A Classification Algorithm for Network Traffic based on Improved Support Vector Machine*

Lei Ding
School of Information Science and Engineering, Jishou University, Jishou 416000, China
Email: dinglei_39@ yahoo.com.cn

Fei Yu[1,] Sheng Peng[2,] and Chen Xu[1,3]
[1] Jiangsu Provincial Key Laboratory for Computer Information Processing Technology, Soochow University, 215000 Soochow, China
[2] School of Information Science and Engineering, Jishou University, Jishou 416000, China
[3] School of Information Science and Engineering, Hunan University, 416000 Changsha, China
Email: hunanyufei@126.com

***Abstract*—An algorithm to classify the network traffic based on improved support vector machine (SVM) is presented in this paper. Each feature of the traditional support vector machine (SVM) algorithm has the same effect on classification rather than considering its practical effect. To improve the classification accuracy of SVM, the probabilistic distributing area of a feature in a kind of network traffic is obtained from the real network traffic. Then the overlapped degree of the feature's probabilistic distributing area between two different kinds of network traffic is calculated to obtain the feature's contribution degree, and the corresponding weight value of the feature is derived from its contribution degree. Thus each feature has different effect on the classification according to its weight value. Considering the feature's probabilistic distributing area is affected by the outliers or noises intensively, the data space is mapped to high dimension feature space, and the Gustafson-Kessel clustering algorithm is employed to deal with the outliers or noises existing in the input samples. The experimental results show that the method presented in this paper has a higher classification accuracy.**

***Index Terms*—improved SVM, probabilistic distributing area of a feature, contribution degree, Gustafson-Kessel clustering algorithm**

I. INTRODUCTION

With the increasing development of network technology, the network applications have caused the severe shortage of the limited network resource. How to identify and control the internet traffic flows effectively is the key to solve these problems. In addition to this, how to identify the internet traffic is of great significance to network management, traffic control and anomaly detection etc. However, in order to effectively identify the internet traffic, it is necessary to understand the characteristics of internet traffic.

The classification method based on well-known port numbers is the most common traffic classification method because many traditional applications are associated with a known port number, for example the web traffic is associated with TCP port 80 [1,2]. The port-based classification method is quite simple and easily extended by adding new application-port pairs to its database. However, the identification method based on well-known ports has become more and more difficult to adapt to the development of the network technology. At present there appears to be increasing Internet applications which do not use well-known port numbers and use dynamic ports to communicate with each other. Thus the network traffic which use dynamic ports to communicate with each other cannot be easily detected by the port-based method. In addition to this, the traffic which is classified as Web may easily be something else carried over TCP port 80. So the port-based method can't adapt to a variety of network traffic.

Another traffic classification method is the Deep Packet Inspection (DPI) approach using complete protocol parsing [3]. However, there are many difficulties to use this method. For example, for the purpose of the security, some protocols are employed to encrypt the network data stream, and the contents of the data stream are intentionally hidden from sniffer programs [4]. Furthermore, some countries may prohibit netizens parse the contents of network packets, which would be a privacy violation in their view. Due to these reasons, the complete protocol parsing method is not a valid solution, and the DPI method cannot be employed to effectively identify the traffic flows based on encryption technology.

Recently, the traffic identification method using statistical characteristics have become a hot research topic, and many algorithms have been used to classify different

*Resrach supported by grants from the Science and Technology Plan of Hunan Province (2010 GK3018).

L. Ding and S. Peng are with the Jishou University, Jishou 416000, China (phone: 15174493666; fax: 0743-8564492; e-mail: dinglei_39@ yahoo.com.cn).

F. Yu is with Jiangsu Provincial Key Laboratory for Computer Information Processing Technology of Soochow University, Soochow 215325, China (e-mail: hunanyufei@126.com).

