# FAFS: A Fuzzy Association Feature Selection Method for Network Malicious Traffic Detection

**Yongxin Feng*, Yingyun Kang, Hao Zhang and Wenbo Zhang**
School of Information Science and Engineering, Shenyang Ligong University,
Shenyang, Liaoning 110159 – P.R.China
[e-mail: fengyongxin@263.net, kangyingyun22@163.com, haozhang_stu@163.com, zhangwenbo@sylu.edu.cn]
*Corresponding author: Yongxin Feng

*Received April 28, 2019; revised June 22, 2019; revised September 1, 2019; accepted October 5, 2019; published January 31, 2020*

## Abstract

Analyzing network traffic is the basis of dealing with network security issues. Most of the network security systems depend on the feature selection of network traffic data and the detection ability of malicious traffic in network can be improved by the correct method of feature selection. An FAFS method, which is short for Fuzzy Association Feature Selection method, is proposed in this paper for network malicious traffic detection. Association rules, which can reflect the relationship among different characteristic attributes of network traffic data, are mined by association analysis. The membership value of association rules are obtained by the calculation of fuzzy reasoning. The data features with the highest correlation intensity in network data sets are calculated by comparing the membership values in association rules. The dimension of data features are reduced and the detection ability of malicious traffic detection algorithm in network is improved by FAFS method. To verify the effect of malicious traffic feature selection by FAFS method, FAFS method is used to select data features of different dataset in this paper. Then, K-Nearest Neighbor algorithm, C4.5 Decision Tree algorithm and Naïve Bayes algorithm are used to test on the dataset above. Moreover, FAFS method is also compared with classical feature selection methods. The analysis of experimental results show that the precision and recall rate of malicious traffic detection in the network can be significantly improved by FAFS method, which provides a valuable reference for the establishment of network security system.

***Keywords:*** Network security; malicious traffic detection; association rules; fuzzy inference; feature selection

## 1. Introduction

Network traffic analysis technology is the cornerstone of network security system. At the same time, network traffic analysis of malicious and benign programs behave quite differently [1]. To achieve network traffic analysis, the current traffic is scanned and analyzed by IDS(intrusion detection system). IDS is one of the network traffic analysis technology and is used to identify network malicious traffic in normal traffic [2]. Sandnet is another network behavior analysis environment, which focuses on network traffic analysis [3]. Fantasm is also a system, which can support safe and productive malware experimentation [4]. With the development of Internet, network traffic analysis is facing the challenge of great traffic flux, low accuracy of detecting malicious traffic. Meanwhile the classification of network traffic is usually only considered in a few of the most relevant features [5]. Therefore, it is necessary to perform some form of feature selection methods to avoid over-fitting and improve the performance of classification. Moreover, it's also helpful to deal with limited computing resources for online or real-time IDS, through feature selection method. In addition, because of the relatively high feature space of network data sets, most machine learning methods, such as Bayes, Decision Tree and Neural Network, are easy to be over-fitting, when they are used in network anomaly traffic detection. Over-fitting can be effectively avoided, and the complexity of time and space can be reduced by feature selection method adopted in initial data sets. The accuracy of detection can be improved as well [6].

## 2. Related Work

In recent years, feature dimensionality reduction of network traffic data has been carried out extensively and studied deeply. Feature extraction and feature selection are the two main methods of feature dimension reduction [7]. Feature extraction, including Principal Component Analysis (PCA) and Kernel Principal Component Analysis (KPCA), is a combining transformation of the original features to form a new feature space. In the analysis of network traffic data, PCA and KPCA have been widely applicated and great progress has been made in the research on feature selection of network traffic data. Feature selection refers to the selection of some features from the original feature set to represent the entire data set [8]. According to different evaluation criteria, feature selection can be divided into filter model, embedding model and wrapper model. Although the principles of feature selection and feature extraction are different, the ultimate goal is to reduce the dimension of data sets and improve the effect of data analysis. A systematic method is required to select high-quality features. A divide-conquer and voting strategy [9] is proposed. In the divide-conquer and voting strategy, firstly, the original training set is segmented. Secondly, the feature subset is obtained by using the segmented subset. Thirdly, the final feature subset is obtained by voting. Aiming at solving the problem of serious data missing in massive data sets, mForest algorithm is proposed based on RF (random forest) algorithm [10]. The correlation among random features is further enhanced by the interpolation performance of RF algorithm, but the classification effect of multi-class data sets is not very obvious. The problems of multi-class imbalance and low recall rate of a few classes are also studied [11][12] and targeted feature selection methods are proposed. Although good results have been achieved in the experimental environment, the number of features selected is the same as the number of network applications, which is still facing considerable challenges in practical applications. Compared with feature extraction,

feature selection methods of these systems are more complex generally, but the malicious traffic detection model constructed by these systems also has much higher detection accuracy.

## 3. FAFS Method

In order to solve the problem of system complexity in high dimensional network traffic data and reduce the difficulty in feature selection, an FAFS method based on the present research is proposed in this paper. Through fuzzy inference calculation, important features in network traffic data can be automatically selected. Most of traditional feature selection methods depend on experts or intelligent recognition systems, but the information used in human and intelligent recognition systems is often uncertain. Human thinking, which is not as accurate as classical mathematics, is uncertain, complex and fuzzy. Therefore, fuzzy inference is used to represent and process the uncertain information of feature selection in network traffic. On the premise of fuzzy judgment, an approximate fuzzy judgment conclusion is derived by using fuzzy language rules. The data features with the most correlation strength of feature attributes in network traffic are obtained and the data containing in these features are used to detect network malicious traffic.

### 3.1 Fuzzification

FAFS method firstly needs to be fuzzified, which means to transform association rules into fuzzy association rules. The process of fuzzification is to establish the mapping relationship among the exact values of association rules and the fuzzy sets through the memberships function to form the fuzzy association rules. Obtaining fuzzy association rules is divided into four steps: association rules mining, feature word selection and construction of fuzzy sets, data standardization processing, and calculation of membership degree. A detailed description of each step is as follows.

#### 3.1.1 Association Rules Mining

Features with certain relevance can be calculated from a large number of network traffic data by association rule mining and appear together in different data categories. Those features are of great significance to the classification of network traffic data. Hence, relevance analysis of data features is indispensable in feature selection. At present, the main methods include Chi-square check, information gain, Pearson correlation coefficient and CfsSubsetEval[13]. The limitation of Chi-square verification is the "low-frequency defect", which exaggerates the role of low-frequency features. The low-frequency defect of Chi-square verification also results in excessive square value and eventually leads to errors in feature selection. The limitation of information gain is that it can only examine the contribution of features to the whole system, but not specifically to a certain category. Pearson correlation coefficient mainly measures the linear correlation degree of two variables, but there is no linear relationship among most data features in the network traffic data. Pearson correlation coefficient can not be applied in most network traffic data for feature selection[14]. CfsSubsetEval evaluates the attributes of subsets by considering the individual predictive ability of each feature along with the degree of redundancy between them. Subsets of features that are highly correlated with the class while having low intercorrelation are preferred. This method has a good effect in feature selection, but in the classification of network traffic data, the decisive features are usually only included in some of the most closely related features. Under the continuous iteration search, the relevance of network traffic data characteristics can be calculated through support and confidence. Although these rules contain a large amount of data redundancy, some of the rules

here contain some features that can be the best representation of data set. Therefore, association rule mining is used in this paper to measure the correlation among network traffic characteristics. [15].

The definitions of association rules mining are described as follows, before mining association rules from network traffic data.

**Definition** 1: Association Rule

Association rule reflects the relationship among items, which is denoted by $r$ and can be deemed to an implicative relation: $X \to Y$ . $X$ and $Y$ are called itemsets. $X \subseteq C, Y \subseteq C, X \bigcap Y = \varnothing$ . Itemset $C$ is an itemset that contains numbers of items $i_{\mathrm{n}}(\mathrm{n} = 1, 2, 3, ...)$ . Item is the specific content of network traffic dataset $D$. An itemset that contains k items is called a k-itemset $C_{\mathrm{k}}(1 \le k \le \mathrm{n})$.

**Definition** 2: Frequent Itemset

The itemset satisfying minimum support degree is called frequent itemset. According to the number of items included in frequent itemset, frequent itemset is also called frequent k-itemset, denoted by $L_{\mathrm{k}}(1 \le k \le \mathrm{n})$. The emergency frenquency of itemset is called support degree. Thus, support($C$) is used to express the support degree of itemsets $C$. support($C$)=$\dfrac{\mathrm{count}(C)}{\mathrm{m}}$, count($C$) is the number of occurrences of itemset $C$ in all transactions. A selected support degree threshold is called minimum support degree and is denoted by min_sup.

