Hindawi
Security and Communication Networks
Volume 2019, Article ID 6346708, 11 pages
https://doi.org/10.1155/2019/6346708

Research Article

A Bayesian Classification Intrusion Detection Method Based on the Fusion of PCA and LDA

Zhidong Shen, Yuhao Zhang, and Weiying Chen

Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University, Wuhan, China

Correspondence should be addressed to Zhidong Shen; shenzd@whu.edu.cn

Received 15 August 2019; Revised 23 October 2019; Accepted 5 December 2019; Published 28 December 2019

Academic Editor: Kaitai Liang

The rapid development of network technology is facing severe security threats while bringing convenience to people. How to build a secure network environment has become an important guarantee for social development. Intrusion detection plays an important role in the field of network security. With the complexity and diversification of networks, intrusion detection systems also need to be constantly improved and developed to match external environmental changes. The innovative work of this paper is as follows: principal component analysis and linear discriminant analysis are used to reduce the dimensionality of the data set, which avoids unnecessary detection content and improves detection efficiency and accuracy. The principal component analysis method, linear discriminant analysis algorithm, and Bayesian classification are combined to construct the PCA-LDA-BC classification algorithm, and the intrusion detection model is established based on this algorithm. The simulation experiment was carried out on the algorithm CICIDS2017 data set proposed in this paper. From the experimental results, it can be analysed that in the intrusion detection of missing data, the improved algorithm is compared with the traditional naive Bayesian classification algorithm, the detection rate is improved, and the false detection rate and the missed alarm rate are reduced. In terms of intrusion detection for various types of attacks, the detection rate, false detection rate, and missed alarm rate have been improved accordingly. It is proved that the algorithm has certain validity and feasibility.

1. Introduction

Network security technology has a long history, and has been gradually transformed from a traditional passive defense technology (firewall) to an active security defense technology (intrusion detection technology). Most of the evaluations of an intrusion detection product are analysed from the following three aspects: effectiveness, adaptability, and timeliness. Due to the large amount of network data traffic, network attacks are extremely frequent and fast and it is difficult to find them in time. Most of the current network security products rely on network security experts for manual operations, which are not only difficult to deal with unknown attacks, but also cannot meet the actual needs in terms of detection rate and timeliness. Therefore, combining the current popular artificial intelligence technology and data mining technology, it is very urgent and necessary to realize the automatic operation of the intrusion detection system.

The purpose of intrusion detection is to accurately classify normal events and abnormal events in massive unknown network event data, so as to find network attack events and reduce the false alarm rate. Intrusion detection technology can generally be divided into two kinds: misuse detection and anomaly detection. Misuse detection refers to predefining intrusion patterns according to known attack methods and completing detection tasks by judging whether these intrusion patterns will appear. The disadvantage of misuse detection is that it is limited to the detection range of existing knowledge and cannot detect attacks beyond existing knowledge. Anomaly detection refers to the use of resources or the behaviour of users to determine whether they have been invaded, rather than the specific behaviour as a detection criterion. Relatively speaking, the applicability of anomaly detection is relatively strong and it can detect unfamiliar attacks, unlike misuse detection, which is not limited by known attack means and its main defect is the

false detection rate. Especially in the environment, where many users, working conditions, system parameters, network structure, and other factors are constantly changing. At present, more effective intrusion detection classification models have been proposed.

For example, in literature [1], a hybrid intrusion detection model combining misuse detection and anomaly detection is proposed. In [2–7], other learning algorithms are applied to intrusion detection, such as support vector machine, genetic algorithm, and artificial neural network.

At present, the application of data mining technology in the field of IDS is emerging. The literature [8–12] uses data mining technology to detect intrusion data. In literature [13], the minimum intraclass spread in Fisher discriminant analysis is combined with the traditional support vector machine (SVM) and a minimum intraclass spread support vector machine (WCS-SVM) is proposed, which is better than the traditional support vector machine. Literature [14] applies the backpropagation artificial neural network model to intrusion detection, making the intrusion detection system more adaptive to the new environment and responding to new types of attacks. The research in the literature [15–20] has a higher detection rate and a lower false alarm rate and the combination of clustering and classification can achieve better results. Literature [21, 22] used fuzzy cluster analysis to classify the data and achieved good results. The literature [23, 24] used vector machines to implement intrusion detection algorithms, and the literature [25–27] uses the neighborhood algorithm for classification. Literature [28] used data dimensionality reduction methods to analyse data features.

In order to improve the performance of the Naive Bayesian classifier and make it more suitable for intrusion detection in ICS networks, based on previous research, an improved Naive Bayesian classification algorithm (PCA-LDA-BC algorithm) is proposed in this paper. This algorithm adds a comprehensive weighting coefficient to the traditional Naive Bayesian classification model, which integrates covariance theory and weighting coefficient. The weighting coefficients proposed in [4] make up for the shortcoming that only the frequency relation of attributes is considered in document [4], but the influence of the content of attributes on classification is neglected, which makes the original concise and efficient algorithm in document [4] more perfect.

From the point of view of the whole structure flow of pattern recognition, the purpose of feature selection is to get the most information features while retaining the classification information as much as possible and to delete those information or features that are not conducive to classification or have little use. Another purpose of feature selection is to speed up the processing speed. In terms of features, intrusion detection data have the characteristics of multidimensionality, but intrusion behaviour is often concentrated in a few attribute items. The existence of redundant attributes will reduce the effectiveness and efficiency of intrusion detection, so it is necessary to extract features from intrusion data. Principal component analysis (PCA) and linear discriminant analysis (LDA) can reduce the dimension of sample space, so that matching and recognition can be carried out in low-dimensional subspace.

2. Bayesian Classification Overview

2.1. Application of Bayesian Classification in Intrusion Detection System. At present, the more common model is based on Wenke Lee's intrusion detection classification idea [29], as shown in Figure 1.

The naive Bayesian intelligent intrusion detection model is based on the number of network connection records. The network connection records are divided into training sets and test sets. Through the training set learning, the naive Bayesian classification model is obtained and then tested. Test on the set to verify the accuracy of the model.

The Naive Bayes-based intrusion detection model consists of two parts: training and detection. The method of establishing an intrusion detection model is to represent the records in the network connection by n-dimensional vectors $(A_1, A_2, \ldots, A_n)$, where n represents the n characteristic attributes of the connection record. Since the structure of the naive Bayes network is static, it is here. In this case, only the conditional probability of each node needs to be calculated.

2.2. Naive Bayesian Classification Model. Although the Naive Bayesian classification algorithm uses simple classification with relatively high accuracy and less time to predict and learn than other classification algorithms, it has an ideal assumption that the impact of each attribute on a given class is independent of other attributes, which is difficult to satisfy in reality.