doi:10.4304/jcp.8.4.1090-1096

types of traffic flows, such as machine learning and neural network etc [5-9]. The machine learning method extracts the corresponding feature attribute through the statistical analysis of network traffic in application layer, and performs the classification using various machine learning algorithm. SVM is a new machine learning method put forward by V. Vapnik et al. and based on SLT (Statistics Learning Theory) and SRM (structural risk minimization) [10]. Compared with other learning machine, SVM has some unique merits, such as small sample sets, high accuracy and strong generalization performance etc. At present SVM has become one of the hottest research topics. For example, Alice E. et al. solved the multi-class problems with SVM to the task of statistical traffic classification, and they described a simple optimization algorithm, which use only a few hundred samples to perform correctly the classification problem [11]. Zhou X.S. employed the network traffic statistical characteristic and the SVM method based on the statistical theory to classify the different P2P traffic application [12]. Dusi M. et al. used GMM and SVM-Based statistical traffic analysis techniques to break the user behavior protection when applied to SSH tunnels [13]. Kumar S. et al. presented a P2P network traffic classification method using nu-Maximal Margin Spherical Structured Multiclass Support Vector Machine (nu-MSMSVM) classifier [14].

However, there is still a main disadvantage exiting in the traditional SVM. Although some features have been chosen to classify different Internet traffic, not every feature has the same importance. Therefore, in order to improve the identification rate, each selected feature should have a weight value to represent its importance. To improve the prediction accuracy of SVM, Wang X.Z. et al. applied mutual information values between each selected input variable and output (to be predicted) variable to calculate the weights of the input variables [15]. Gu C.J. et al. employed the contribution degree of each feature derived from the feature's information gain to improve the classification accuracy [16]. It is worthwhile to note that there are still some areas not considered before. The probabilistic distributing area of a selected feature of network traffic must be in a certain range corresponding to the network traffic type. Then the probabilistic distributing area of a selected feature can be employed to classify the network traffic. Furthermore, for a selected feature, the overlapped degree of the probabilistic distributing area between two different kinds of network traffic directly determines the feature's classification accuracy. The method presented in [15] and [16] didn't consider the influence brought by the probabilistic distributing area of a selected feature. So we can improve the accuracy using the probabilistic distributing area of a selected feature.

The feature's probabilistic distributing area is very sensitive to the outliers or noises. At present, some method based on clustering algorithm are employed to find the outliers or noises from the input samples and eliminate them. For example, Wu X.H. et al. proposed a kernel improved possibilistic c-means (KIPCM) algorithm based on fuzzy clustering algorithm to make data clustering in kernel feature space. In this paper the input data can be mapped into a high-dimensional feature space with kernel methods, and the nonlinear pattern now appears linear [17]. Yang X.W. et al. used the FCM clustering to cluster each of two classes from the training set in the high-dimensional feature space. The farthest pair of clusters is searched and forms one new training set with membership degrees. The farthest pair of clusters consists of one comes from the positive class and the other from the negative class [18]. In FCM Euclidian distance is employed to calculate the distance between data object and cluster prototype. At present there are many types of data, and clusters may have various different shapes. According to research, FCM is only suitable for clusters with a spherical shape, and does not work well when clusters' sizes or densities are different. The reason is because the distances between data points to prototypes of clusters are calculated by Euclidian distance. To solve these problems, there are some algorithms are developed, such as Gustafson-Kessel (GK) algorithm [19]. Wang J.H. et al. proposed a heuristic algorithm to reduce noises by clustering theory (GK-means). In this paper the results show that GK-means can effectively handle noises in the real-world database [20]. To effectively eliminate the outliers or noises, we can use the GK clustering to cluster each of two classes from the training set in the high-dimensional feature space in this paper.

The remainder of this paper is organized as follows. Section 2 describes in detail our approach to an improved SVM-based classifier. Section 3 describes the method to find the outliers or noises from the input samples and eliminate them, Experiment results are presented in section 4 and concluding remarks are in section 5.

II. Improved Svm Algorithm For Network Traffic

A. Original SVM Algorithm

The original SVM algorithm for classification is to seek the hyperplane, which best separates two classes of data vectors $\{x_i, y_i\}$, $i=1, \ldots, N$, $y_i \in \{-1, 1\}$, $x_i \in R^n$, where x_i is the i^{th} data vector that belong to a binary class y_i. If these data are linearly separable, we can determine the decision function:

$$D(x) = w^T x + b \tag{1}$$

where w is a vector and b is a scalar. The separating hyperplane satisfies:

$$y_i(w^T x_i + b) \geq 1 \quad \text{for } i = 1,K\ ,N. \tag{2}$$

The separating hyperplane that has the maximum distance between the hyperplane and the nearest data, i.e., the maximum margin, is called optimal hyperplane. If these data are nonlinearly separable, the input data $\{x_1, \ldots, x_N\}$, $i=1, \ldots, N$ are mapped into a high-dimensional feature space using a function column vector $\Phi(\cdot)$, and a linear regression is performed in this feature space, then we have