**Definition** 3: Strong Association Rule

Association rules which are satisfied minimum confidence degree, are called strong association rules. Confidence is used to indicate the frequency of itemset $Y$ in transactions involving itemset $X$. Thus, confident($X \to Y$) is used to express the confidence degree of $X \to Y$ . confident$(X \to Y) = \dfrac{\mathrm{count}(X \bigcup Y)}{\mathrm{count}(Y)}$ , count($Y$) is the number of occurrences of itemset in all transactions. A selected confidence threshold is called minimum confidence degree, denoted by min_conf.

The process of mining association rules is described as follows:

(1) Find_ $L_1$ (Finding frequent 1-itemset $L_1$)

Scanning the network traffic data set $D$, starting with itemset $C_1$, $L_1$ is found according to the given min_sup.

(2) Gen_$C_{\mathrm{k}}$ (Generating candidate k-itemsets) and Gen_ $L_{\mathrm{k}}$ (Generating frequent k-itemset)

According to the priori principle, if a set of items is frequent, then all its subsets must be frequent. Therefore, when generating candidate itemset $C_2$, $L_1$ can be directly used to generate it. After generating $C_2$, the candidate itemset $C_2$ is pruned according to the given min_sup, and the frequent itemset $L_2$ is generated. $L_2$ is the frequent itemset. By analogy, $C_{\mathrm{k}\text{-}1}$ is generated from $L_{\mathrm{k}\text{-}1}$, and $C_{\mathrm{k}}$ is pruned to produce $L_{\mathrm{k}}$ until frequent itemset of maximum items $L_{\mathrm{k}}$ is generated.

(3) Gen_StrongAssociationRules (Generating strong association rules)

According to the given min_conf, the frequent k-itemset $L_{\mathrm{k}}$ is pruned and strong association rules are generated.

### 3.1.2 Selection of Feature Word and Construction of Fuzzy Set

Selecting feature words from association rules is an important step in fuzzifying association rules. The definitions of correlation are as follows:

**Definition** 4: Fuzzy Set

Let *G* be a domain, and mapping $\mu_F(\mathrm{w}): G \to [0,1]$ is a fuzzy set *F* in *G*. Mapping $\mu_F(\mathrm{w})$, which is called membership function of *F*, which is used to express the membership degree of w to *F*. The input domain *U* and output domain *V* are included in the domain *G*.

**Definition** 5: Feature Word

Each feature attribute in association rule *r* is called feature word w. Then each association rule can be expressed as $r = (\mathrm{w}_1, \mathrm{w}_2, \mathrm{w}_3, ..., \mathrm{w}_l)\ (1 \le l \le \mathrm{k})$ and *l* is the number of feature attributes in each association rule.

The importance of each feature word in association rules and the association strength of feature words in association rules are divided into three levels: high (H), medium (M), and low (L). In the input domain *U*, the association rule $r = (\mathrm{w}_1, \mathrm{w}_2, \mathrm{w}_3, ..., \mathrm{w}_l)$ is used as input, and the fuzzy set $A=\{H_{in}, M_{in}, L_{in}\}$ is established to indicate the importance of the feature word. $\mu_A(w)$ is the membership function of *w* to the fuzzy set *A*. In the output domain, $y \in V$, which indicates the association strength of the features contained in the association rule after the calculation of the fuzzy inference, is the value of the output variable, and the fuzzy set $B = \{H_{out}, M_{out}, L_{out}\}$ is established to indicate the strong association of the features contained in the association rule. $\mu_B(y)$ is the membership function of *y* subordinate to fuzzy set *B*.

### 3.1.3 Data Standardization Processing

Before the membership degree of each feature word is calculated, the input data should be standardized.

(1) Constructing a frequency matrix of feature words

The frequency matrix of feature words is defined as Eq.(1).

$$W = [\mathrm{w}_{ij}]_{\mathrm{n}\times l}, i = 1,2,3,...,n \quad j = 1,2,3,...,l \tag{1}$$

$\mathrm{w}_{ij}$ is the number of occurrences of the feature word *j* in the set of rule *i*. *n* is the number of rules, *l* is the number of feature words in the rule *i*.

(2) Normalization

In order to balance the distribution of each feature word in the set, each feature word in the frequency matrix is normalized as Eq. (2).

$$b_{ij} = \frac{\mathrm{w}_{ij}}{(\sum_{\mathrm{k}=1}^{\mathrm{n}} \mathrm{w}_{ik}^2)^{1/2}}, i = 1,2,3,...,n \quad j = 1,2,3,...,l \tag{2}$$

### 3.1.4 Membership Degree Calculation

In this paper, the Gauss function is chosen as the membership function. The expression of the Gauss membership function is defined as Eq. (3).

$$f(x,\sigma,c) = e^{-\frac{(x-c)^2}{2\sigma^2}} \tag{3}$$

The central position of the function is determined by *c* and the shape of the function is determined by $\sigma$.

Commonly used methods to construct membership function include: fuzzy statistics method, reference function method (Gauss function, Triangular function, Trapezoidal function) and so on [16]. Traffic in the network usually presents a large number of traffic bytes in a short time.

These traffic byte data, such as the number of bytes in the network stream, the duration of the network flow, the average number of bytes per packet and other network traffic characteristics, tend to concentrate near a certain value, showing a symmetrical distribution of a peak value. Therefore, the Gauss distribution with this distribution law is chosen as the membership function. Although the Triangular function has similar distribution law and Trapezoidal function, the Gauss distribution is smoother and stabler, and its resolution can be controlled by operating the smoothness of the Gauss distribution.

The normalized value $b_{ij}$ is brought into the corresponding membership function to calculate the MS_Degree (membership degree) of each feature word belonging to each fuzzy set.

The fuzzy association rules $r = (\mu(\mathrm{w}_1), \mu(\mathrm{w}_2), \mu(\mathrm{w}_3), ..., \mu(\mathrm{w}_l))$ is formed by the membership degree of each association rules $r = (\mathrm{w}_1, \mathrm{w}_2, \mathrm{w}_3, ..., \mathrm{w}_l)$.

### 3.2 Establishment of Fuzzy Rule Base

Fuzzy rules are generated mainly by manually compiling some rules to meet certain needs. The following three fuzzy rules, which are the most important rules in the network traffic data set, are manually compiled, to find some features with high correlation strength.

$$\mathrm{FR}_1: \text{IF } \mathrm{w}_1 \text{ IS } H, \mathrm{w}_2 \text{ IS } H, \mathrm{w}_3 \text{ IS } H, ..., \mathrm{w}_l \text{ IS } H \text{ THEN } r \text{ IS } H \quad (4)$$

$$\mathrm{FR}_2: \text{IF } \mathrm{w}_1 \text{ IS } M, \mathrm{w}_2 \text{ IS } M, \mathrm{w}_3 \text{ IS } M, ..., \mathrm{w}_l \text{ IS } M \text{ THEN } r \text{ IS } M \quad (5)$$

$$\mathrm{FR}_3: \text{IF } \mathrm{w}_1 \text{ IS } L, \mathrm{w}_2 \text{ IS } L, \mathrm{w}_3 \text{ IS } L, ..., \mathrm{w}_l \text{ IS } L \text{ THEN } r \text{ IS } L \quad (6)$$

In order to reduce the computational complexity, only the three cases mentioned above are considered in this paper. When the importance of features in association rules is high, the importance of the association rules is also high. When the importance of features in association rules is medium, the importance of the association rules is also medium. When the importance of features in association rules is low, the importance of the association rules is also low. The importance of each association rule is calculated according to these three fuzzy rules.

### 3.3 Fuzzy Inference Calculation

The calculation results of fuzzy inference mainly depend on the fuzzy implication relation $R(U,V)$ and the synthetic operation rule between the fuzzy relation and the fuzzy set. The commonly used methods of fuzzy inference are Mamdani, Larsen and Sugeno inference [17]. The purpose of this paper is to select features that can improve malicious traffic detection from the feature attributes of network traffic data. Mamdani-type fuzzy inference method is adopted in this paper. The fuzzy implication relation of Mamdani-type fuzzy inference method is a compound proposition composed of $U(\mathrm{w})$ and $V(y)$, denoted by "$U(\mathrm{w}) \to V(y)$".

