clustering algorithm of three-way dynamic threshold is proposed to eliminate the influence of human intervening k value on the clustering effect of the k -means algorithm.

In the process of k-means clustering, a distance threshold α is introduced, which is predicted by the hard clustering algorithm and dynamically optimized during the algorithm execution. It adopts distance as the similarity evaluation index, and the introduction of α can effectively cluster those

```
Input: data sample set $X = \{x_1, x_2, \cdots, x_n\}$ and the number of clustering $k$.
Initialize: Randomly select $k$ data objects to form the first cluster center point set $U = \{u_1, u_2, .., u_k\}$ from the data set $X = \{x_1, x_2, \cdots, x_n\}$.
Loop:
    Repeat:
        Calculate the distance $d(x_i, u_j)$ between all remaining $x_i$ in $X$ and $k$ cluster center points $u_j$.
        Update the cluster $C_j = \{x_i | d(x_i, u_j) \leq d(x_i, u_l), j \neq l\}$.
        Calculate $\alpha = 1/k \sum_{j=1}^{k} (1/|C_j| \sum_i d(x_i, u_j))$.
        Traverse all $\{x_i | x_i \notin U\}$ in $X = \{x_1, x_2, \ldots, x_n\}$,
         IF $\exists d(x_i, u_j) < \alpha$, $x_i$ belongs to $C_j{}' = \{x_i | d(x_i, u_j) \leq d(x_i, u_l), j \neq l\}$;
            ELSE.
            Let $u_{k+1} = x_i$, update the center point set to $U = \{u_1, u_2, \cdots, u_k, u_k + 1\}$, where $x_i$ is the stand-alone cluster in the $X$ and the cluster number is updated to $k'$.
        Recalculate the new mean in each cluster $u_j{}' = 1/|C_j{}'| \sum_{x_i \in C_j}{}' x_i$.
         Update the cluster center point set $U = \{u_1, u_2, .., u_k\}$.
         Calculate the objective function $J = \sum_{j=1}^{k} \sum_{x_i \in C_j}{}' d(x_i, u_j{}')^2$.
    Until: $J$ convergence.
Obtain the two-way clustering result: $C' = \{C_1{}', C_2{}', \cdots, C_k{}''\}$.
Calculate $q = 1/10k' \sum_{j=1}^{k} |C_j{}'|$.
Traverse all the class $C_j{}'$ in $C' = \{C_1{}', C_2{}', \cdots, C_k{}''\}$:
        $\forall x_i \notin C_j{}'$, consider the $q$ neighborhood of $x_i$, i.e., $Neig_q(x_i)$, which is the set of q data points closest to $x_i$. IF $Neig_q(x_i) \cap C_j{}' \neq \phi$, we have $x_i \in C_j^{B}{}'$;
        $\forall x_i \in C_j{}'$, IF $Neig_q(x_i) \subseteq C_i{}'$, $x_i \in C_j^{P}{}'$; ELSE $x_i \in C_j^{B}{}'$.
Obtain $C_P = \{C_1^{P}{}', C_2^{P}{}', \cdots, C_{k'}^{P}{}'\}$ and $C_B = C_1^{B}{}' \cup C_2^{B}{}' \cup \cdots \cup C_{k'}^{B}{}'$.
Let $X = C_1^{B}{}' \cup C_2^{B}{}' \cup \cdots \cup C_{k'}^{B}{}'$, then randomly select $k$ data objects to form the first cluster center point set $U = \{u_1, u_2, .., u_k\}$ from $X$.
Do "Loop" on the set $X = C_1^{B}{}' \cup C_2^{B}{}' \cup \cdots \cup C_{k'}^{B}{}'$.
Obtain the two-way clustering result: $C_B' = \{C_1', C_2', \cdots, C_{k'}'\}$.
Output: the final clustering result $C = \{C_B', C_P\}$.
```

Algorithm 2

outlier data objects separately, which can be used as a new clustering center to participate in data training. The dynamic adjustment of the clustering center can eliminate the influence of human intervening k value on the clustering effect of the k-means algorithm to a certain extent.

The k-means clustering algorithm based on three-way dynamic thresholds is as follows.

The final clustering result consists of two data sets C_B and C_P, where C_P contains all the core region data objects through deterministic division and C_B contains the deterministic data objects obtained from all the data in the uncertainty border area by a secondary deterministic division. The accuracy of the resulting clustering set is significantly higher than that of the traditional two-way clustering algorithm.

3.3. Similarity Measure of the Multiple Perspective Cosine Distance.

Euclidean distance is a common measure of the distance between samples used in clustering algorithms. As shown in equations (4) and (5), the traditional k-means clustering method achieves the purpose of clustering by minimizing the sum of the distance between each sample and the center of the class.

$$\text{dist}\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \boldsymbol{x}_i - \boldsymbol{x}_j{}^2, \tag{4}$$

$$\min \sum_{r=1}^{k} \sum_{\boldsymbol{x}_i \in S_r} \boldsymbol{x}_i - C_r{}^2. \tag{5}$$

In the methods of similarity measurement between samples, Euclidean distance focuses on measuring the numerical differences of attribute values between samples, while cosine distance, mainly measuring the differences between dimensions without paying attention to the numerical differences, focuses on the consistency of value directions between dimensions. For wireless network data with higher dimensions, these two traditional measurement methods have their limitations. In this paper, the improved cosine distance measurement method is introduced to the k-means clustering algorithm of wireless network data, and the similarity between data objects in a wireless network is measured from multiple perspectives to obtain a more reasonable and real similarity between two data objects, thus, making the clustering results more ideal. The distance based on cosine can be expressed as:

$$\operatorname{dist}\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = 1 - \cos\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = 1 - \frac{\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j}{\boldsymbol{x}_i \times \boldsymbol{x}_j}, \tag{6}$$

where $\cos\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$ is the cosine value of the angle between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, measuring the similarity between the data objects [18]. According to equation (6), the cosine distance can be regarded as the angle between two objects observed from the perspective of the origin. Therefore, the cosine distance can also be expressed as:

$$\operatorname{dist}\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = 1 - \cos\left(\boldsymbol{x}_i - \boldsymbol{0}, \boldsymbol{x}_j - \boldsymbol{0}\right) = 1 - \frac{\left(\boldsymbol{x}_i - \boldsymbol{0}\right)^T \left(\boldsymbol{x}_j - \boldsymbol{0}\right)}{\boldsymbol{x}_i - \boldsymbol{0} \times \boldsymbol{x}_j - \boldsymbol{0}}. \tag{7}$$

Equation (7) only takes 0 as the reference point, and the angle between two objects is only the angle from the origin, as shown in Figure 5(a). However, if two data objects are approximately in line with the origin, the cosine distance measurement with the origin as the only reference point will lose its effect, as shown in Figure 5(b). Therefore, cosine distance measurement from multiple perspectives will be effective in solving this problem.