In order to improve the performance of the Naive Bayesian classifier and make it more suitable for network intrusion detection, based on previous research, this paper proposes a Naive Bayesian classification algorithm which improves the comprehensive weighting coefficient. This algorithm adds a comprehensive weighting coefficient to the traditional Naive Bayesian classification model. The comprehensive weighting coefficient combines covariance theory and weighting. Coefficient makes up for the previous literature only considering the frequency relationship of attributes, while ignoring the impact of the content of attribute values on classification makes the original concise and efficient algorithm more perfect.

2.2.1. Covariance Attribute Weighting Coefficient. In practical application, different attributes of things have different effects on the classification of things. According to the influence degree of attributes, they can be divided into conditional attributes and decision attributes. Decision attributes refer to the attributes that have a significant impact on classification. Conditional attributes refer to the remaining attributes. In addition, the degree of correlation between different conditional attributes and decision attributes is also different. The system consisting of the decision attribute X and the conditional attribute Y reflects the relativity between the attributes X and Y. The larger the system ρ, the greater the influence of the conditional attribute Y on

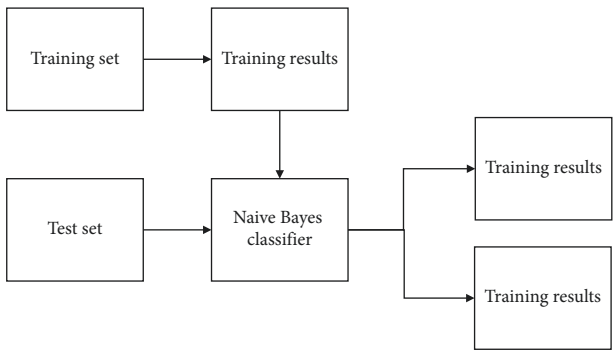

Figure 1: Intrusion detection system model.

the decision attribute X, and vice versa. The formula of correlation coefficient between attributes is as follows:

$$\rho_1 = \frac{\mathrm{Cov}(X,Y)}{\sqrt{D(X)}\sqrt{D(Y)}}. \tag{1}$$

2.2.2. Improving the Comprehensive Weighting Coefficient. According to the idea of setting weighting coefficients according to the influence of different attribute values on classification results, a new method of weighting calculation is proposed in this paper. N_{Z_m} denotes the number of sample objects whose attribute Z_m takes value, $N(Z_m = k)$ denotes the number of sample objects whose attribute Z_m takes value k, and $N_{(Z_m=k\cap C_i)}$ denotes the number of sample objects whose attribute Z_m takes value K and belongs to class C_i. The weighted coefficient formula is expressed as follows:

$$\rho_2 = \frac{N_{(Z_m=k\cap C_i)}/N_{(Z_m=k)}}{N_{Z_m}}. \tag{2}$$

The original Bayesian classification calculates the weights according to the influence of different values of each attribute on the classification, but it considers the frequency relationship of the attribute values and does not consider the influence of the content of the attribute values on the classification. Covariance theory mainly uses the content of attribute values to express the correlation between attributes. Therefore, by combining the two methods, more reasonable and accurate weighting coefficients can be obtained.

Therefore, the improved comprehensive weighting coefficient is defined as

$$\rho = \frac{\rho_1 + \rho_2}{2}. \tag{3}$$

3. Data Dimensionality Reduction Method

3.1. Principal Component Analysis

3.1.1. Traditional Principal Component Analysis (PCA). In the intrusion detection system, it is often necessary to analyse dozens or even hundreds of features. The resulted huge computational complexity makes the traditional intrusion detection method have poor real-time performance. The method of PCA dimensionality reduction can greatly improve the intrusion detection performance. The basis for being able to use the PCA algorithm is that although there are many influencing factors to be considered, a large number of influencing factors have a certain degree of correlation, so there will be some overlap in the statistical data. In addition, the importance and impact of each influencing factor are different. We certainly expect fewer variables to be involved as much as possible in the process of training the sample and using the algorithm and get as much information as possible. Therefore, the principal component analysis algorithm has become a key method to solve this problem. There is a certain similarity between the many influencing factors. Through the study of the matrix structure of the original variables, we can find the comprehensive index parameters that can play a key role and replace the original index parameters with the comprehensive indicators, not only retaining the original data. The main information can also make each indicator unrelated and independent, which can make us grasp the main factors in the analysis of complex problems.

Let the n-dimensional vector w be a mapping vector of the target space and maximize the variance obtained by the mapping by the following formula:

$$\max_w \frac{1}{m-1}\sum_{i=1}^{m}\left(w^T\left(x_i - \overline{x}\right)\right), \tag{4}$$

where m represents the number of instances, x_i represents the vector representation of instance i, $\overline{x}$ represents the average vector, and W represents a matrix, which includes all the mapping vectors as column vectors. After performing linear algebra transformation, the following objective functions can be obtained:

$$\begin{aligned} \min_w \quad & \mathrm{tr}\left(W^T A W\right) \\ \text{s.t.} \quad & W^T W = I, \end{aligned} \tag{5}$$

where tr represents the trace of the matrix and A is the data covariance matrix, as shown in the following formula:

$$A = \frac{1}{m-1}\sum_{i}^{m}\left(x_i - \overline{x}\right)\left(x_i - \overline{x}\right)^T. \tag{6}$$

3.1.2. Improved Principal Component Analysis. The improved PCA algorithm is based on the classical PCA algorithm, aiming at the shortcoming of the classical PCA algorithm that is sensitive to light. The algorithm reduces the impact by weighting the first three eigenvectors. The first three feature vectors of the classical PCA method embody the whole information of the image. At the same time, the doctoral thesis of Kepenekci [30] mentions that when the illumination condition is obvious, the first three principal components of the feature may be most polluted and processing them will improve the recognition rate to a certain extent. Enlightened by this, the first three principal components in the data set are weighted to reduce their proportion in the recognition stage. The data are transformed

into a new matrix composed of the K eigenvectors mentioned above:

$$Y = (Y_1, Y_2, Y_3, Y_4, \ldots, Y_k),$$
$$Y' = (kY_1, kY_2, kY_3, Y_4, \ldots, Y_k). \quad (7)$$

3.2. Linear Discriminant Analysis Algorithm (LDA). The principle of the multiclass linear discriminant analysis algorithm is described as follows.

Suppose the data set D

$$D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}, \quad (8)$$

define the i-dimensional vector as a sample, $x_i, y_i \in \{C_1, C_2, \ldots, C_k\}$. We define the number of samples of class j as $N_j (j = 1, 2, \ldots, k)$. We define the set of samples of class j as $X_j = (j = 1, 2, \ldots, k)$. We define the mean vector of class j samples as $\mu_j (j = 1, 2, \ldots, k)$. We define the covariance matrix of class j samples as $\sum_j (j = 1, 2, \ldots, k)$. If the dimension projected into the low-dimensional space is d, then the corresponding base vector is $(w_{1,} w_2, \ldots, w_d)$ and we can form a matrix W of $n \times d$.