The language information carrying capacity of Mamdani fuzzy inference system is prominent and it is suitable for expressing expert experience. It is very suitable for selecting network traffic feature. Larsen's inference method is very similar to Mamdani's inference process. The difference is that the product operation is used instead of the small operation in the calculation of excitation intensity and inference synthesis, which cannot calculate the importance of each feature in feature selection accurately. The membership function of Sugeno inference output can only be linear or constant, which is not universal. Therefore, Mamdani-type fuzzy inference method is adopted.

In Eq.(4), Eq.(5) and Eq.(6), for the preceding part of IF-THEN rule, the form of atomic

fuzzy proposition is "T: w IS *A*". Its true value takes the membership degree $\mu_A(\mathrm{w})$ of the variable w to the fuzzy set *A*, is defined as Eq. (7).

$$\mathrm{P(T)} = \mu_A(\mathrm{w}) \tag{7}$$

According to the three fuzzy rules established in Section 3.2, three sub-fuzzy implication relations are derived from Mamdani's fuzzy inference system as Eq. (8), Eq. (9) and Eq. (10).

$$\begin{aligned} R_{FR_1} &= \mu_{H_{in}}(\mathrm{w}_1) \wedge \mu_{H_{in}}(\mathrm{w}_2) \wedge \mu_{H_{in}}(\mathrm{w}_3) \wedge \ldots \wedge \mu_{H_{in}}(\mathrm{w}_l) \wedge \mu_{H_{out}}(y) \\ &= \min\left\{\mu_{H_{in}}(\mathrm{w}_1), \mu_{H_{in}}(\mathrm{w}_2), \mu_{H_{in}}(\mathrm{w}_3), \ldots, \mu_{H_{in}}(\mathrm{w}_l), \mu_{H_{out}}(y)\right\} \end{aligned} \tag{8}$$

$$\begin{aligned} R_{FR_2} &= \mu_{M_{in}}(\mathrm{w}_1) \wedge \mu_{M_{in}}(\mathrm{w}_2) \wedge \mu_{M_{in}}(\mathrm{w}_3) \wedge \ldots \wedge \mu_{M_{in}}(\mathrm{w}_l) \wedge \mu_{M_{out}}(y) \\ &= \min\left\{\mu_{M_{in}}(\mathrm{w}_1), \mu_{M_{in}}(\mathrm{w}_2), \mu_{M_{in}}(\mathrm{w}_3), \ldots, \mu_{M_{in}}(\mathrm{w}_l), \mu_{M_{out}}(y)\right\} \end{aligned} \tag{9}$$

$$\begin{aligned} R_{FR_3} &= \mu_{L_{in}}(\mathrm{w}_1) \wedge \mu_{L_{in}}(\mathrm{w}_2) \wedge \mu_{L_{in}}(\mathrm{w}_3) \wedge \ldots \wedge \mu_{L_{in}}(\mathrm{w}_l) \wedge \mu_{L_{out}}(y) \\ &= \min\left\{\mu_{L_{in}}(\mathrm{w}_1), \mu_{L_{in}}(\mathrm{w}_2), \mu_{L_{in}}(\mathrm{w}_3), \ldots, \mu_{L_{in}}(\mathrm{w}_l), \mu_{L_{out}}(y)\right\} \end{aligned} \tag{10}$$

As shown in Eq.(11), the total fuzzy implication relation of the system is composed of three sub-fuzzy implication relations of $R_{FR_1}$, $R_{FR_2}$ and $R_{FR_3}$, and the relationship among them is 'or'.

$$R = R_{FR_1} \bigcup R_{FR_2} \bigcup R_{FR_3} \tag{11}$$

The input feature words are expressed by vector $\vec{W} = (\mu(\mathrm{w}_1), \mu(\mathrm{w}_2), \mu(\mathrm{w}_3), \cdots, \mu(\mathrm{w}_l))$, Through fuzzy inference system, the output fuzzy quantity $\vec{Y}$ can be obtained by Eq.(12).

$$\begin{aligned} \vec{Y} &= \vec{W}^{\mathrm{T}} \circ (R_{FR_1} \bigcup R_{FR_2} \bigcup R_{FR_3}) \\ &= \vec{W}^{\mathrm{T}} \circ R_{FR_1} \bigcup \vec{W}^{\mathrm{T}} \circ R_{FR_2} \bigcup \vec{W}^{\mathrm{T}} \circ R_{FR_3} \\ &= (\mu(\mathrm{w}_1), \mu(\mathrm{w}_2), \mu(\mathrm{w}_3), \ldots, \mu(\mathrm{w}_l))^{\mathrm{T}} \circ R_{FR_1} \bigcup \\ &\quad (\mu(\mathrm{w}_1), \mu(\mathrm{w}_2), \mu(\mathrm{w}_3), \ldots, \mu(\mathrm{w}_l))^{\mathrm{T}} \circ R_{FR_2} \bigcup \\ &\quad (\mu(\mathrm{w}_1), \mu(\mathrm{w}_2), \mu(\mathrm{w}_3), \ldots, \mu(\mathrm{w}_l))^{\mathrm{T}} \circ R_{FR_3} \\ &= \min\left\{\mu_{H_{in}}(\mathrm{w}_1), \mu_{H_{in}}(\mathrm{w}_2), \mu_{H_{in}}(\mathrm{w}_3), \ldots, \mu_{H_{in}}(\mathrm{w}_l), \mu_{H_{out}}(y)\right\} \bigcup \\ &\quad \min\left\{\mu_{M_{in}}(\mathrm{w}_1), \mu_{M_{in}}(\mathrm{w}_2), \mu_{M_{in}}(\mathrm{w}_3), \ldots, \mu_{M_{in}}(\mathrm{w}_l), \mu_{M_{out}}(y)\right\} \bigcup \\ &\quad \min\left\{\mu_{L_{in}}(\mathrm{w}_1), \mu_{L_{in}}(\mathrm{w}_2), \mu_{L_{in}}(\mathrm{w}_3), \ldots, \mu_{L_{in}}(\mathrm{w}_l), \mu_{L_{out}}(y)\right\} \end{aligned} \tag{12}$$

The symbol '$\circ$' denotes the composition operator in the fuzzy relation. The operation method is the same as the ordinary matrix multiplication, but the 'multiplication' is changed to 'minimum'. The symbol '$\cup$' indicates that the final output of the system is the result of the interaction of three fuzzy implication relations. Therefore, the output $\vec{Y}$ at this time is still a fuzzy subset and must be defuzzified. Y stands for the Fuzzy quantity in fuzzy set $\vec{Y}$, $\mu_{\vec{Y}}(\mathrm{Y})$ stands for the membership values of Y subordinated to $\vec{Y}$.

### 3.4 Defuzzification

The task of defuzzification is to find a clear value to represent the fuzzy subset. In this paper, the area center of gravity method is used as the defuzzification method. Area center of gravity method is defined as Eq. (13).

$$y = \frac{\int_{\vec{Y}} \mathrm{Y}\mu_{\vec{Y}}(\mathrm{Y})d\mathrm{Y}}{\int_{\vec{Y}} \mu_{\vec{Y}}(\mathrm{Y})d\mathrm{Y}} \tag{13}$$

Y stands for the Fuzzy quantity in fuzzy set $\vec{Y}$, $\mu_{\vec{Y}}(\mathrm{Y})$ stands for the memberships values of Y subordinated to $\vec{Y}$.

Area center of gravity method, maximum membership degree method are commonly used methods of defuzzification [18]. The element with the largest membership degree is chosen by the maximum membership degree method in the inference result fuzzy set as the output value. And the shape of the output membership function is not considered, but only the output value at the maximum membership degree. Much information will be lost inevitably, and some important features may be lost in the process of feature selection. Therefore, in order to obtain the accurate control quantity, as the defuzzification method used in the area center of gravity method. The area center of gravity method has smoother output inference control than the maximum membership method. The slightly changes of input value can lead to change in output value. It not only increases the applicability of feature selection algorithm, but also reduces the possibility of missed judgments.

In Eq. (13) y is the determinate value after defuzzification, Y is used to denote the fuzzy quantity in the fuzzy set $\vec{Y}$, and $\mu_{\vec{Y}}(\mathrm{Y})$ is used to denote the membership value of Y to $\vec{Y}$. The association rules with the largest value of y are screened out and placed in the set of association rules (*MAXVALUE_r*) to determine the features. The features contained in *MAXVALUE_r* are used as the features for malicious traffic detection.