Introduce a third nonorigin point $\boldsymbol{d}_h$ as the reference point, and the distance between the data $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ can be expressed as:

$$\begin{aligned}\operatorname{dist}\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) &= 1 - \cos\left(\boldsymbol{x}_i - \boldsymbol{d}_h, \boldsymbol{x}_j - \boldsymbol{d}_h\right) \\ &= 1 - \frac{\left(\boldsymbol{x}_i - \boldsymbol{d}_h\right)^T \left(\boldsymbol{x}_j - \boldsymbol{d}_h\right)}{\boldsymbol{x}_i - \boldsymbol{d}_h \times \boldsymbol{x}_j - \boldsymbol{d}_h}.\end{aligned} \tag{8}$$

When measuring the similarity between two data objects, we can observe the angle between two data objects from each point in the reference point set S_h, that is, the angle between vectors $\boldsymbol{x}_i - \boldsymbol{d}_h$ and $\boldsymbol{x}_j - \boldsymbol{d}_h$. The distance between data $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ can be expressed by the mean of the cosine distance observed from multiple reference points:

$$\begin{aligned}\operatorname{dist}\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) &= 1 - \frac{1}{|S_h|} \sum_{\boldsymbol{d}_h \in s_h} \cos\left(\boldsymbol{x}_i - \boldsymbol{d}_h, \boldsymbol{x}_j - \boldsymbol{d}_h\right) \\ &= 1 - \frac{1}{|S_h|} \sum_{\boldsymbol{d}_h \in S_h} \frac{\left(\boldsymbol{x}_i - \boldsymbol{d}_h\right)^T \left(\boldsymbol{x}_j - \boldsymbol{d}_h\right)}{\boldsymbol{x}_i - \boldsymbol{d}_h \times \boldsymbol{x}_j - \boldsymbol{d}_h},\end{aligned} \tag{9}$$

where $|S_h|$ is the base of the set S_h.

The main idea of selecting reference points for multiple perspectives is as follows.

Assume that A is the point on the outer hypersphere of the unit hypercube in the n-dimensional space, and O is the center of the sphere. When point A is selected on the unit hypersphere according to the equal angular step of the spherical coordinates, in the Cartesian coordinate system $OX_1X_2 \cdots X_n$, the Cartesian coordinates $(X_1, X_2, \cdots, X_n)$ of point A can be calculated as follows:

$$\begin{aligned}
X_1 &= \frac{\sqrt{n}}{2} \cos \frac{\pi k_1}{N} (k_1 = 1, 2, \cdots, N-1), \\
X_2 &= \frac{\sqrt{n}}{2} \sin \frac{\pi k_1}{N} \cos \frac{\pi k_2}{N} (k_1, k_2 = 1, 2, \cdots, N-1), \\
&\cdots\cdots \\
X_{n-2} &= \frac{\sqrt{n}}{2} \sin \frac{\pi k_1}{N} \sin \frac{\pi k_2}{\mathrm{N}} \cdots \cos \frac{\pi k_{n-2}}{N} \\
&\quad \cdot (k_1, \cdots, k_{n-2} = 1, 2, \cdots, N-1), \\
X_{n-1} &= \frac{\sqrt{n}}{2} \sin \frac{\pi k_1}{N} \sin \frac{\pi k_2}{\mathrm{N}} \cdots \sin \frac{\pi k_{n-2}}{N} \cos \frac{2\pi k_{n-1}}{N} \\
&\quad \cdot (k_1, \cdots, k_{n-2} = 1, 2, \cdots, N-1, k_{n-1} = 0, 1, 2, \cdots, N-1), \\
X_n &= \frac{\sqrt{n}}{2} \sin \frac{\pi k_1}{N} \sin \frac{\pi k_2}{N} \cdots \sin \frac{\pi k_{n-2}}{N} \sin \frac{2\pi k_{n-1}}{N} \\
&\quad \cdot (k_1, \cdots, k_{n-2} = 1, 2, \cdots, N-1, k_{n-1} = 0, 1, 2, \cdots, N-1).
\end{aligned} \tag{10}$$

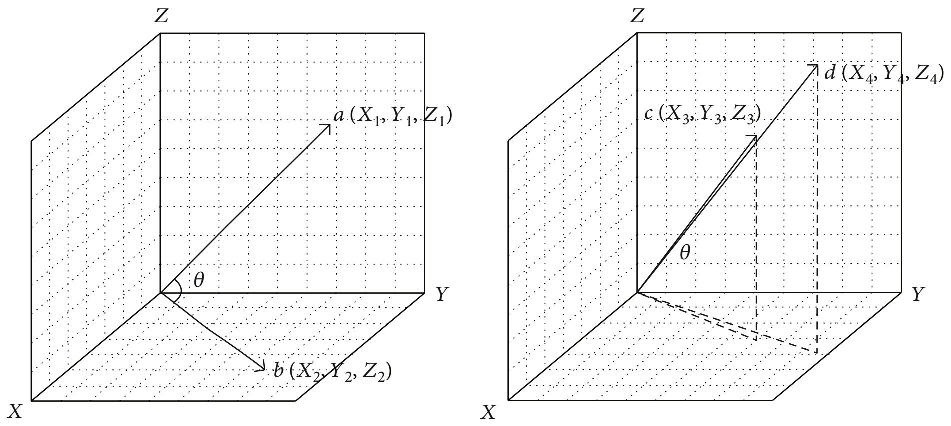
In particular, suppose that A is any a point on the unit sphere in three-dimensional space, α_1 is the angle between $\overrightarrow{OA}$ and the x-axis, and α_2 is the angle between the projection of $\overrightarrow{OA}$ on the plane YOZ and the y-axis, as shown in Figure 6(a). When choosing point A on the unit hypersphere by equal angular step, let $\alpha 1 = \pi k_1 / N (k_1 = 0, 1, 2, \cdots, N-1)$, where $\lambda_1 = \pi / N$ is the radian step length of α_1 whose value varies with k_1. $X = |\overrightarrow{OA}| \cos \alpha 1$ is the coordinate of $\overrightarrow{OA}$ on the x-axis, and $|\overrightarrow{OA}| \sin \alpha 1$ is the projection of $\overrightarrow{OA}$ on the plane YOZ. Let $\alpha 2 = 2\pi k_1 / N (k_2 = 0, 1, 2, \cdots, N-1)$, and we can obtain the coordinates of the vector $\overrightarrow{OA}$ on the y-axis and z-axis:

$$Y = |\overrightarrow{OA}| \sin \alpha 1 \cos \alpha 2 = \frac{\sqrt{n}}{2} \sin \alpha 1 \cos \alpha 2 = \frac{\sqrt{3}}{2} \sin \alpha 1 \cos \alpha 2, \tag{11}$$

$$Z = |\overrightarrow{OA}| \sin \alpha 1 \sin \alpha 2 = \frac{\sqrt{n}}{2} \sin \alpha 1 \sin \alpha 2 = \frac{\sqrt{3}}{2} \sin \alpha 1 \sin \alpha 2, \tag{12}$$

where n is the spatial dimension, and the value of n in three-dimensional space is 3. Thus, the coordinates of point A in the spatial Cartesian coordinate system are (X, Y, Z). For example, in three-dimensional space, when $N = 3$ is selected, the datum point coordinates obtained by the multiple perspective method are shown in Table 1 and Figure 6(b).