The optimization goal is

$$\frac{W^T S_b W}{W^T S_\omega W}, \quad (9)$$

$$S_b = \sum_{j=1}^{k} N_j (\mu_j - \mu)(\mu_j - \mu)^T, \quad (10)$$

$$S_\omega = \sum_{j=1}^{k} S_{\omega j} = \sum_{j=1}^{k} \sum_{x \in X_j} (x - \mu_j)(x - \mu_j)^T. \quad (11)$$

And in Formula (10), μ is the mean vector. It can be seen from the above formula that both $W^T S_b W$ and $W^T S_\omega W$ are not scalars, but belong to the matrix, so it is necessary to optimize the scalar function. For problems that cannot directly use the second-class linear discriminant method, we can use other methods to achieve.

In general, the multiclass optimization objective function formula of linear discriminant analysis is as follows:

$$\arg\max J(W) = \frac{\prod_{\text{diag}} W^T S_b W}{\prod_{\text{diag}} W^T S_w W,} \quad (12)$$

where $\prod_{\text{diag}} A$ represents the product of the main diagonal elements of matrix A and W is a matrix of $n \times d$.

The optimization process of $J(W)$ can be expressed as

$$J(W) = \frac{\prod_{i=1}^{d} w_i^T S_b w_i}{w_i^T S_w w_i} = \prod_{i=1}^{d} \frac{w_i^T S_b w}{w_i^T S_w w}. \quad (13)$$

Let $X_k^{(i)}$ be the I-th test sample of the kth ($k = 1, 2, \ldots, C$) class and μ_k be the sample average of the kth class, then the interclass divergence matrix can be expressed as

$$S_B = \frac{1}{L} \sum_{k=1}^{q} \left[N_k (\mu_k - \mu)(\mu_k - \mu)^T \right]. \quad (14)$$

The intraclass divergence matrix can be expressed as

$$S_W = \frac{1}{L} \sum_{k=1}^{C} \sum_{i=1}^{N_k} \left[\left(X_k^{(i)} - \mu_k \right) \left(X_k^{(i)} - \mu_k \right)^T \right]. \quad (15)$$

Among them, N_k is the number of samples in the kth class.

3.3. Reasons for Merging of PCA and LDA. According to the above description, it can be seen that the principal component analysis algorithm focuses on expressing the original data features, but does not reach the category information of the effectively utilized samples. Therefore, the dimensionality reduction results obtained by using the principal component analysis algorithm are only the features that best represent the original data and are not the most resolving features. The linear discriminant analysis algorithm mainly selects the most discriminative low-dimensional data from the high-dimensional space. Therefore, by combining the principal component analysis algorithm and the linear discriminant algorithm, it is possible to retain both the original data features and the strong discriminative power to the greatest extent.

4. PCA-LDA-BC Intrusion Detection Algorithm

In this paper, combined with principal component analysis, linear discriminant analysis, and naive Bayesian classification algorithm, the fusion algorithm is proposed: PCA-LDA-BC algorithm. In essence, the intrusion detection algorithm actually designs a classifier to classify the collected data into two categories: normal and abnormal. When designing this classifier, we need to pay attention to which attributes can be used as filtering and classification conditions. In the CICIDS2017 dataset [31], there are many features that are related to each other. If all features are used for intrusion detection, it is inevitable that the computational complexity is too large and unnecessary. We only need to select several feature dimensions with the highest contribution rate to compare, and the features with a large contribution rate are more representative.

The basic processing steps of the model are as follows:

(1) Preprocessing the original data set, CICIDS2017 contains millions of data and takes 10% of the data through a random algorithm to discretize the continuous values.

(2) The obtained new data set is used as the training data set, and step (3) is performed. If the sample data needs to be classified, the step (4) is directly performed.

(3) The principal component analysis algorithm (Algorithm 1) and the linear discriminant algorithm

(1) Remove the average value and subtract every element in matrix X from μ_i, $[\mu_i]$ is the mean value of the dimension i of the sample matrix (matrix column element), $\mu_i = (1/m)\sum_{j=1}^{m} x_{ji}$.
(2) Computing covariance matrix C of matrix X, $C = (1/m)X^T X$.
(3) Calculate the eigenvalues and eigenvectors of the covariance matrix C.
(4) Sort them according to the numerical value of the eigenvalues, retaining the first k eigenvectors.
(5) It is generally believed that the first three eigenvectors are the most representative of the overall information. When the data are affected, the three eigenvectors are most susceptible to contamination. If weighting is performed, the degree of influence can be reduced, the accuracy can be improved, and the data can be converted to the above. A new matrix of k eigenvectors $Y = (Y_1, Y_2, Y_3, Y_4, \ldots, Y_k)$ and $Y' = (kY_1, kY_2, kY_3, Y_4, \ldots, Y_k)$ is defined as in (7).
(6) $X' = X^T * Y'$.)

Algorithm 1

(1) Using matrix Y, calculate the intraclass divergence matrix $S_w = \sum_{i=1}^{L} \sum_{x_j \in w_k} (u_i - X_i)(u_i - X_i)^T$ defined as in (15)
(2) Using matrix Y, calculate the interclass divergence matrix $S_b = \sum_{i=1}^{L} P(i)(u_i - u)(u_i - u)^T$ defined as in (14)
$P(i)$ is a prior probability, and u is a mean vector
(3) Perform eigenvalue decomposition on the matrix
(4) Take the eigenvector corresponding to the largest d eigenvalues
(5) Composition of a new matrix $V = [V_1, V_2, \ldots, V_d]$