### 3.5 Process

The theoretical knowledge and implementation details of the FAFS method are comprehensively introduced above, and the pseudo-code of the method is given as follows:

Methods: FAFS method

Input：Network Traffic Data Set *D*, min_sup, min_conf
Output：Feature Set of Network Traffic Data after Feature Selection (*Features*)
1：$L_1$ = Find_ $L_1$ (*D*);
2：for(k = 2; $L_{k-1} \neq \varnothing$;k++)
3：  $C_k$ = Gen_$C_k$ ($L_{k-1}$);
4：  $L_k$ = Gen_$L_k$ ($C_k$, min_sup);
5：end for
6：return $L_k$;
7：*r* = Gen_StrongAssociationRules ($L_k$, min_conf);
8：for each w in *r*
9：  M_Degree = $\mu_A(\mathrm{w})$;
10：end for

```
11： Y⃗ = Gen_Y⃗ (r, W⃗ );
12： y = Gen_y(Y⃗ , μ(Y⃗ ) );
13：List_y[ ] = sort(y);
14：for (i=0;i<n; i ++)
15：   MAXVALUE_r[i] = List_y[i];
16：end for
17：Features[ ] = Gen _features(MAXVALUE_r);
18：end
```

The network traffic data set $D$, min_sup and min_conf are input. The frequent k-itemset $L_k$ is obtained by min_sup and the strong association rule $r$ is obtained by min_conf. After all strong association rules are fuzzified, the fuzzy value of each strong association rule is obtained by fuzzy inference calculation. Then, the fuzzy value of each strong association rule is defuzzified. Representing the $y$ value of the feature association strength contained in all strong association rules, the value $y$ is sorted from large to small, and the strong association rules with the largest value $y$ are screened out. The feature attributes contained in these association rules are the features obtained by using the method of fuzzy association feature selection. At this point, the whole process of FAFS method is completed.

## 4. Experimental Results and Analysis

### 4.1 Experimental Data and Evaluation Methods

In order to verify the effect and superiority of FAFS method, FAFS method is used to select data features of KDD CUP99 data set [19], NSL-KDD data set [20] and Modbus_traffic network traffic data set [21], which contain dozens of common network attacks. The NSL-KDD data set, which is more suitable for effective and accurate evaluation among different machine learning algorithms, eliminates the redundancy of KDD CUP99 data set, and the proportion of normal data and abnormal data are chosen properly in the data set. The volume of test and training data is more reasonable. There are 41 feature attributes and a data category label in NSL-KDD data set as that in KDD CUP99 data set. The Modbus_traffic data set is the data captured by Darryfei simulating network attacks. Data volume from large to small is KDD CUP99, NSL-KDD, Modbus_traffic.

Then, KNN algorithm (K-Nearest Neighbor algorithm), C4.5 Decision Tree algorithm and Naïve Bayes algorithm are used to test on the dataset above. The parameter $k$ represents the data points with the smallest $k$ distance for judgment in KNN algorithm. The value of parameter $k$ is set 11 in this paper. In C4.5 Decision Tree algorithm, the best feature of data set is selected by calculating the information gain rate of each features. The threshold of information gain rate is set as 0.1.

At the same time, the precision rate refers to examples correctly labeled as positive and recall rate refers to negative examples correctly labeled as negative are taken as evaluation indexes.

Moreover, FAFS method is also compared with classical feature selection methods, such as CFS, GainR, InfoG, Sym, which is short for CfsSubsetEval, GainRatioAttributeEval, InfoGainAttributeEval, SymmetricalUncertAttributeEval respectively. CfsSubsetEval evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy among them. Subsets of features that are highly correlated with the class while having low intercorrelation are preferred.

GainRatioAttributeEval evaluates the worth of an attribute by measuring the gain ratio with respect to the class. InfoGainAttributeEval evaluates the worth of an attribute by measuring the information gain with respect to the class. SymmetricalUncertAttributeEval evaluates the worth of an attribute by measuring the symmetrical uncertainty with respect to the class.

### 4.2 Experimental Results and Analysis

The precision and recall rate of different original data sets and different data sets, the feature of which are selected by CFS, GainR, InfoG, Sym feature selection method and FAFS method are calculated.

(1) KDD CUP99 data set

Let $f_k$ ($k$=1,2,3, … ,41) be 41 feature attributes in KDD CUP99 original data set and $f_1$ =duration, $f_2$=protocol_type, $f_3$=service, …, $f_{41}$=dst_host_srv_rerror_rate. The results of KDD CUP99 data set using FAFS method are partly shown in **Table 1**.

**Table 1.** Results of KDD CUP99 data set using FAFS method

| Association Rules | Value of $y$ |
|---|---|
| (2,3,32,4,23,1,29) | 0.5607 |
| (2,3,32,23,1,29,4) | 0.5607 |
| (2,32,4,23,1,29,3) | 0.5607 |
| (2,3,4,23,1,29,32) | 0.5607 |
| … … | … … |
| (2,4,23,1,29,3,32) | 0.5607 |
| (2,23,1,29,3,4,32) | 0.5607 |
| (2,5,32,3,4,1,29) | 0.5607 |
| (2,32,3,4,23,1,29) | 0.5607 |

According to **Table 1**, the association rules with the maximum y value of feature attribute association strength are obtained. The maximum value of y is 0.5607, and the number of association rules with the value 0.5607 is 17. The features of these association rules are as follows: $f_1$=duration, $f_2$=protocol_type, $f_3$=service, $f_4$=flag, $f_5$=src_bytes, $f_{23}$=count, $f_{29}$=same_srv_rate, $f_{32}$=dst_host_count. The eight features are features of KDD CUP99 data set selected by FAFS method.

**Table 2.** Detection results of KDD CUP99 data set

(a) Precision rate of KDD CUP99 data set

| Method | Precision Rate/% | | | | | |
|---|---|---|---|---|---|---|
| | Original | CFS | GainR | InfoG | Sym | FAFS |
| K-NN | 80.566 | 89.015 | 87.462 | 87.554 | 87.549 | 88.473 |
| C4.5 | 80.22 | 9.138 | 81.209 | 67.148 | 1.843 | 82.819 |
| Naïve Bayes | 73.023 | 80.255 | 77.158 | 78.444 | 83.039 | 81.183 |

(b) Recall rate of KDD CUP99 data set

| Method | Recall Rate/% | | | | | |
|---|---|---|---|---|---|---|
| | Original | CFS | GainR | InfoG | Sym | FAFS |
| K-NN | 90.256 | 99.167 | 97.437 | 97.539 | 97.533 | 98.563 |
| C4.5 | 87.876 | 10.011 | 88.973 | 73.567 | 2.02 | 88.113 |
| Naïve Bayes | 79.992 | 87.927 | 84.534 | 85.943 | 90.975 | 88.931 |

Then, K-Nearest Neighbor algorithm, C4.5 Decision Tree algorithm and Naïve Bayes algorithm are used to test on the data containing the eight features from KDD CUP99 data set. Moreover, FAFS method is also compared with classical feature selection methods, such as CFS, GainR, InfoG, Sym. The network malicious traffic detection results of original KDD CUP99 data set, and KDD CUP99 data set, the features of which are selected by CFS, GainR, InfoG , Sym feature selection method and FAFS method are shown in **Table 2**.

(2) NSL-KDD data set

Let $v_k$ ($k$=1,2,3, …, 41) be 41 feature attributes in NSL-KDD original data set and $v_1$=duration, $v_2$=protocol_type, $v_3$=service, …, $v_{41}$=dst_host_srv_rerror_rate. The results of NSL-KDD data set using FAFS method are partly shown in **Table 3**.

**Table 3.** Results of NSL-KDD data set using FAFS method

| Association Rules | Value of $y$ |
|---|---|
| (29,30,31,33,34,35,1) | 0.5869 |
| (1,30,31,33,34,35,29) | 0.5869 |
| (30,31,33,34,35,1,29) | 0.5869 |
| (29,30,31,33,34,35,1) | 0.5869 |
| … | … |
| … | … |
| (29,30,31,33,34,35,1) | 0.5869 |
| (1,30,31,33,34,35,29) | 0.5869 |
| (30,31,33,34,35,1,29) | 0.5869 |
| (29,30,31,33,34,35,1) | 0.5869 |

**Table 3** shows that the association rules with the maximum y value of feature attribute association strength are obtained. The maximum value of y is 0.5869, and the number of association rules with the value 0.5869 is 198. The features of these association rules are as follows: $v_1$=duration, $v_{29}$=same_srv_rate, $v_{30}$=diff_srv_rate, $v_{31}$=srv_diff_host_rate, $v_{33}$=dst_host_srv_count, $v_{34}$=dst_host_same_srv_rate, $v_{35}$=dst_host_diff_srv_rate. The seven features are features of NSL-KDD data set selected by FAFS method.