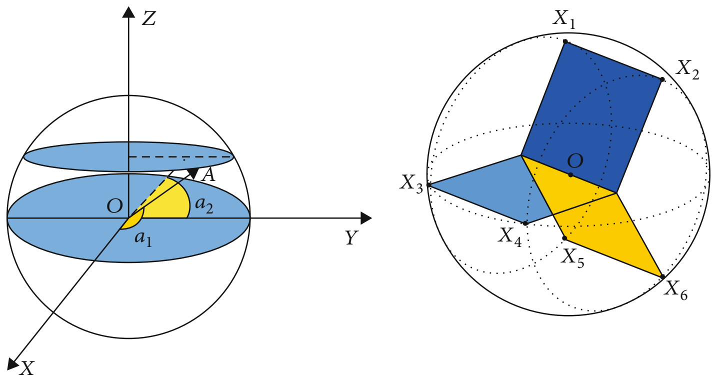
The datum point set Sh contains data objects from all angles, so the cosine distance can more reasonably measure the similarity between two high-dimensional data objects in the case of multiple perspectives. In this paper, the multiple perspective cosine distance is used as the distance measurement method of the improved k-means clustering and applied to the wireless network intrusion



(a)

(b)

Figure 5: Measures the distance between data objects from the origin.



(a)

(b)

Figure 6: Three-dimensional space datum point set.

Table 1: Coordinates of 6 datum points.

X-axis	Y-axis	Z-axis
0.43301	0.750	0
0.43301	-0.375	0.649510
0.43301	-0.375	-0.649510
-0.43301	0.750	0
-0.43301	-0.375	0.649510
-0.43301	-0.375	-0.649510

detection algorithm, and more accurate detection results are obtained. Compared with the traditional Euclidean distance, this algorithm adopts the cosine distance to calculate the distance between high-dimensional data objects, which ensures a higher detection rate and a lower false detection rate. Unfortunately, its time complexity increases significantly, thus, decreasing its detection efficiency. Therefore, the PCA method is used to reduce the dimensionality of the data set in the wireless network, reducing the impact of the time complexity on the detection efficiency of intrusion detection.

4. Improved K-Means Clustering Intrusion Detection Algorithm for the Wireless Networks Based on Federated Learning

In the improved k-means clustering intrusion detection algorithm for the wireless network based on Federated Learning, each participant with the local training data set needs not to share its private data set and preprocesses its local data by itself. All participants use the processed data set to train the classifier model and timely transmit the related parameters. Each participant trains the classifier model by downloading the latest global model parameters in the next iteration. The classifier has a good detection effect on all data sets by cyclic iteration until the overall model reaches the optimal, as shown in Figure 7.

During the actual training, participants and the server exchange relevant parameters at the proper time. In this paper, we assume that each participant is an independent and equal individual, and its training data size is equal or the difference is small. Therefore, the server carries out an arithmetic average operation on the model parameters uploaded by each participant. The algorithm is as follows:

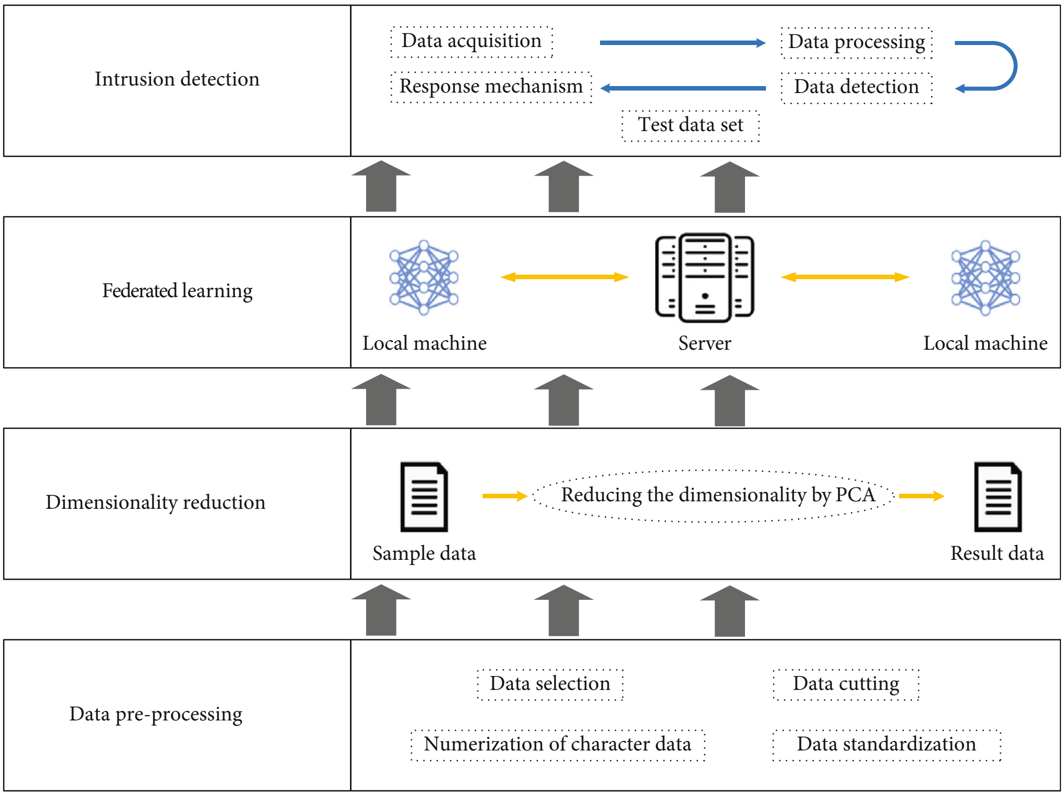


Figure 7: Framework of the improved k-means clustering intrusion detection algorithm based on Federated Learning.

```
1. Input: Data sample set $X = \{x_1, x_2, \cdots, x_n\}$, the step length of multiple perspective $N$, the initial cluster number $k$, the weight vector $l = (l_1, l_2, \cdots, l_p)$, and the largest number of iterations $T$.
2. Initialize: Reduce the dimensionality of all the data objects in the local data set $X = \{x_1, x_2, \cdots, x_n\}$ by use of $l = (l_1, l_2, \cdots, l_p)$.
3. Repeat:
3.      Each participant calculate $U = \{U_1, U_2, \cdots, U_n\}$ according to k-means algorithm.
4.      Each participant passes the $k$-means clustering threshold $\alpha_t$ to the server.
5.      The server sends the new threshold $\alpha_{t+1} = 1/H \sum_{H=1}^{H} a_t^H$ to each participant.
6.      $t = t + 1$
7. Until: $t = T$.
8. Output: Clustering result sets $C$.
```

Algorithm 3:

5. Experiments and Result Analysis

The experimental equipment in this paper is 11 laptop computers with Windows 10 operating system, Intel i5 CPU, and 8 G memory, where one acts as the server. The experimental data set is the wireless network data set AWID. The development environment is Python 3.7. The comparative tests are as follows.