Algorithm 2

(1) Use a feature vector X to represent matrix V.
(2) Generally, classify X into the class with the largest posterior probability value, which is essentially the maximum value of $P(C_i \mid X)$, which is $P(C_i \mid X) = P(X \mid C_i)P(C_i)/P(X)$.
(3) To obtain the maximum value of $P(Ci \mid X)$, you only need to maximize $P(C_i \mid X)P(C_i)$. If the prior probabilities are not known, they are generally considered to be equally probabilistic, i.e., $P(C_1) = P(C_2) = \cdots = P(C_n)$. Otherwise, the knowledge based on probability can be calculated from the prior probability formula: $P(C_i) = S_i/S$, where S_i is the number of training samples and S is the total number of training samples.
(4) Define formula: $\rho = \rho_1 + \rho_2/2$, $\rho_1 = \mathrm{Cov}(X, Y)/D(X)\sqrt{D(Y)}$, and $\rho_2 = (N_{(Z_m=k\cap C_i)}/N_{(Z_m=k)})/N_{Z_m}$ defined as in (1)–(3).
That is, the linear correlation coefficients of X and Y. $D(X)$ and $D(Y)$ are variances of X and Y. For intrusions coded as Smurf, $\rho(\mathrm{dst\_host\_srv\_rerror\_rate}, \mathrm{dst\_host\_rerror\_rate})$ is calculated and the attributes of other attacks are selected, and so on. N_{Z_m} denotes the number of sample objects with attribute Z_m, $N(Z_m = k)$ denotes the number of sample objects with attribute Z_m, and $N_{(Z_m=k\cap C_i)}$ denotes the number of sample objects with attribute Z_m belonging to class C_i.
(5) $P(C_i \mid Z) = \arg\max \prod_{j=1}^{m} \rho_j P(z_j \mid C_i)$.
(6) In addition, the effect will be very unsatisfactory in the environment of small amount of data or high probability of some remote data. This is because the Naive Bayesian formula is a continuous product, so in order to improve the classification effect, we can change the continuous product into a continuous sum: that is to say, change $P(C_i)\prod_{j=1}^{m} \rho_j P(z_j \mid C_i)$ to $P(C_i)\sum_{j=1}^{m} \rho_j P(z_j \mid C_i)$.
(7) In the case of a large number of attribute sets, in order to save the cost of computing time, it is generally assumed that the class conditions are independent of each other, that is, the individual attribute values are independent of each other,
$P(X \mid C_i) = P(x_k \mid C_i)$.
If A_k is a discrete attribute, the probability can be calculated by the formula: $P(x_n \mid C_i) = s_i k_i$.
$s_i k_i$ indicates that A_k takes the value of x_i and belongs to the number of training samples of C_i, while s_i represents the total number of training samples in C_i.

Algorithm 3

(Algorithm 2) are used to reduce the features in the data set.

(4) The obtained dimensionality reduction data set is subjected to Bayesian classification (Algorithm 3).

(5) The data set is classified by the new model to obtain the classification result.

Related algorithms in the model are given in (Algorithm 1), (Algorithm 2), and (Algorithm 3).

5. Simulation Experiment and Analysis

5.1. Intrusion Detection Data Set. The 1998 Pentagon DARPA Intrusion Detection and Evaluation Program was produced by the MIT Lincoln Laboratory. The goal is to provide a standard set of data for evaluating intrusion detection studies that include various intrusions simulated in a military network environment. The 1999 Data Mining and Knowledge Discovery Competition used a version of the data set and named it CICIDS2017 [31].

Like the KDD CUP99, the CICIDS2017 dataset simulates data from real networks. CICIDS2017 has more complex data descriptions and more data set. CICIDS2017 uses 85 feature dimensions to describe the activity of a network connection. For example, 41 of them are consistent with KDD CUP99 [32]. In order to compare with KDD CUP99, we selected these 41 feature dimensions for testing. The data feature dimensions and examples are shown in Table 1.

Among them, each data 1~84 dimension is a description of the data feature, and the 85th dimension is a Label which means the identification whether the data is normal or not, for example, BENIGN means normal data, Web Attack identifies a specific network attack, PortScan is a port scan intrusion, etc.

5.2. Preprocessing of Data Set. This article uses the CICIDS2017 dataset, which is a relatively new data set currently published by the Canadian Institute for Cybersecurity in a format similar to kdd99. In this data set, the connection is described by 81 features written in the CSV format, with the final label added, and source flow data for researchers' backtracking and other research projects.

In the above data set, the performance of the system depends to a large extent on the number of inputs and their quality. If these inputs are not standardized, the performance of the system will be degraded. Using normalization can reduce the size of the data set and can also greatly reduce processing time. The specific operations are as follows:

(1) Standardization: properties of the text format must be converted to numeric values.

Let X'_{ij} be the value standardized by the value X_{ij}

$$X'_{ij} = \frac{X_{ij} - \mathrm{AVG}_j}{\mathrm{STAD}_j},$$

$$\mathrm{AVG}_j = \frac{1}{n}\left(x_{1j} + x_{2j} + \cdots + x_{nj}\right),$$

$$\mathrm{STAD}_j = \frac{1}{n}\left(\left|x_{1j} - \mathrm{AVG}_j\right| + \left|x_{2j} - \mathrm{AVG}_j\right| + \cdots + \left|x_{nj} - \mathrm{AVG}_j\right|\right). \tag{16}$$

(2) Normalization: attributes with very high range must be scaled to [−1,1] or [0,1] for best performance.

Let X''_{ij} be the value normalized by the value X'_{ij}

$$X''_{ij} = \frac{X'_{ij} - X_{\min}}{X_{\max} - X_{\min}},$$

$$X_{\min} = \min\left\{X'_{ij}\right\}, \tag{17}$$

$$X_{\max} = \max\left\{X'_{ij}\right\}.$$

5.3. Experimental Environment Construction. The experimental environment uses windows7 system, Intel(R) Core(TM) i3-2120 CPU, 4G memory. The raw data are randomly selected 10% of the data, the encoding uses Python2.7 as the development language, and pycharm5.0.3 is used as the IDE.

5.4. Experimental Results and Analysis. Figure 2 shows the process of the PCA-LDA-BC algorithm.

After dimensionality reduction, the experimental results of using different dimension features for intrusion detection are as follows and the results are displayed in the following order: correct rate, recall rate, false detection rate, accuracy rate, F value, average information amount, detection time, and contribution rate of each feature.

The definition is as shown in the following equations:

$$\text{Correct rate: } \mathrm{CR} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}, \tag{18}$$

$$\text{Recall rate: } \mathrm{RC} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{19}$$

$$\text{False detection rate: } \mathrm{MR} = \frac{\mathrm{FP}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}, \tag{20}$$

$$\text{Accuracy: } \mathrm{PC} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{TN}}, \tag{21}$$

$$\text{Value of } F\text{: } F = \frac{2 * \mathrm{PC} * \mathrm{RC}}{\mathrm{PC} + \mathrm{RC}}, \tag{22}$$

$$\text{Detection time: } T_{\mathrm{all}} = T_{\mathrm{train}} + n \times T_{\mathrm{classify}}. \tag{23}$$

In the above definition formulas, the meaning of TP is that the machine is classified into the correct sample and is the number of normal samples at the beginning. The meaning of TN is classified as an abnormal sample and is the number of abnormal samples at the beginning, and the meaning of FN is classified as abnormal samples but it is the number of normal samples at the beginning; the meaning of FP is classified as the correct sample, but it is the number of abnormal samples at the beginning; T_{all} means training time, n means the number of samples, and the meaning of T_{classify} is the time required to classify each sample. Table 2 shows the complete test results of Gauss Bayes. Table 3 shows the complete test results for the algorithm without improvement. Table 4 shows the complete test results using Bernoulli Bayes. Figure 3 shows a comparison of the relationship between dimensions and accuracy. Figure 4 shows the relationship between dimensions and time.