**Table 4.** Detection results of NSL-KDD data set

(a) Precision rate of NSL-KDD data set

| Method | Precision Rate/% | | | | | |
|---|---|---|---|---|---|---|
| | Original | CFS | GainR | InfoG | Sym | FAFS |
| K-NN | 71.62 | 83.392 | 78.383 | 83.273 | 82.535 | 75.843 |
| C4.5 | 71.709 | 66.259 | 67.276 | 66.342 | 67.279 | 74.5 |
| Naïve Bayes | 73.428 | 73.32 | 59.741 | 73.331 | 70.783 | 76.323 |

(b) Recall rate of NSL-KDD data set

| Method | Recall Rate/% | | | | | |
|---|---|---|---|---|---|---|
| | Original | CFS | GainR | InfoG | Sym | FAFS |
| K-NN | 81.304 | 94.669 | 88.983 | 94.534 | 93.6962 | 86.099 |
| C4.5 | 77.663 | 71.609 | 72.708 | 71.698 | 72.712 | 80.515 |
| Naïve Bayes | 79.357 | 79.24 | 64.565 | 79.252 | 76.5 | 82.486 |

Then, K-Nearest Neighbor algorithm, C4.5 Decision Tree algorithm and Naïve Bayes algorithm are used to test on the data containing the seven features from NSL-KDD data set. Moreover, FAFS method is also compared with classical feature selection methods, such as CFS, GainR, InfoG, Sym. The network malicious traffic detection results of original NSL-KDD data set, and NSL-KDD data set, the features of which are selected by CFS, GainR, InfoG, Sym feature selection method and FAFS method are shown in **Table 4**.

(3) Modbus_traffic data set

Let $z_k$ ($k$=1,2,3, …, 25) be 25 feature attributes in Modbus_traffic original data set and $z_1$=right_ar, $z_2$=left_ar, $z_3$=sip, …, $z_{25}$=content. The results of Modbus_traffic data set using FAFS method are partly shown in **Table 5**.

As shown in **Table 5**, the association rules with the maximum y value of feature attribute association strength are obtained. The maximum value of y is 0.5737, and the number of association rules with the value 0.5737 is 1690. The features of these association rules are as follows: $z_1$=right_ar, $z_4$=sport, $z_6$=doprt, $z_9$=ptc_lable, $z_{11}$=uni_lable, $z_{12}$=fun_code, $z_{14}$=direction, $z_{16}$=source_port, $z_{18}$=destination_port, $z_{22}$=length, $z_{23}$=unit_lable, $z_{25}$=content. The twelve features are features of Modbus_traffic data set selected by FAFS method.

**Table 5.** Results of Modbus_traffic Data set using FAFS method

| Association Rules | Value of y |
|---|---|
| (4,6,9,11,12,18,25,16,22,23,14) | 0.5737 |
| (4,6,9,11,12,18,25,22,23,14,16) | 0.5737 |
| (4,6,9,11,12,18,25,16,23,14,22) | 0.5737 |
| (1,4,6,9,11,12,16,22,23,14,25) | 0.5737 |
| … … | … … |
| (1,4,6,9,11,12,22,23,14,25,16) | 0.5737 |
| (1,4,6,9,11,12,23,14,25,16,22) | 0.5737 |
| (1,4,6,9,11,12,22,14,25) | 0.5737 |
| (4,6,9,11,12,18,25,22,14) | 0.5737 |

**Table 6.** Detection results of Modbus_traffic data set

(a) Precision rate of Modbus_traffic data set

| Method | Precision Rate/% | | | | | |
|---|---|---|---|---|---|---|
| | Original | CFS | GainR | InfoG | Sym | FAFS |
| K-NN | 72.355 | 88.516 | 88.463 | 88.463 | 88.463 | 82.721 |
| C4.5 | 70.813 | 70.814 | 2.84 | 70.814 | 70.814 | 80.497 |
| Naïve Bayes | 74.45 | 92.74 | 92.81 | 92.775 | 92.775 | 78.08 |

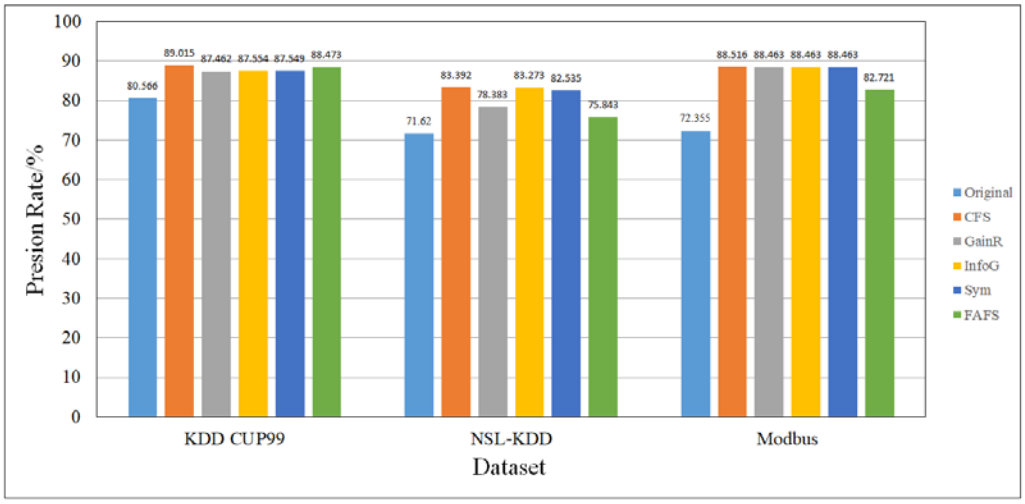
(b) Recall rate of Modbus_traffic data set

| Method | Recall Rate /% | | | | | |
|---|---|---|---|---|---|---|
| | Original | CFS | GainR | InfoG | Sym | FAFS |
| K-NN | 81.741 | 99.82 | 99.76 | 99.76 | 99.76 | 93.453 |
| C4.5 | 81.881 | 81.882 | 3.266 | 81.882 | 81.882 | 93.079 |
| Naïve Bayes | 80.037 | 99.7 | 99.775 | 99.737 | 99.717 | 83.939 |

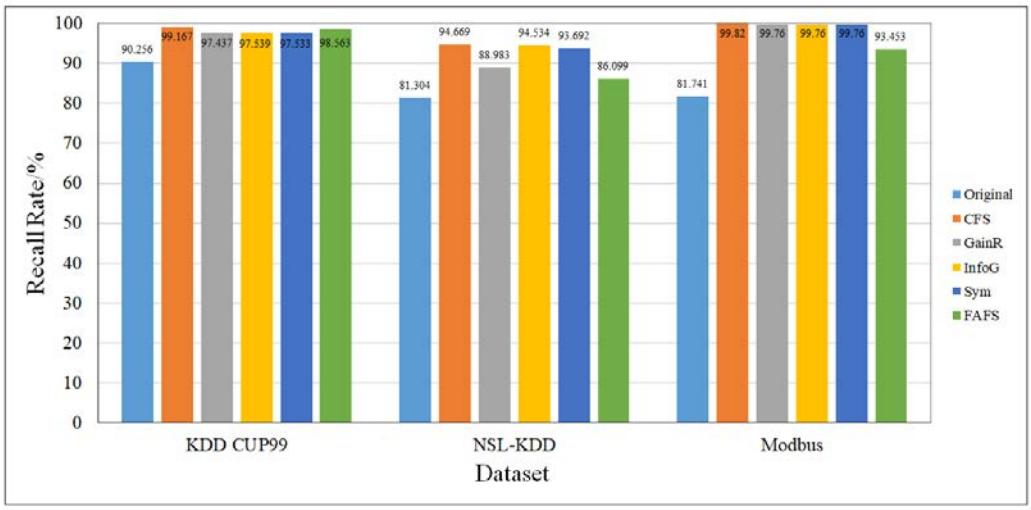
Then, K-Nearest Neighbor algorithm, C4.5 Decision Tree algorithm and Naïve Bayes algorithm are used to test on the data containing the twelve features from Modbus_traffic data

set. Moreover, FAFS method is also compared with classical feature selection methods, such as CFS, GainR, InfoG, Sym. The network malicious traffic detection results of original Modbus_traffic data set, and Modbus_traffic data set, the features of which are selected by CFS, GainR, InfoG, Sym feature selection method and FAFS method are shown in **Table 6**.