(1) For detection rate and false detection rate, the proposed algorithm is compared with the intrusion detection algorithms based on traditional KNN classification and the density clustering DBSCAN

(2) For detecting unknown attack types, the proposed algorithm is compared with the intrusion detection algorithms based on traditional KNN classification and the density clustering DBSCAN

Table 2: Data distribution.

Data type	Training data set	Testing data set
Normal	1633190	530785
Flooding	48484	8097
Impersonation	48522	20079
Injection	65379	16682
Total	1795575	575643

5.1. Experimental Data Set. The AWID data set is derived from Kolias, which is the network attack data set collected

```
1.Input: The wireless network Data $\boldsymbol{x} = (x_1, x_2,\cdots,x_p)^{\mathrm{T}}$and principal component cumulative variance contribution threshold $R$.
2. Initialize: Construct the initial forward transformation weight vector $\boldsymbol{l} = (l_1, l_2,\cdots,l_p)^{\mathrm{T}}$, the initial backward transformation weight vector $\boldsymbol{l}' = (l_1', l_2', \cdots, l_p')^{\mathrm{T}}$, and the backward transformation data set $\boldsymbol{x}' = (x_1', x_2', \cdots, x_p')^{\mathrm{T}}$.
3. i =1.
4. Repeat
        IF $\mathrm{Var}(y_i) = l_i^{\mathrm{T}} \sum l_i$ reaches its maximum, get $y_{i.}$
        i = i + 1
5.Until i = J
6. Determine $m$, the number of selected principal components, according to $R$ and then obtain $\boldsymbol{y} = (y_1, y_2,\cdots,y_m)^{\mathrm{T}}$.
7. IF $D = 1/n\sum_{i=1}^{p}(x_i - x_i')^2$, the mean square error, reaches its minimum, get the best $\boldsymbol{l} = (l_1, l_2,\cdots,l_p)^{\mathrm{T}}$ and $\boldsymbol{l}' = (l_1', l_2', \cdots, l_p')^{\mathrm{T}}$.
8. Obtain the final $\boldsymbol{y} = (y_1, y_2,\cdots,y_m)^{\mathrm{T}}$ according to the best $\boldsymbol{l} = (l_1, l_2,\cdots,l_p)^{\mathrm{T}}$.
9. Output: the final principal component data set $\boldsymbol{y} = (y_1, y_2,\cdots,y_m)^{\mathrm{T}}$.
```

Algorithm 4:

Table 3: Dimension reduction data results of PCA.

Component	Initial eigenvalue			Extracted square sum of load			Rotated square sum of load		
	Total	Variance %	Acc. %	Total	Variance %	Acc. %	Total	Variance %	Acc. %
1	11.144	14.472	14.472	11.144	14.472	14.472	11.063	14.367	14.367
2	9.271	12.040	26.513	9.271	12.040	26.513	6.425	8.345	22.712
3	7.302	9.483	35.996	7.302	9.483	35.996	6.186	8.033	30.745
4	6.640	8.624	44.620	6.640	8.624	44.620	5.937	7.710	38.455
5	5.745	7.461	52.081	5.745	7.461	52.081	5.741	7.456	45.910
6	3.703	4.809	56.890	3.703	4.809	56.890	4.858	6.309	52.219
7	2.594	3.368	60.259	2.594	3.368	60.259	3.768	4.893	57.113
8	2.468	3.206	63.464	2.468	3.206	63.464	2.532	3.288	60.401
9	2.147	2.788	66.252	2.147	2.788	66.252	2.440	3.169	63.570
10	2.001	2.599	68.851	2.001	2.599	68.851	2.205	2.863	66.433
11	1.623	2.108	70.959	1.623	2.108	70.959	2.105	2.734	69.167
12	1.408	1.829	72.788	1.408	1.829	72.788	2.003	2.601	71.769
13	1.219	1.583	74.370	1.219	1.583	74.370	1.750	2.272	74.041
14	1.199	1.557	75.927	1.199	1.557	75.927	1.442	1.873	75.914
15	1.024	1.330	77.258	1.024	1.330	77.258	1.031	1.339	77.253
16	1.000	1.299	78.556	1.000	1.299	78.556	1.003	1.303	78.556

under the real WiFi network environment with the largest and most comprehensive data volume [21, 22]. According to the attack type level, the dataset is divided into two data subsets: the CLS dataset with four large attack types and the ATK dataset with 16 seed attack types. The 16 seed attack types of the latter are included in the four major attack types of the former. For example, the attack types of Caffe-Latte, Hirte, Honeypot, and EvilTwin in the ATK data set belong to the camouflage attack types in the CLS data set. At the same time, the AWID data set includes two versions: complete data set and condensed data set. This paper uses the condensed version of the CLS dataset. The distribution of data types in the dataset is shown in Table 2, and a normal data record in the dataset is

(0,0,0,1393661303,0.055325,0.055325,0.081227,159,159,-0,0,0,0,26,1,1,1,1,0,1,0,0,0,0,0,0,1,0,0,1,0,0,0,0,0x00000000,0,0,-0,2101680214,0,0,0,0,1,0,0,0,1,2437,0,1,0,1,0,0,0,0,0,0,0,0,0,-32,1,0,0x08,0,0,8,0x00,

0,0,0,0,0,0,ff:ff:ff:ff:ff:ff,ff:ff:ff:ff:ff:ff,00 : 13 : 33 : 87 : 62:6d,00 : 13 : 33 : 87 : 62:6d,00 : 13 : 33 : 87 : 62:6d,0,3684,0,0,0,0,0,0,1,1, 0,0x0000,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0x00000044a18671bc,100,0,0,0,0,0,0,0,0,0,0,1,OTE29224e,6,0,1,0,0x00,0,1,2,2,1,2,0,0,0x0000, 0x0000,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0).

The process of data set preprocessing includes data completion, data rationalization, numerization of character data, data normalization, and reduction of the data attribute dimensionality.

5.1.1. Data Clipping. In the AWID data set, some attributes of a few network data are missing. To ensure the effectiveness of its results, delete those attributes with the missing rate of 80% or more and fill the remaining attribute bits with 0 s.

5.1.2. Data Selection. The number of normal behavior records in the AWID is far greater than the number of attack behavior records. Such is the case in the real network environment. Therefore, we select the 10 : 1 normal behavior record and aggressive behavior record as the training data set to construct the classifier. To fully verify the detection performance of the proposed algorithm on the data behaviors of different attack types, we also select 10 : 1 normal behavior records and attack behavior records to build a test data set. Besides, the training data set and test data set allocated to each participant contain a different number of attack behaviors. To the greatest extent, it restored the situation that some attack behavior data in the local data set of some users in the real wireless network environment is less or does not exist, but it can also have the ability to detect such attack behavior through the training of the Federated Learning detection model.