Based on the above experimental results, we can draw the following column chart to show the results of dimensionality reduction and no dimensionality reduction, as shown in Figure 5.

Based on the above different sets of intrusion detection results, we analyse the following five aspects:

(1) *Accuracy rate.* As the reduced dimension continues to increase, the best results are achieved in 8 dimensions and the accuracy is not the same as without dimensionality reduction. But time performance is better

(2) *The rate of false detection*. In the 8D, the false detection rate is slightly more than 1% when there is no dimensionality reduction

Table 1: Description of CICIDS2017.

Dimension	1	2	3	⋯	84	85
Description	Flow ID	Source IP	Source port	⋯	Idle min	Label
E.g. 1	192.168.*.*	192.168.*.*	33898	⋯	6737603	BENIGN
E.g. 2	172.16.*.*	172.16.*.*	44380	⋯	0	Web Attack
E.g. 3	172.16.*.*	172.16.*.*	58812	⋯	0	PortScan

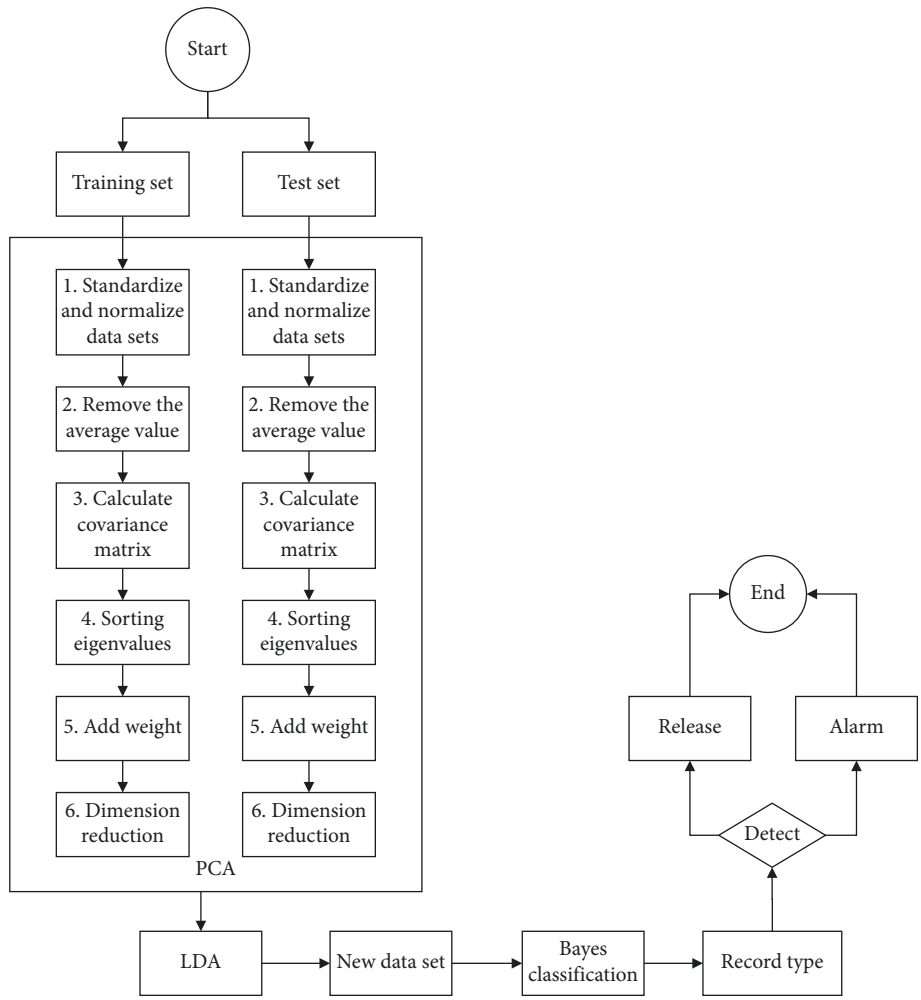

Figure 2: Flow chart of PCA-LDA-BC model.

Table 2: Complete test results of Gauss Bayes.

Dimension	41	40	39	38	37	36	35	34	33	32	31
Accuracy	0.95	0.95	0.94	0.94	0.91	0.94	0.96	0.86	0.90	0.91	0.84
Time	38.7	32.5	32.5	31.4	30.0	29.6	29.3	29.1	33.2	27.7	18.4
Dimension	30	29	28	27	26	25	24	23	22	21	20
Accuracy	0.90	0.90	0.89	0.86	0.88	0.83	0.84	0.87	0.91	0.88	0.88
Time	30	24	23	23	22	22	21.6	20.3	20.1	19.6	19
Dimension	19	18	17	16	15	14	13	12	11	10	9
Accuracy	0.84	0.86	0.86	0.85	0.87	0.90	0.90	0.90	0.92	0.94	0.96
Time	18.6	18.5	17.3	16	15.8	14.7	14	13.4	13	15	15.3
Dimension	8	7	6	5	4	3					
Accuracy	0.97	0.95	0.94	0.95	0.94	0.9					
Time	14.3	13.8	15.3	11.6	10.9	10					

Table 3: Complete test results for the algorithm without improvement.

Dimension	41	40	39	38	37	36	35	34	33	32	31
Accuracy	0.94	0.94	0.92	0.92	0.91	0.91	0.90	0.83	0.90	0.91	0.89
Time	21.2	21.0	19.9	18.5	18.4	18.1	17.4	17.0	16.5	16.5	15.9
Dimension	30	29	28	27	26	25	24	23	22	21	20
Accuracy	0.87	0.87	0.83	0.87	0.87	0.80	0.80	0.84	0.88	0.83	0.86
Time	15.3	15.9	14.6	14.1	13.7	13.2	13.2	12.6	12	11.8	11.6
Dimension	19	18	17	16	15	14	13	12	11	10	9
Accuracy	0.86	0.87	0.83	0.82	0.88	0.90	0.88	0.85	0.86	0.90	0.89
Time	11.1	10.8	10.5	10.1	9.7	9.2	8.8	8.6	11.6	10.6	9.3
Dimension	8	7	6	5	4	3					
Accuracy	0.90	0.91	0.89	0.90	0.85	0.84					
Time	10.2	8.6	7.9	7.3	10.9	10					

Table 4: Complete test results using Bernoulli Bayes.