The detection effect of K-Nearest Neighbor algorithm on different original data sets, and data sets, the feature of which are selected by CFS, GainR, InfoG, Sym feature selection method and FAFS method are shown in **Fig. 1**.




(a) Precision rate of K-NN




(b) Recall rate of K-NN

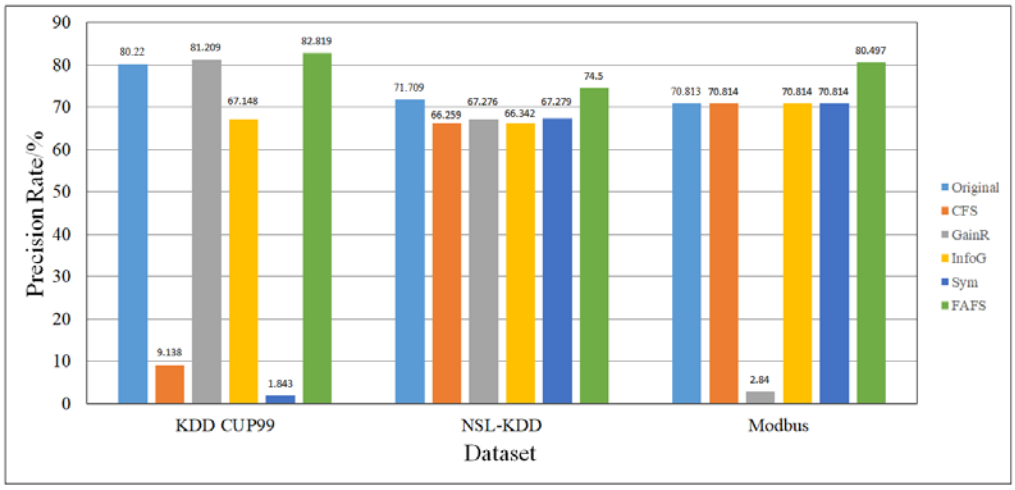
**Fig. 1.** Detection effect of K-nearest neighbor algorithm

**Table 7.** Execution time required by K-NN algorithm

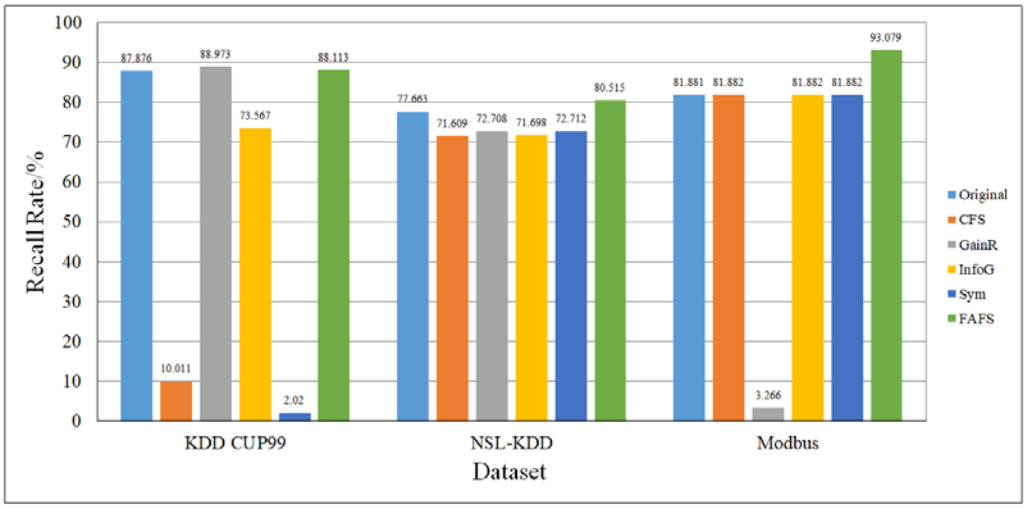
| Data set | Time（s） | | | | | |
|---|---|---|---|---|---|---|
| | Original | CFS | GainR | InfoG | Sym | FAFS |
| KDD CUP99 | 10644.6 | 9129.24 | 2016.63 | 2377.86 | 2248.107 | 1891.07 |
| NSL-KDD | 1489.32 | 504.447 | 360.558 | 400.018 | 361.704 | 366.11 |
| Modbus_traffic | 0.33 | 0.2404 | 0.2689 | 0.2877 | 0.305 | 0.26 |

The experiments are implemented at a server with Intel(R) Core(TM) i5-7500 3.4GHz. The execution time required by K-NN algorithm detecting malicious traffic in different data sets using different feature dimensionality reduction methods are shown in **Table 7**.

The detection effect of C4.5 Decision Tree algorithm on different original data sets, and data sets, the feature of which are selected by CFS, GainR, InfoG, Sym feature selection method and FAFS method are shown in **Fig. 2**.




(a) Precision rate of C4.5 Decision Tree algorithm




(b) Recall rate of C4.5 Decision Tree algorithm

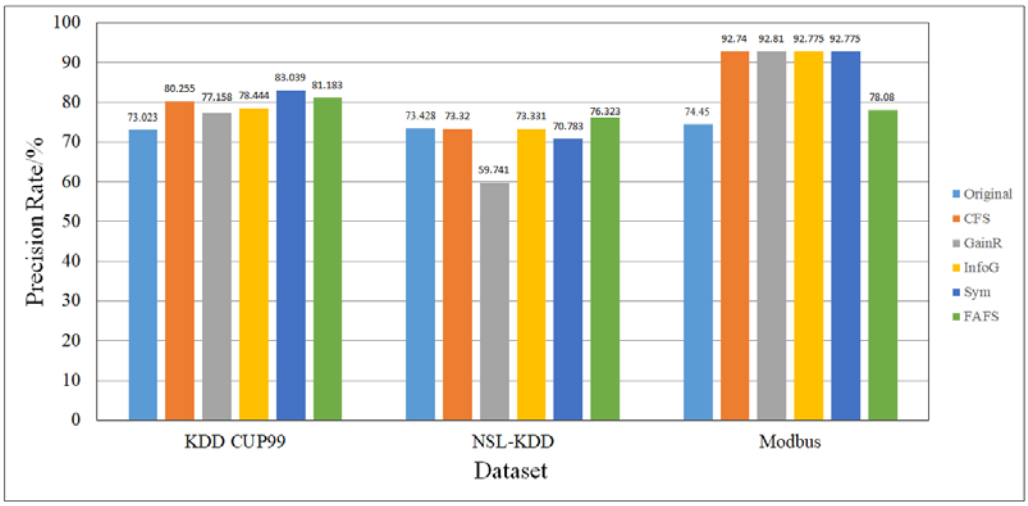
**Fig. 2.** Detection effect of C4.5 Decision Tree algorithm

The execution time required by C4.5 Decision Tree algorithm detecting malicious traffic in different data sets using different feature dimensionality reduction methods are shown in **Table 8**.

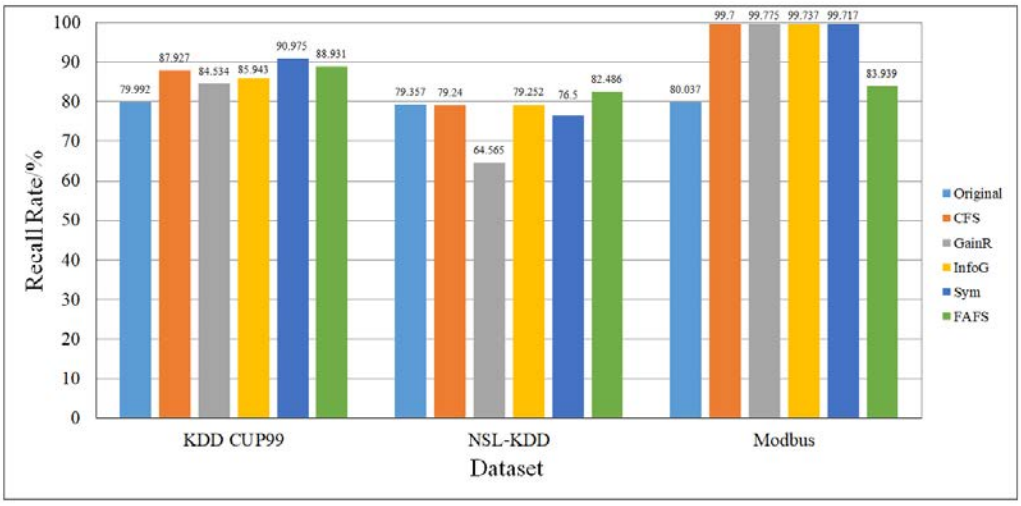
**Table 8.** Execution time required by C4.5 Decision Tree algorithm

| Data set | Time（s） | | | | | |
|---|---|---|---|---|---|---|
| | Original | CFS | GainR | InfoG | Sym | FAFS |
| KDD CUP99 | 10644.6 | 9129.24 | 2016.63 | 2377.86 | 2248.107 | 2091.07 |
| NSL-KDD | 76.95 | 8.612 | 4.108 | 4.602 | 3.947 | 4.44 |
| Modbus_traffic | 1.24 | 0.327 | 0.875 | 0.629 | 0.629 | 0.43 |

The detection effect of Naïve Bayes algorithm on different original data sets, and data sets, the feature of which are selected by CFS, GainR, InfoG, Sym feature selection method and FAFS method are shown in **Fig. 3**.