5.1.3. Numerization of Character Data. The hexadecimal attribute value in the AWID is converted to the decimal attribute value, the MAC address attribute value is converted to the number of its occurrence in the whole data set, and the data attribute value in the form of characters is numerically processed by the one-hot encoding [23, 24] method. The character attribute variable processed by the encoding method can retain the influence degree of the original attribute on the clustering result more reasonably.

5.1.4. Dimensionality Reduction of Data Attributes. The wireless network data in the AWID has 154 attribute values. In this paper, we delete all the attributes with the same values in the data set before the experiment and use the PCA to extract the attributes with a greater contribution rate to reduce the dimensionality of the wireless network data and then reduce the time complexity of the detection algorithm.

5.1.5. Data Standardization. Different attributes in the data set have different ranges. To reduce the impact of such difference on the detection model, we can use z-score standardization [25, 26] on the data to make it a normal distribution. Using distance to measure similarity and the PCA to reduce dimensions, the z-score normalization is better than the Min-max normalization in classification and clustering algorithms.

$$\boldsymbol{y}_i = \frac{\boldsymbol{x}_i - \mu}{\sigma}, \tag{13}$$

where $\boldsymbol{y}_i$ represents the normalized data of $\boldsymbol{x}_i$, $\boldsymbol{x}_i$ represents the i-th eigenvalue, μ represents the data mean value of the feature, and σ represents the standard data deviation of the feature.

5.2. PCA Method for Dimensionality Reduction of Wireless Network Data. In wireless network data, each piece of network data often involves dozens or even hundreds of attribute variables. Too many attribute variables will not only increase the time complexity of the detection algorithm but also bring difficulties to the reasonable analysis of detection results. Although each attribute variable of the network data provides a certain amount of information, its importance and contribution vary. Moreover, in most cases, there is a certain correlation between various attribute variables of network data, which makes the information provided by these attribute variables overlap to a certain extent and affects the accuracy of detection results. Therefore, we adopt the PCA method to deal with these attribute variables and replace the original attribute variables with a small number of variables, so as to achieve dimensionality reduction of wireless network data [27]. The dimensionality reduction process is as follows.

In wireless network data set AWID (154 attributes) [28], the 77-dimensional attributes that have an influence on clustering results were extracted and dimensional reduction was conducted by the PCA method. The principal component variance contribution rate and cumulative variance contribution rate obtained were shown in Table 3. When the PCA is used to reduce the dimensionality of wireless network data sets, the appropriate number of principal components can be selected by adjusting the threshold value of the contribution

Table 4: Structure of test data sets in experiments of the ACC and FAR.

Name of data set	Normal data/piece	Attack data/piece	Attack behavior/type
$H1$	1000	100	3
$H2$	2000	200	5
$H3$	3000	300	6
$H4$	4000	400	8
$H5$	5000	500	10
$H6$	6000	600	11
$H7$	7000	700	13
$H8$	8000	800	14
$H9$	9000	900	15
$H10$	10000	1000	16

Table 5: Structure of test data sets in experiments of detecting unknown attack types.

Name of data set	Normal data/piece	Attack data/piece	Attack behavior/type	Unknown attack behavior/type
$D1$	1000	100	2	1
$D2$	2000	200	3	2
$D3$	3000	300	3	3
$D4$	4000	400	4	4
$D5$	5000	500	5	5
$D6$	6000	600	5	6
$D7$	7000	700	6	7
$D8$	8000	800	6	8
$D9$	9000	900	6	9
$D10$	10000	1000	6	10

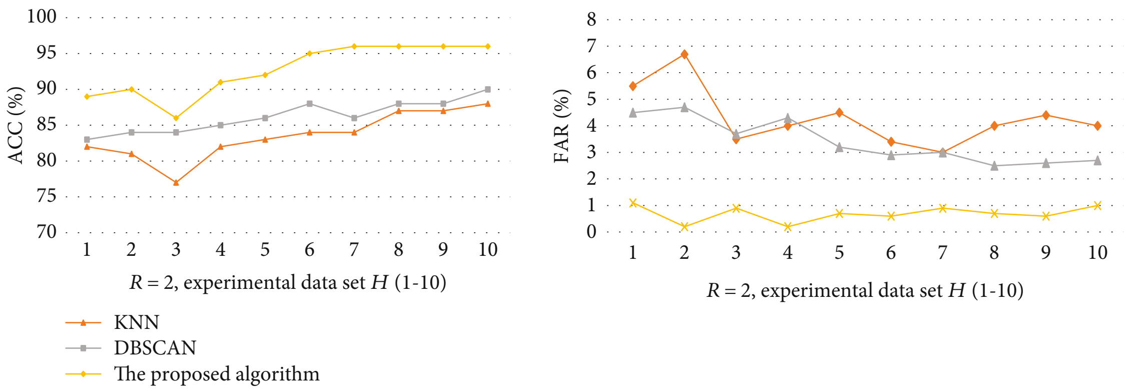


Figure 8: Comparison of ACC and FAR in experiment 1, $R = 2$.

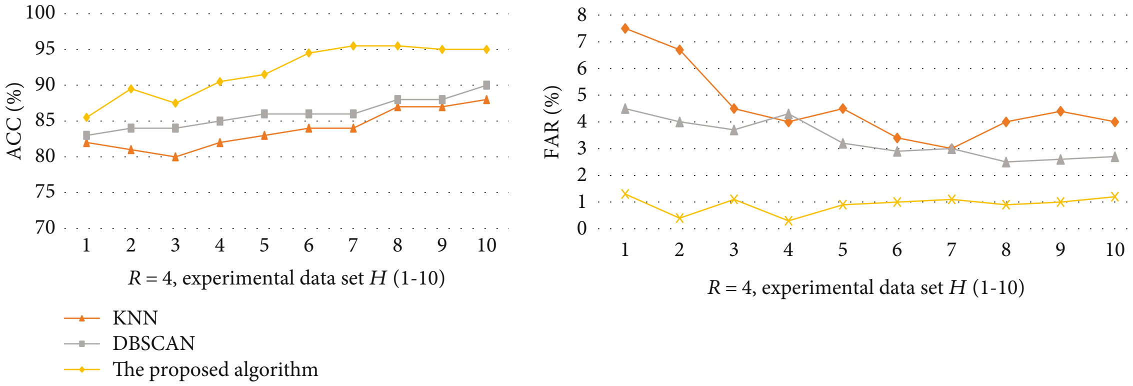


Figure 9: Comparison of ACC and FAR in experiment 1, $R = 4$.