Dimension	41	40	39	38	37	36	35	34	33	32	31
Accuracy	0.96	0.96	0.95	0.95	0.92	0.95	0.94	0.88	0.92	0.91	0.84
Time	39.6	30.5	32.5	31.4	32.0	27.6	29.3	29.1	30.2	27.7	18.4
Dimension	30	29	28	27	26	25	24	23	22	21	20
Accuracy	0.90	0.90	0.89	0.86	0.89	0.85	0.84	0.87	0.91	0.88	0.87
Time	30	24	23	23	22	22	21.6	20.3	20.1	19.6	19
Dimension	19	18	17	16	15	14	13	12	11	10	9
Accuracy	0.84	0.86	0.86	0.88	0.89	0.92	0.91	0.90	0.92	0.94	0.95
Time	18.6	17.5	17.3	16	15.8	14.7	15	13.4	13	15	15.3
Dimension	8	7	6	5	4	3					
Accuracy	0.97	0.95	0.94	0.95	0.95	0.92					
Time	14.3	13.8	15.3	11.6	10.9	11					

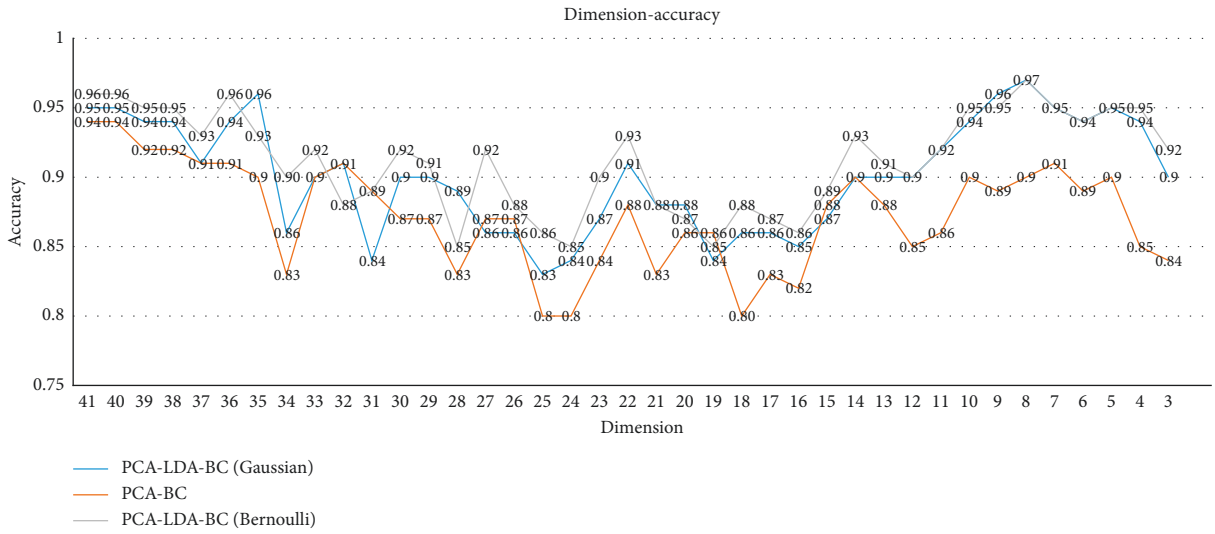

Figure 3: Relationship between dimension and accuracy.

(3) *F value*. The smaller error term of the two *F* values indicates that the test accuracy is higher

(4) *Feature selection*. The PCA algorithm is the top *n* feature with the highest contribution rate

(5) *Remove noise*. After the dimension reduction, a large amount of noise in the original sample features was removed and the experimental results obtained good results.

From a comprehensive point of view, the two methods can guarantee the accuracy of the original dimension after the dimension reduction by the feature, and at the same time, the detection time is greatly reduced.

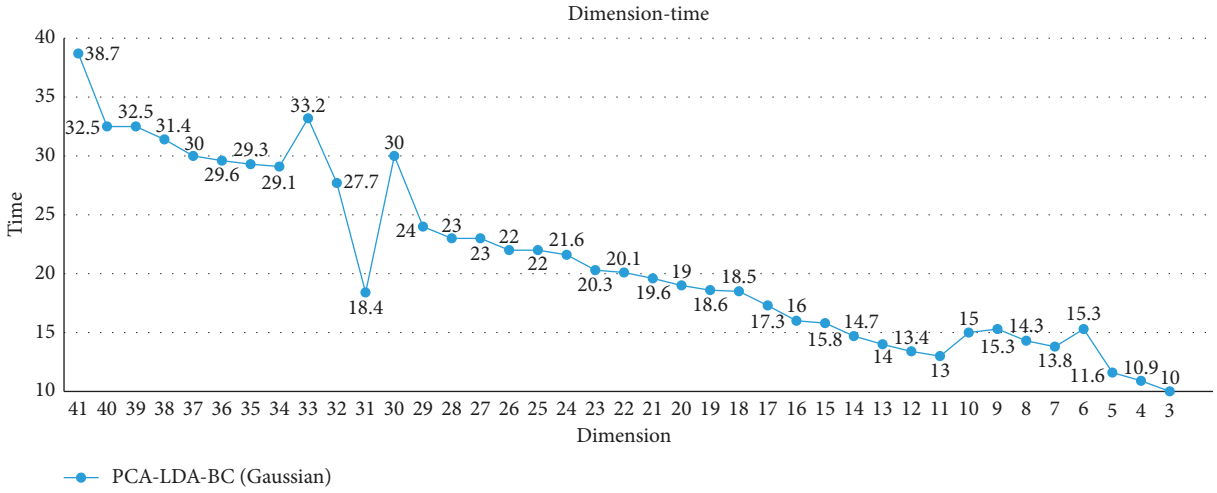

Figure 4: Relationship between dimension and time.