(a) Precision rate of Naïve Bayes




(b) Recall rate of Naïve Bayes

**Fig. 3.** Detection effect of Naïve Bayes algorithm

The execution time required by Naïve Bayes algorithm detecting malicious traffic in different data sets using different feature dimensionality reduction methods are shown in **Table 9**.

**Table 9.** Execution time required by Naïve Bayes algorithm

| Data set | Time（s） | | | | | |
|---|---|---|---|---|---|---|
| | Original | CFS | GainR | InfoG | Sym | FAFS |
| KDD CUP99 | 0.78 | 0.733 | 0.500 | 0.449 | 0.455 | 0.32 |
| NSL-KDD | 1.2 | 0.508 | 0.515 | 0.5107 | 0.447 | 0.41 |
| Modbus_traffic | 0.33 | 0.240 | 0.269 | 0.288 | 0.305 | 0.21 |

As shown in **Table 2**, the precision rate and recall rate of K-NN algorithm on KDD Cup99 data set are 88.473% and 98.563% based on FAFS feature selection method. Based on FAFS feature selection method, the precision rate and recall rate of C4.5 Decision Tree algorithm on KDD Cup99 data set are 82.819% and 88.113%. Based on FAFS feature selection method the precision rate and recall rate of Naïve Bayes algorithm on KDD Cup99 data set are 81.183% and 88.931%. The results show that the classification algorithm can be significantly improved by FAFS method, compared with the original data set. Similarly, based on FAFS feature selection method, the precision rate and recall rate of different algorithms on NSL-KDD and Modbus_traffic data set are improved, compared with the original data set in **Table 4** and **6**. It shows that the classification performance of learning algorithm can be improved by FAFS algorithm.

As the classical detection algorithm, K-NN algorithm is based on the distance between different eigenvalues to classify. As shown in **Fig. 1**, based on FAFS feature selection method the precision rate and recall rate of K-NN algorithm on KDD Cup99 data set are better than that of K-NN algorithm on KDD Cup99 data set based on GainR, InfoG and Sym feature selection method. As depicted in **Table 7**, the execution time required by K-NN algorithm on KDD Cup99 data set, NSL-KDD data set and Modbus_traffic data set are 1891.07s, 366.11s and 0.26s based on FAFS feature selection method. The execution time required by K-NN algorithm on KDD Cup99 data set, NSL-KDD data set and Modbus_traffic data set are 9129.24s, 504.447s and 0.2404s based on CFS feature selection method. Though the precision rate and recall rate of K-NN algorithm on KDD Cup99 data set based on FAFS feature selection method are slightly worse than that of K-NN algorithm on KDD Cup99 data set based on CFS selection method, FAFS method can significantly reduce the complexity of the detection model, the occupation of system resources, and the modeling time. Similarly, the classification results of K-NN algorithm on NSL-KDD and Modbus_traffic data set based on FAFS feature selection method are steable and the execution time required is much less than that of other feature selection methods.

According to **Fig. 2**, the precision rate and recall rate of C4.5 Decision Tree algorithm on KDD Cup99 data set are 82.819% and 88.113% based on FAFS feature selection method. Based on CFS feature selection method the precision rate and recall rate of C4.5 Decision Tree algorithm on KDD Cup99 data set are 9.138% and 10.011%. Based on GainR feature selection method the precision rate and recall rate of C4.5 Decision Tree algorithm on KDD Cup99 data set are 81.209% and 88.973%, etc. It indicates that the precision rate and recall rate of C4.5 Decision Tree algorithm on KDD Cup99 data set based on FAFS feature selection method are better than that of C4.5 Decision Tree algorithm on KDD Cup99 data set based on classical feature selection methods.

Furthermore, C4.5 Decision Tree algorithm can also search decisive features in the data set. The characteristic of C4.5 Decision Tree is that one feature can play a better role in classification after certain other features are classified. Hence, the classification results of C4.5 can be affected by the data characteristics of input data. The data selected by different feature

selection methods contains different data features. The classification results are affected differently by taking these features as the input of C4.5 Decision Tree algorithm. Therefore, the accuracy and recall rate of some data set fluctuate considerably, after selecting the features through feature selection methods such as CFS and Sym in **Fig. 2**. As shown in **Fig. 2**, the classification results of C4.5 Decision Tree algorithm on NSL-KDD and Modbus_traffic data set based on FAFS feature selection method are better than that on NSL-KDD and Modbus_traffic data set based on classical feature selection methods.

Meanwhile, as shown in **Table 8**, the execution time required by C4.5 Decision Tree algorithm on KDD Cup99 data set, NSL-KDD data set and Modbus_traffic data set are 9129.24s, 8.612s and 0.327s based on CFS feature selection method. The execution time required by C4.5 Decision Tree algorithm on KDD Cup99 data set, NSL-KDD data set and Modbus_traffic data set are 2016.63s, 4.108s and 0.875s based on GainR feature selection method. The execution time required by C4.5 Decision Tree algorithm on KDD Cup99 data set, NSL-KDD data set and Modbus_traffic data set are 2377.86s, 4.602s and 0.629s based on InfoG feature selection method. The execution time required by C4.5 Decision Tree algorithm on KDD Cup99 data set, NSL-KDD data set and Modbus_traffic data set are 2248.107s, 3.947s and 0.629s based on Sym feature selection method. The execution time required by C4.5 Decision Tree algorithm on KDD Cup99 data set, NSL-KDD data set and Modbus_traffic data set are 2091.07s, 4.44s and 0.43s based on FAFS feature selection method. It can be seen from **Table 8** that FAFS method can significantly reduce the system modeling time. FAFS method has a good effect on reducing the complexity of the system, and it also has a strong universality.

The core idea of Naïve Bayes algorithm is to select the decision with the highest probability. As can be seen from **Fig. 3,** the accuracy of the FAFS method has a relatively stable result on different data sets. In addition, as shown in **Table 9**, the execution time required by Naïve Bayes algorithm on KDD Cup99 data set, NSL-KDD data set and Modbus_traffic data set are 0.733s, 0.508s and 0.240s based on CFS method. The execution time required by Naïve Bayes algorithm on KDD Cup99 data set, NSL-KDD data set and Modbus_traffic data set are 0.500s, 0.515s and 0.269s based on GainR feature selection method. The execution time required by Naïve Bayes algorithm on KDD Cup99 data set, NSL-KDD data set and Modbus_traffic data set are 0.449s, 0.5107s and 0.288s based on InfoG method. The execution time required by Naïve Bayes algorithm on KDD Cup99 data set, NSL-KDD data set and Modbus_traffic data set are 0.455s, 0.447s and 0.305s based on Sym feature selection method. The execution time required by Naïve Bayes algorithm on KDD Cup99 data set, NSL-KDD data set and Modbus_traffic data set are 0.32s, 0.41s and 0.21s based on FAFS feature selection method. It can be seen from **Table 9**, FAFS method can ensure the detection accuracy and reduce the modeling time.

On the whole, FAFS method outperforms the traditional feature selection method. Furthermore, the overall effect of machine learning detection has declined dealing with the data set, which is similar to the real network attack. The precision rate and recall rate have been significantly improved by using FAFS method.

## 5. Conclusion

Aiming at solving the problems of too many characteristic attributes of network traffic and low accuracy of malicious traffic detection, an FAFS method is proposed in this paper. Fuzzy inference system is used to filter the rules generated by association mining and the more important features are selected, implementing the feature dimensionality reduction of high di-

mensional data of network traffic. Furthermore, the detection effect of malicious traffic in the network is improved. According to the experimental results, FAFS method has achieved good results in the feature selection of malicious traffic in network and has been applied to different data sets. Meanwhile the detection effect of different detection algorithms has been improved.