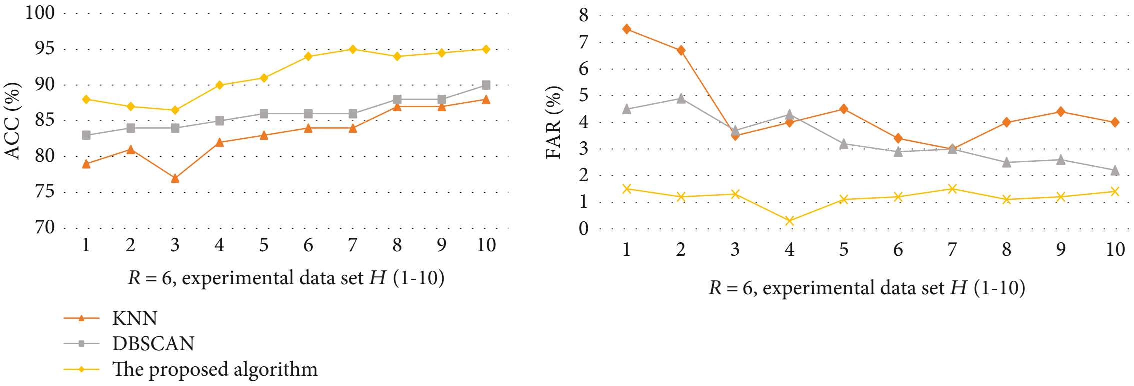


Figure 10: Comparison of ACC and FAR in experiment 1, $R = 6$.

rate of cumulative variance of principal components. The choice of principal component quantity directly affects the characterization ability of original network data. Choosing a small number of principal components to replace the original data may result in poor clustering results and greatly reduce the detection performance of intrusion detection algorithm. Selecting a large number of principal components to replace the original data cannot achieve the purpose of dimensionality reduction. Therefore, how to choose the appropriate number of principal components to replace the original network data needs to be decided according to the specific algorithm and algorithm function, so as to achieve the goal of data dimensionality reduction to the maximum extent on the basis of guaranteeing the high performance of the algorithm. After many experiments, this paper selects the first 16 attributes after dimensionality reduction to carry out intrusion detection experiments and obtains the most ideal detection results. When additional attributes were added for the experiment, the time complexity gradually increased, but the intrusion detection results did not change significantly, so the first 16 attributes were selected in this paper.

5.3. Analysis of Experimental Results. In this paper, detection rate ACC and false detection rate FAR are used as

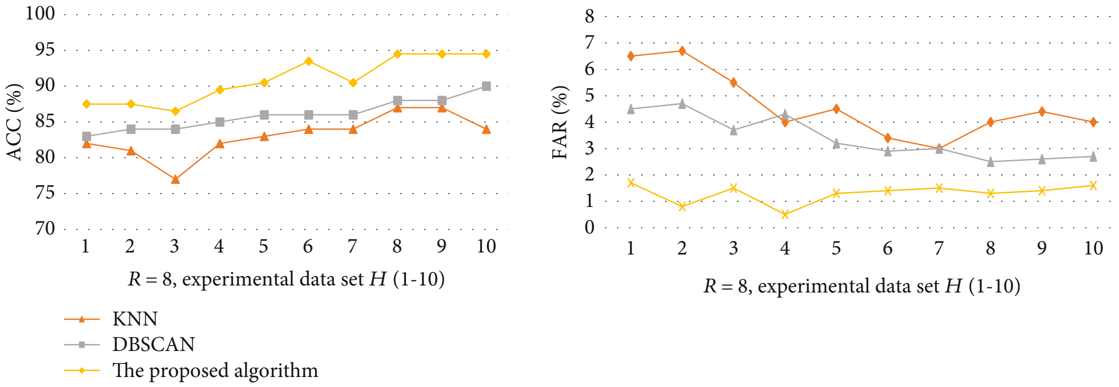


Figure 11: Comparison of ACC and FAR in experiment 1, $R = 8$.

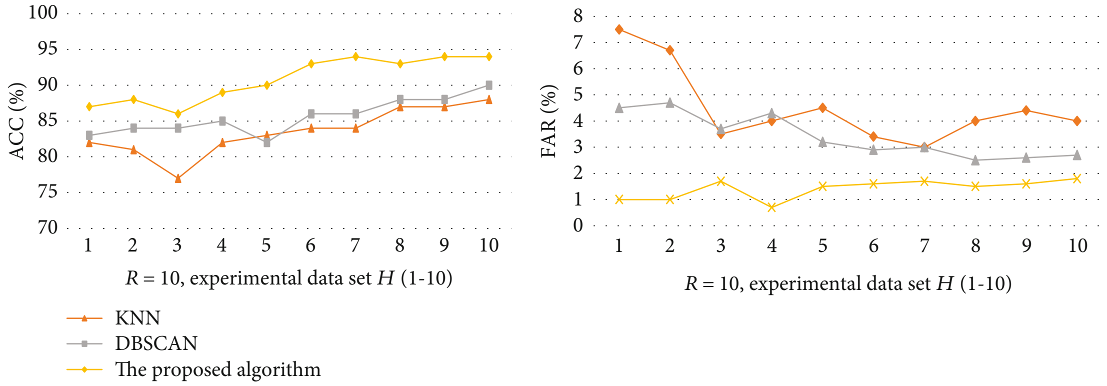


Figure 12: Comparison of ACC and FAR in experiment 1, $R = 10$.

performance evaluation indexes of wireless network intrusion detection algorithm [29]. The details are as follows:

(1) Detection rate ACC is the ratio between the network data of the correctly judged category and the sum of network data. The higher the detection rate, the better the performance of the intrusion detection algorithm

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}. \tag{14}$$

(2) False detection rate FAR is the ratio between the amount of normal behavior data wrongly judged as aggressive behavior and the sum of normal behavior data. In intrusion detection algorithms, the lower the false detection rate, the better the detection performance of the algorithm

$$\mathrm{FAR} = \frac{\mathrm{FP}}{\mathrm{TN} + \mathrm{FP}}, \tag{15}$$

where TN (true negative) represents the number of network data behaviors that correctly identified as normal; TP (true positive) refers to the amount of network data that correctly identifies the network attack behavior as the attack type; FN (false negative) represents the amount of data that misidentifies the network attack behavior as normal; FP (false positive) refers to the amount of network data that wrongly identifies normal data behavior as some attack behavior.

To avoid the contingency of experimental results caused by intrusion detection algorithm testing on a single experimental data set, experimental data sets $H1 - H10$ and $D1 - D10$ of different sizes with different attack behavior classes are randomly selected from the CLS data set for experiments. The structure of the data sets used in the experiments is shown in Tables 4 and 5, according to which the sample data are extracted from the CLS data sets. The attack behavior data of data set $D1 - D10$ all contain several unknown attack behavior data of corresponding class number (disguised by known attack behavior), which is used for the comparative experiments of the performance of intrusion detection algorithm to detect unknown attack behavior.