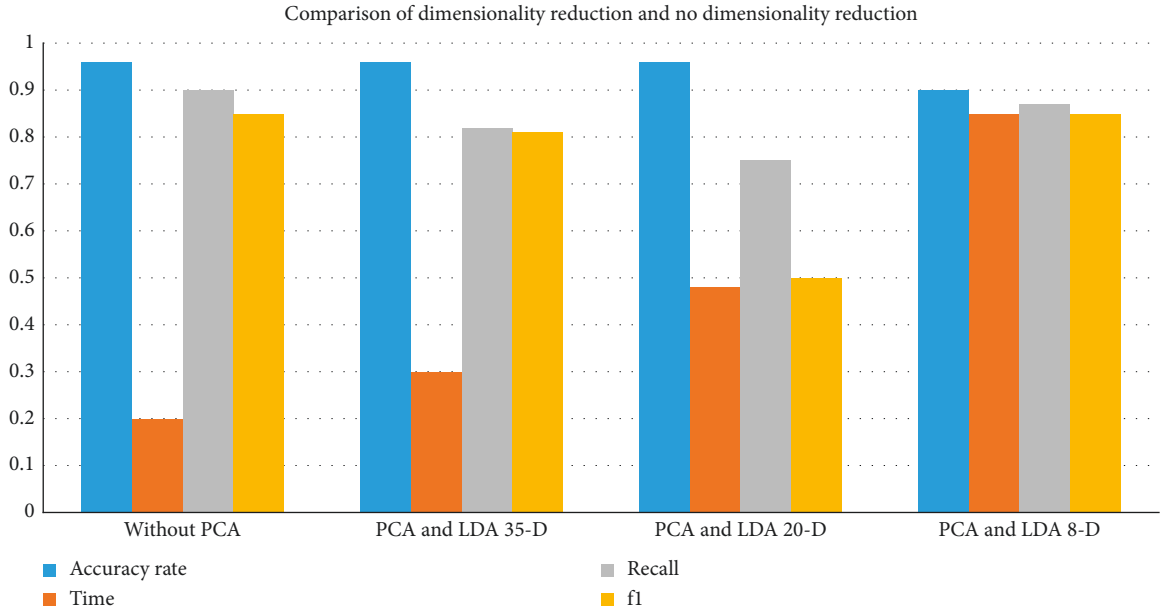

Figure 5: Comparison of dimensionality reduction and no dimensionality reduction.

5.5. Compare Bernoulli Naive Bayes and Gaussian Bayes. Table 5 shows the results of Bernoulli Naive Bayes and Gaussian Naive Bayes when the data set is reduced to 8 dimensions.

According to Table 5, Bernoulli Naive Bayes is superior to Gaussian Bayes in all aspects: higher accuracy, lower false detection rate, and better time efficiency.

5.6. Experiment on a Complete Data Set. We applied Bernoulli Naive Bayes to a complete data set for integrity testing to ensure the integrity of the experimental results. On the complete data set, there are some intrusions that are not available on the training set. By analysing the final experimental results, we can see that the method is very robust and can detect unknown intrusion activities. The experimental results are as follows. The result of Gaussian Naive Bayes in 8 dimensions is shown in Table 6:

The ROC diagram of Gaussian Naive Bayes in 8 dimensions is shown in Figure 6:

5.7. Compare with Other Methods. According to the same data set, referring to the results of other scholars [29, 30, 32, 33], as shown in Table 7, their machine learning classification is 2-Class.

According to the results of Table 7, we can see that several commonly used machine learning classification methods have the best performance accuracy of 91% under the same conditions, while the other methods are only about 80% and the accuracy of the PCA-LDA-BC algorithm is more than 92%.

Table 5: Results of Bernoulli Naive Bayes and Gaussian Naive Bayes in 8 dimensions.

Index	Bernoulli Naive Bayes	Gaussian Naive Bayes
Accuracy	0.987	0.951
Recall	0.903	0.904
False positive rate	0.00018	0.038
Precision	0.999	0.842
F-measure	0.949	0.872
Entropy	0.0007	0.1441
Time	12.5	15.2

Table 6: Results of Gaussian Naive Bayes in 8 dimensions.

Index	Gaussian Naive Bayes
Accuracy	0.924
Recall	0.976
False positive rate	0.008
Precision	0.637
F-measure	0.771
Entropy	0.287
Time	216.27

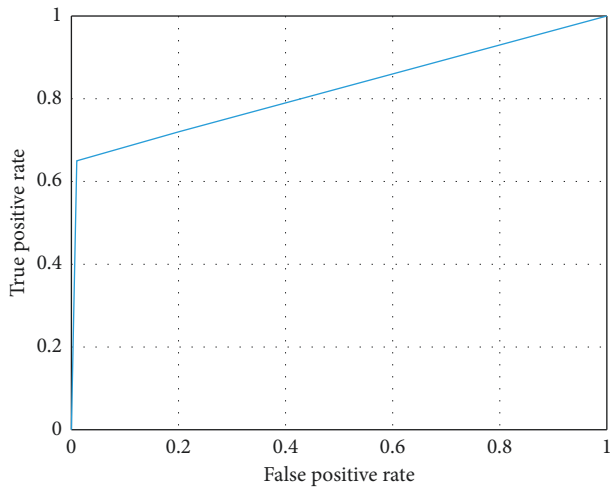

Figure 6: ROC diagram of Gaussian Naive Bayes in 8 dimensions.

Table 7: Comparison of other algorithms with PCA-LDA-BC.

Classification type	Accuracy (%)
Logistic regression	78.5
SVM	82.0
SVM (kernel function)	90.7
PCA-LDA-BC	96.0

The experimental results show that the intrusion detection system proposed by the PCA-LDA-BC algorithm has a certain improvement in the accuracy of intrusion detection compared with other commonly used methods. The fusion of the principal component analysis algorithm and linear discriminant algorithm can effectively remove the noise in the data and reduce the computational load of the classifier. The time performance is improved to some extent.

6. Conclusion

With the continuous development and expansion of network technology, cyber attacks are increasing day by day. Due to the high false detection rate, weak self-learning performance, and low robustness of traditional network security protection measures, how to quickly and accurately identify existing intrusions and increasing new attacks and new intrusions have become key issues to be solved.

Based on some experiences and methods of the predecessors, this paper makes a reasonable improvement on the current popular intrusion detection algorithm and proposes an improved Bayesian classification algorithm based on principal component analysis and linear discriminant analysis. The algorithm makes up for the insufficiency of the naive Bayesian hypothesis attribute independence, and at the same time makes up for the shortcomings of the current intrusion detection algorithm with poor real-time and accuracy when detecting a large number of behavioural features. Finally, the experiment proves that the proposed algorithm is compared with other classification algorithms, which effectively improves the accuracy and efficiency. However, this algorithm still needs to be improved. The next step continues to study how to optimize the algorithm to further improve the accuracy of classification, so as to adapt to the more complex multilateral network environment.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they do not have any commercial or associative interest that represents conflicts of interest in connection with the work submitted.

Acknowledgments

The authors would like to acknowledge the support provided by the National Key R&D Program of China (no. 2018YFC1604000) and the Natural Science Foundation of Hubei Province (no. 2017CFB663).

References

[1] M. Panda, A. Abraham, and M. R. Patra, "A hybrid intelligent approach for network intrusion detection," *Procedia Engineering*, vol. 30, pp. 1–9, 2012.

[2] L. Li, "The application of genetic algorithm to intrusion detection in MP2P network," in *Proceedings of the International Conference in Swarm Intelligence*, Springer, Brussels, Belgium, 2012.

[3] G. Wang, J. Hao, J. Ma, and L. Huang, "A new approach to intrusion detection using artificial neural networks and fuzzy clustering," *Expert Systems with Applications*, vol. 37, no. 9, pp. 6225–6232, 2010.