The system modeling time, data redundancy and prediction error are reduced, and the detection ability of malicious traffic in the network has been significantly improved. At the same time, compared with the traditional feature selection algorithm, the feature selection effect has also been significantly improved by FAFS method, which has a good application value in malicious traffic detection in the network.

## Acknowledgment

This work was supported by China Postdoctoral Science Foundation (2016M590234), Postdoctoral fund of Shenyang Ligong University, Project of Applied Basic Research of Shenyang (18-013-0-32), Natural Science Foundation of Liaoning Province (20180551066), Program for Liaoning Distinguished Professor, Program for Liaoning Innovative Research Team in University, Liaoning BaiQianWan Talents Program (2016) and supported by Natural Science Foundation of Liaoning Province Project (No.20170540793). The author declares that there is no conflict of interest regarding the publication of this article.

## References

[1] Jose Andre Morales, Areej Al-bataineh, Shouhuai Xu, Ravi Sandhu, "Analyzing and exploiting network behaviors of Malware," in *Proc. of 6th International Congerence on Security and Privacy in Communication Systems*, vol. 50, pp. 20-34, September 7-9, 2010. Article (CrossRef Link)

[2] Wei Wang, Yiqiang Sheng, and Jinlin Wang, Xuewen Zeng, Xiaozhou Ye, Yongzhong Huang, Ming Zhu, "HAST-IDS: Learning Hierarchical Spatial-Temporal Features Using Deep Neural Networks to Improve Intrusion Detection," *IEEE Access*, vol. 6, no. 99, pp. 1792-1806, December, 2017. Article (CrossRef Link)

[3] Christian Rossow, Christian J. Dietrich, Herbert Bos, Lorenzo Cavallaro, Maarten van Steen, Felix C. Freiling, Norbert Pohlmann, "Sandnet: Network Traffic Analysis of Malicious Software," in *Proc. of Workshop on Building Analysis Datasets & Gathering Experience Returns for Security*, pp. 77-78, April 10, 2011. Article (CrossRef Link)

[4] Xiyue Deng, Hao Shi, Jelena Mirkovic, "Understanding Malware's Network Behaviors using Fantasm," in *Proc. of LASER 2017 Learning from Authoritative Security Experiment Results*, pp. 1-11, October 18-19, 2017. Article (CrossRef Link)

[5] Razieh Sheikhpour, Mehdi Agha Sarram, Sajjad Gharaghani, Mohammad Ali Zare Chahooki. Chahooki, "A survey on semi-supervised feature selection methods," *Pattern Recognit*, vol. 64, pp. 141–158, April, 2017. Article (CrossRef Link)

[6] Zhihong Zhang, Lu Bai, Yuanheng Liang, Edwin Hancock, "Joint hypergraph learning and sparse regression for feature selection," *Pattern Recognit*, vol. 63, pp. 291-309, June, 2017. Article (CrossRef Link)

[7] Sergio Ramírez-Gallego, Héctor Mouriño-Talín, David Martínez-Rego, Verónica Bolón-Canedo, José Manuel Benítez, Amparo Alonso-Betanzos, Francisco Herrera, "An information theory-based feature selection framework for big data under apache spark," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, pp. 1441 - 1453, September, 2018. Article (CrossRef Link)

[8] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P. Trevino, Jiliang Tang, Huan Liu, "Feature selection: a data perspective," *ACM Computing Surveys*, vol. 50, pp. 94:1-94:45, 2017. Article(CrossRef Link)

[9] Wen Gao, Yaguan Qian, Chunming Wu, Ye Guo, Kai Zhu, Shuangxi Chen, "The Divide-Conquer and Voting Strategy for Traffic Feature Selection," *Chinese Journal of Electronic Science*, vol. 43, no. 4, pp. 795-799, April, 2015. Article(CrossRef Link)
[10] Fei Tang, and Hemant Ishwaran, "Random Forest Missing Data Algorithms," *Statistical Analysis & Data Mining the Asa Data Science Journal*, vol. 10, no. 6, pp. 221-246, June, 2017. Article(CrossRef Link)
[11] Xingbin Sun, Yanzan Sun, and Xiaoying Zheng, "A feature selection method for multi-class network traffic," *Computer Application Research*, vol.34, no. 2, pp. 568-571, February, 2017.
[12] Xingbin Sun, and Yun Rui, "A Statistical Frequency-Based Method for Network Traffic Feature Selection," *Small Micrcomputer System*, vol. 37, no. 11, pp. 2483-2487, November, 2016. Article (CrossRef Link)
[13] Mohd Mahmood Ali, Mohd S Qaseem, Lakshmi Rajamani, A Govardhan, "Extracting useful rules through improved decision tree induction using information entropy," *International Journal of Information Sciences & Techniques*, vol. 3, no. 1, pp. 27-41, January 2013. Article (CrossRef Link)
[14] Frederico Coelho, Antônio Pádua Braga, Michel Verleysen, "Multi-Objective Semi-Supervised Feature Selection and Model Selection Based on Pearson's Correlation Coefficient," *International Journal of Information Sciences & Techniques*, vol. 6419, no. 1, pp. 509-516, November, 2010. Article (CrossRef Link)
[15] Qilei Yin, and Pingping Wu, "Detection of Attack Time Series Association Rules Based on Apriori Algorithms," *Computer Security*, no. 9, pp. 2-7, September, 2014. Article (CrossRef Link)
[16] A. Salama, R. Saatchi and D. Burke, "Adaptive Sampling Technique for Computer Network Traffic Parameters Using a Combination of Fuzzy System and Regression Model," in *Proc. of 4th International Conference on Mathematics and Computers in Sciences and in Industry (MCSI)*, pp. 206-211, August 24-27, 2017. Article (CrossRef Link)
[17] T. V. Avdeenko and E.S. Makarova, "Integration of Case-based and Rule-based Reasoning Through Fuzzy Inference in Decision Support Systems," *Procedia Computer Science*, vol. 103, pp. 447-453, January, 2017. Article (CrossRef Link)
[18] R. Khosravanian, M. Sabah, D. A. Wood, and A. Shahryari, "Weight on drill bit prediction models: Sugeno-type and mamdani-type fuzzy inference systems compared," *Journal of Natural Gas Science and Engineering*, vol. 36, pp. 280 – 297, November, 2016. Article (CrossRef Link)
[19] KDDCup1999Data. http://kdd.ics.uci.edu/databases/kddcup99/kddcup1999.html.
[20] DARPA Intrusion Detection Evaluation. http://www.11.mit.edu/IST/ideval/index.html.
[21] Modbus_traffic. http://download.csdn.net/download/a1187006940/9540421.html.

**Yongxin Feng** received the B.S., M.S. and Ph.D. degree in Computer Application Technology from Northeastern University, China in 1997, 2000, and 2003. She is currently a professor in Shenyang Ligong University. She has published over 60 papers in related international conferences and journals. She had been awarded the ICINIS 2011 Best Paper Awards and up to 15 Science and Technology Awards including the National Science and Technology Progress Award and Youth Science and Technology Awards from China Ordnance Society. Her research interests are in the areas of Network Security, Wireless Sensor Network, Communication and Information Systems.

**Yingyun Kang** received the B.S. degree in Communication Engineering from Shenyang Ligong University, China in 2015. She received the M.S. degree in Communication and Information System from Shenyang Ligong University, China in 2018. She is currently pursuing the Ph.D. degree in Armament Science and Technology. Her research interests include Communication and Information Systems and Network Security.

**Hao Zhang** received the B.S. degree in Communication Engineering from Shenyang Ligong University, China in 2016. He received the M.S. degree in Department of Computer Engineering with Shenyang Ligong University, Shenyang, Liaoning, China. His research interests include Communication and Information Systems and Network Security.

**Wenbo Zhang** is currently a professor of School of Information Science & Engineering, Shenyang Ligong University, China. He received his Ph.D. in Computer Science & Technology at Northeastern University, China, in 2006. He has published over 100 papers in related international conferences and journals. He has served in the editorial board of up to 10 journals, including Chinese Journal of Electronics and Journal of Astronautics. He had been awarded the ICINIS 2011 Best Paper Awards and up to 9 Science and Technology Awards including the National Science and Technology Progress Award and Youth Science and Technology Awards from China Ordnance Society. His current research interests are Ad hoc networks, Sensor Networks, Satellite networks, Network Security.