5.3.1. Experiments of ACC and FAR. Intrusion detection algorithms based on traditional KNN classification and density clustering DBSCAN are compared with the proposed algorithm. When the number of participants R is 2, 4, 6, 8, and 10, ten test data sets $H1$, $H2$, $H3$, $H4$, $H5$, $H6$, $H7$, $H8$, $H9$, and $H10$ are randomly selected from the test data packets

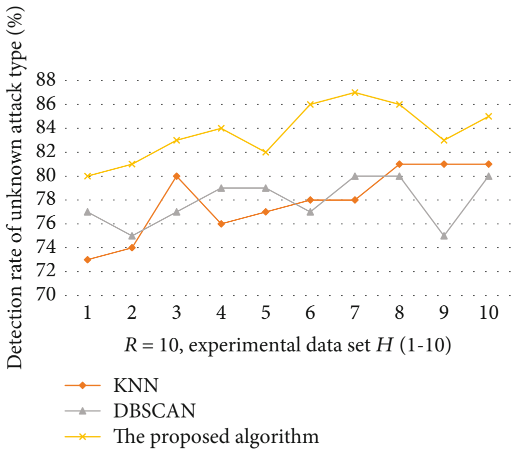


Figure 13: Detection rate of detecting unknown attack types, $R = 10$.

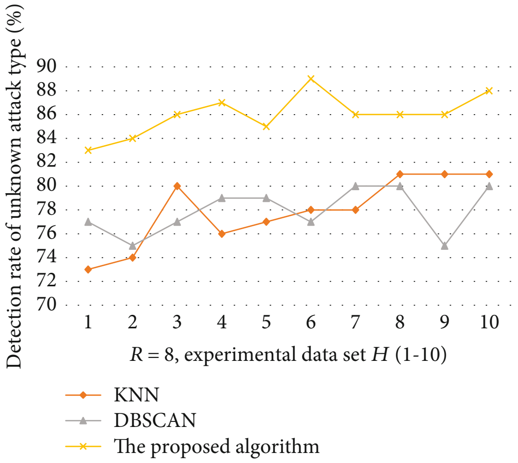


Figure 14: Detection rate of detecting unknown attack types, $R = 8$.

for comparative experiments. The experimental results are shown in Figures 8–12.

5.3.2. Experiments of Detecting Unknown Attack Types. For detecting the unknown attack types, the proposed algorithm based on Federated Learning is compared with intrusion detection algorithms based on traditional KNN classification and density clustering DBSCAN. When the number of participants R is 2, 4, 6, 8, and 10, ten test data sets $D1$, $D2$, $D3$, $D4$, $D5$, $D6$, $D7$, $D8$, $D9$, and $D10$ are randomly selected from the test data packets for comparative experiments. The experimental results are as shown in Figures 13–17.

Through the above comparative experiments, the results show that with the increasing number of participants, the detection performance of the proposed algorithm based on Federated Learning in terms of detection rate, false detection rate, and other aspects maintains at a relatively stable level. It fully verifies the feasibility of this detection algorithm in the real network environment where local data is protected and training data is scarce. Compared with the intrusion detection algorithms based on traditional KNN classification and density clustering DBSCAN, our algorithm has significant improvement in detection rate, false detection rate, and detection of unknown attack behavior. The AWID data set can well represent the characteristics of the original attributes after dimensionality reduction by the PCA method, which can achieve dimensionality reduction, reduce the time

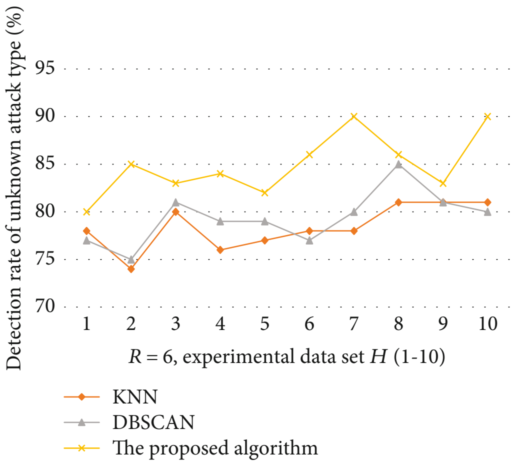


Figure 15: Detection rate of detecting unknown attack types, $R = 6$.

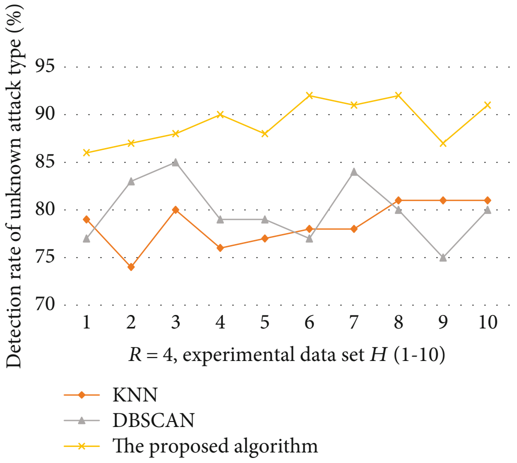


Figure 16: Detection rate of detecting unknown attack types, $R = 4$.

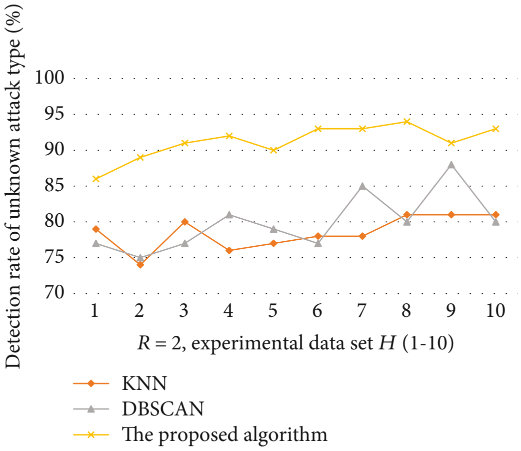


Figure 17: Detection rate of detecting unknown attack types, $R = 2$.

complexity, improve the detection efficiency, and ensure a higher detection rate and a lower false detection rate.

6. Conclusion

To allow multiple participants to conduct cooperative training on a global model without sharing their private data, protect participants' local data, and expand the amount of data in the training model, this paper proposes an improved k -means clustering intrusion detection algorithm based on Federated Learning for the wireless network. This algorithm is combined with three-way decision ideas and introduced a multiple perspectives cosine distance as the similarity measure between data objects to improve and modify the k -means clustering algorithm. Therefore, the clustering result is more reasonable and the network data behavior is determined more accurately. As a result, the detection rate of the algorithm is increased and the error detection rate is decreased. This algorithm, however, is assumed under the ideal condition of unobtrusive parameter transmitted between participants and server, which may be different from that in the real network environment. Moreover, the scale of reference point set selected by the multiple perspective method in this paper is so large that it will affect its overall performance. In the future, we will continue to improve the algorithm structure and use more appropriate data sets to train the classifier, thus, ensuring the security of the interaction between the participants and the server. We will also find a more reasonable and effective way to select the reference point and to reduce the dimensionality of experimental data to further reduce its time complexity and improve its overall performance.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable suggestions given to improve the quality of the manuscript significantly. This work was supported by the National Natural Science Foundation of China under Grants no. 62076088, Natural Science Foundation of Hebei Province of China under Grant no. F2019205163 and no. F2021205004, Science Foundation of Returned Overseas of Hebei Province of China under Grant no. C2020342, and the Technological Innovation Foundation of Hebei Normal University under Grant no. L2020K09.