[4] X.-F. Wang and D. Ting, "Improved weighted Naive Bayes classification algorithm based on attribute selection," *Computer Systems & Applications*, vol. 24, no. 8, pp. 149–154, 2015.

[5] D. E. Denning, "An intrusion-detection model," *IEEE Transactions on Software Engineering*, vol. SE-13, no. 2, pp. 222–232, 1987.

[6] W. Park and S. Ahn, "Performance comparison and detection analysis in snort and suricata environment," *Wireless Personal Communications*, vol. 94, no. 2, pp. 241–252, 2017.

[7] E. A. Shams and A. Rizaner, "A novel support vector machine based intrusion detection system for mobile ad hoc networks," *Wireless Networks*, vol. 24, no. 5, pp. 1821–1829, 2018.

[8] O. Osanaiye, K.-K. R. Choo, and M. Dlodlo, "Distributed denial of service (DDoS) resilience in cloud: review and conceptual cloud DDoS mitigation framework," *Journal of Network and Computer Applications*, vol. 67, pp. 147–165, 2016.

[9] R. Taormina and S. Galelli, "A deep learning approach for the detection and localization of cyber-physical attacks on water distribution systems," *Journal of Water Resources Planning & Management*, vol. 144, no. 10, Article ID 04018065, 2018.

[10] U. Adhikari, T. H. Morris, and S. Pan, "Applying non-nested generalized exemplars classification for cyber-power event and intrusion detection," *IEEE Transactions on Smart Grid*, vol. 9, no. 5, pp. 3928–3941, 2018.

[11] Y.-y. Li, H.-w. Deng, S.-q. Wang, and J. Long, "High-speed intelligent internet intrusion defense system based on deep learning and NetFPGA," *Netinfo Security*, vol. 2, pp. 12–19, 2014.

[12] H. Wang, H.-y. Chen, and S.-f. Liu, "Intrusion detection system based on improved Naive Bayesian algorithm," *Computer Science*, vol. 4, no. 4, pp. 111–115, 2014.

[13] W. An and M. Liang, "A new intrusion detection method based on SVM with minimum within-class scatter," *Security and Communication Networks*, vol. 6, no. 9, pp. 1064–1074, 2013.

[14] E. Kabir, J. Hu, H. Wang, and G. Zhuo, "A novel statistical technique for intrusion detection systems," *Future Generation Computer Systems*, vol. 79, pp. 303–318, 2018.

[15] Z. Muda, W. Yassin, M. N. Sulaiman, and N. I. Udzir, "Intrusion detection based on K-Means clustering and Naïve Bayes classification," in *Proceedings of the 7th International Conference on Information Technology in Asia (CITA'11)*, pp. 1–6, IEEE, Kuching, Malaysia, July 2011.

[16] M. Ishida, H. Takakura, and Y. Okabe, "High-performance intrusion detection using OptiGrid clustering and grid-based labelling," in *Proceedings of the 11th IEEE/IPSJ International Symposium on Applications and the Internet*, pp. 11–19, Munich, Germany, July 2011.

[17] H. Om and A. Kundu, "A hybrid system for reducing the false alarm rate of anomaly intrusion detection system," in *Proceedings of the 2012 1st International Conference on Recent Advances in Information Technology, RAIT-2012*, pp. 131–136, Dhanbad, India, March 2012.

[18] M. F. Umer, M. Sher, and Y. Bi, "Flow-based intrusion detection: techniques and challenges," *Computers & Security*, vol. 70, pp. 238–254, 2017.

[19] W. Yao, J. Wang, and S. Zhang, "Intrusion detection model based on decision tree and Naive-Bayes classification," *Journal of Computer Applications*, vol. 7, pp. 2883–2885, 2015.

[20] P. Bermejo, L. De La Ossa, J. A. Gámez, and J. M. Puerta, "Fast wrapper feature subset selection in high-dimensional datasets by means of filter re-ranking," *Knowledge-Based Systems*, vol. 25, no. 1, pp. 35–44, 2012.

[21] L. Liu, B. Xu, X. Zhang, and X. Wu, "An intrusion detection method for internet of things based on suppressed fuzzy clustering," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, p. 113, 2018.

[22] L. I. Feng and G. C. Polytechnic, "Research on fuzzy clustering algorithm based on PSO in IDS," *Computer Technology and Development*, vol. 24, no. 12, pp. 138–141, 2014.

[23] E. A. Shams and A. Rizaner, "A novel support vector machine based intrusion detection system for mobile ad hoc networks," *Wireless Networks*, vol. 24, no. 5, pp. 1821–1829, 2017.

[24] R. S. Naoum and Z. N. Al-Sultani, "Learning vector quantization (LVQ) and k-nearest neighbor for intrusion classification," *World of Computer Science and Information Technology Journal (WCSIT)*, vol. 2, no. 3, pp. 105–109, 2012.

[25] Y. Jamshidi and H. Nezamabadi-pour, "A lattice based nearest neighbor classifier for anomaly intrusion detection," *Journal of Advances in Computer Research*, vol. 4, pp. 51–60, 2013.

[26] Z. Ma and A. Kaban, "K-nearest-neighbours with a novel similarity measure for intrusion detection," in *Proceedings of the 2013 13th UK Workshop on Computational Intelligence (UKCI)*, September 2013.

[27] Y. Li and Li Guo, "An active learning based TCM-KNN algorithm for supervised network intrusion detection," *Computers & Security*, vol. 26, no. 7-8, pp. 459–467, 2007.

[28] K. Ibrahimi and M. Ouaddane, "Management of intrusion detection systems based-KDD99: analysis with LDA and PCA," in *Proceedings of the International Conference on Wireless Networks and Mobile Communications (WINCOM)*, pp. 1–6, Rabat, Morocco, November 2017.

[29] W. Lee, S. J. Stolfo, and K. W. Mok, "Mining audit data to build intrusion detection models," in *Proceedings of the International Conference on Knowledge Discovery & Data Mining*, pp. 66–72, AAAI Press, Menlo Park, CA, USA, 1998.

[30] B. Kepenekci, F. Boray Tek, O. Cilingir, U. Sakarya, and G. B. Akar, "GAYE: a face recognition system," in *Proceedings of the Image Processing: Algorithms and Systems III*, vol. 5298, pp. 99–106, San Jose, CA, USA, May 2004.

[31] https://www.unb.ca/cic/datasets/ids-2017.html.

[32] KDD Cup 99 Data EB/OL, 2017, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

[33] W.-H. Chen, S.-H. Hsu, and H.-P. Shen, "Application of SVM and ANN for intrusion detection," *Computers & Operations Research*, vol. 32, no. 10, pp. 2617–2634, 2005.