References

[1] S. Gavel, A. Raghuvanshi, and S. Tiwari, "A novel density estimation based intrusion detection technique with Pearson's divergence for wireless sensor networks," *ISA Transactions*, vol. 111, pp. 180–191, 2021.

[2] T. Chenghua, L. Pengcheng, T. Shensheng, and X. Yi, "Anomaly intrusion behavior detection based on fuzzy clustering and features selsection," *Journal of Computer Research and Development*, vol. 52, no. 3, pp. 718–728, 2015.

[3] H. Liu, C. Zhong, A. Alnusair, and S. R. Islam, "A framework for enhancing AI explainability of intrusion detection results using data cleaning techniques," *Journal of Network and Systems Management*, vol. 29, no. 4, 2021.

[4] J. Bozic, D. Tabernik, and D. Skocaj, "Mixed supervision for surface-defect detection: from weakly to fully supervised learning," *Computers in Industry*, vol. 129, article 103459, 2021.

[5] C. Zhuang, B. Zhang, J. Hu, Q. Li, and R. Zeng, "Anomaly detection for power consumption patterns based on unsupervised learning," *Proceedings of the CSEE*, vol. 36, no. 2, pp. 379–387, 2016.

[6] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: concept and applications," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 2, pp. 1249–1258, 2019.

[7] R. Wang, C. Ma, and P. Wu, "An intrusion detection method based on federated learning and convolutional neural network," *Netinfo Security*, vol. 20, no. 4, pp. 47–54, 2020.

[8] Y. Zhao, L. Wang, J. Chen, and J. Teng, "Network anomaly detection based on federated learning," *Journal of Beijing University of Chemical Technology (Natural Science Edition)*, vol. 48, no. 2, pp. 92–99, 2021.

[9] X. Wei, Z. Zhang, B. Song, Y. Mao, and A. Ban, "Social networks cross-platform malicious user detection method based on vertical federated learning," *Journal of Chinese Computer Systems*http://kns.cnki.net/kcms/detail/21.1106.TP.20210506.1032.008.html.

[10] F. Li, J. Cheng, and Y. Qian, "Whole-granulation cluster algorithm," *Journal of Nanjing University(Natural Science)*, vol. 50, no. 4, pp. 505–516, 2014.

[11] W. Ting, W. Na, C. Yunpeng, and L. Huan, "The optimization method of wireless network attacks detection based on semi-supervised learning," *Journal of Computer Research and Development*, vol. 57, no. 4, pp. 791–802, 2020.

[12] J. Guan and D. Liu, "Unsupervised anomaly detection based on principal components analysis," *Journal of Computer Research and Development*, vol. 9, pp. 1474–1480, 2004.

[13] X. Qu, S. Wang, Q. Hu, and X. Cheng, "Proof of federated learning: a novel energy-recycling consensus algorithm," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 8, pp. 2074–2085, 2021.

[14] L. Wang, J. Yang, X. Xu, and P.-J. Wan, "Mining network traffic with the k-means clustering algorithm for stepping-stone intrusion detection," *Wireless Communications & Mobile Computing*, vol. 2021, article 6632671, pp. 1–9, 2021.

[15] R. Liu, J. Zheng, G. Shen, and Z. Liu, "Study on parallel k-means algorithm based on CUDA," *Computer Science*, vol. 45, no. 11, pp. 292–297, 2018.

[16] K. Zhang, J. Dai, and J. Zhan, "A new classification and ranking decision method based on three-way decision theory and TOPSIS models," *Information Sciences*, vol. 568, pp. 54–85, 2021.

[17] T. Wang, H. Li, X. Zhou, D. Liu, and B. Huang, "Three-way decision based on third-generation prospect theory with Z-numbers," *Information Sciences*, vol. 569, pp. 13–38, 2021.

[18] J. Deng, J. Zhan, and W. Wu, “A three-way decision methodology to multi-attribute decision-making in multi-scale decision information systems,” *Information Sciences*, vol. 568, pp. 175–198, 2021.

[19] Q. Liu, H. Shi, and X. Yang, “Three-way clustering analysis based onεneighborhood,” *Computer Engineering and Applications*, vol. 55, no. 6, pp. 140–144, 2019.

[20] Y. Qian, Y. Li, and X. Yu, “Intrusion detection method based on multi-label and semi-supervised learning,” *Computer Science*, vol. 42, no. 2, pp. 134–136, 2015.

[21] P. Kannari, N. Shariff, and R. Biradar, “Network intrusion detection using sparse autoencoder with swish-PReLU activation model,” *Journal of Ambient Intelligence and Humanized Computing*, 2021.

[22] A. Davahli, M. Shamsi, and G. Abaei, “Hybridizing genetic algorithm and grey wolf optimizer to advance an intelligent and lightweight intrusion detection system for IoT wireless networks,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 11, pp. 5581–5609, 2020.

[23] J. Liu, Y. Gao, and F. Hu, “A fast network intrusion detection system using adaptive synthetic oversampling and LightGBM,” *Computers & Security*, vol. 106, article 102289, 2021.

[24] L. He, Z. Xu, Y. Li, and C. Shen, “Conditional-probability zone transformation coding for categorical features,” *Application Research of Computers*, vol. 37, no. 5, pp. 1400–1405, 2020.

[25] A. Blaise, M. Bouet, V. Conan, and S. Secci, “Detection of zero-day attacks: an unsupervised port-based approach,” *Computer Networks*, vol. 180, article 107391, 2020.

[26] X. Chen, L. Wang, Q. Gu, Z. Wang, and C. Ni, “A surey on cross—project software defect predicton methods,” *Chinese Journal of Computers*, vol. 41, no. 1, pp. 254–274, 2018.

[27] K. Pradeep Mohan Kumar, M. Saravanan, M. Thenmozhi, and K. Vijayakumar, “Intrusion detection system based on GA-fuzzy classifier for detecting malicious attacks,” *Concurrency and computation-practice & experience*, vol. 33, no. 3, article e5242, 2021.

[28] C. Kolias, G. Kambourakis, A. Stavrou, and S. Gritzalis, “Intrusion detection in 802.11networks: empirical evaluation of threats and a public dataset,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 184–208, 2016.

[29] Z. Hu, L. Wang, L. Qi, Y. Li, and W. Yang, “A novel wireless network intrusion detection method based on adaptive synthetic sampling and an improved convolutional neural network,” *IEEE Access*, vol. 8, pp. 195741–195751, 2020.