$$f(x) = w^T \cdot \Phi(x) + b = 0 \tag{3}$$

To obtain the two unknown variables (w, b) in equation (2), the following function should be minimized

$$\begin{aligned} &\min_{\mathbf{w},b,\xi} \|\mathrm{w}\|^2 / 2 + C\sum_{i=1}^{N}\xi_i \\ &\text{subject to } y_i(\mathrm{w}\cdot\Phi(\mathrm{x}_i)+b) \ge 1-\xi_i \\ &\qquad \xi_i \ge 0, \quad i = 1,\mathrm{L}\ ,N \\ &\qquad C > 0 \end{aligned} \tag{4}$$

where C determines the trade off between the maximum of margin and the amount up to which deviations are tolerated. and ξ_i is the slack variables. Then the dual quadratic programming classification problem can be written as

$$\begin{aligned} &\min_{a\in\Re} \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} y_i y_j a_i a_j K(\mathrm{x}_i,\mathrm{x}_j) - \sum_{i=1}^{N} a_i \\ &\text{s.t.} \sum_{i=1}^{N} y_i a_i = 0, \quad 0 \le a_i \le C, i = 1,2,\mathrm{K}\ ,N. \end{aligned} \tag{5}$$

where a_i are the Lagrange multiplier, and $K(\mathrm{x}_i, \mathrm{x}_j)$ is a kernel function defined as follows:

$$K(\mathrm{x}_i,\mathrm{x}_j) = \Phi(\mathrm{x}_i)\cdot\Phi(\mathrm{x}_j) \tag{6}$$

The vector w is

$$\mathrm{w} = \sum_{i=1}^{N} y_i a_i \Phi(\mathrm{x}_i)\,. \tag{7}$$

Let the Optimal solution be $\mathrm{a}^* = (a_1^*, a_2^*, \mathrm{L}\ , a_N^*)^{\mathrm{T}}$, and a positive value $a_j^*\ (0 < a_j^* < C)$ be selected, then

$$b^* = y_j - \sum_{i=1}^{N} y_i a_i^* K(\mathrm{x}_i,\mathrm{x}_j) \tag{8}$$

The resulting decision function is

$$f(x) = \mathrm{sgn}(\sum_{i=1}^{N} a_i^* \cdot y_i \cdot K(\mathrm{x},\mathrm{x}_i) + b^*) \tag{9}$$

The points with $a_i \ne 0$ are taken as the support vectors, and they determine the final decision result.

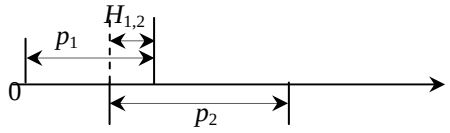
B. Improved SVM Algorithm

As described above, every feature of original SVM algorithm has the same importance, and a method to obtain the weights of the features is presented in this section.

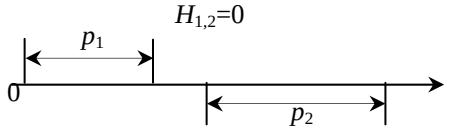
Given an input variable $\mathrm{x}_i=\{x_{i,1}, x_{i,2},\ldots, x_{i,z}\}$, where $x_{i,j}$, j=1, …, z, is a feature, and z is the dimension of the feature. The probabilistic distributing area of $x_{i,j}$ will vary within a certain range corresponding to the network traffic type. Then the overlapped degree of probabilistic distributing area of $x_{i,j}$ between two different kinds of network traffic can be used to measure the classification accuracy. For example, if the overlapped degree is zero, the selected feature can accurately classify the network traffic according to the probabilistic distributing area. But if the overlapped degree is 1, the selected feature has no ability to classify the network traffic according to the probabilistic distributing area. Furthermore, the weight of the selected feature can be derived from the overlapped degree. Suppose the probabilistic distributing area of a feature is p_1 for a given traffic type, and p_2 for another given traffic type, then $H_{1,2}$ is defined as

$$H_{1,2} = p_1 \mathrm{I}\ p_2 \tag{10}$$

where $H_{1,2}$ means the overlapped area between the two given traffic types. As Fig.1.*a* shows that if the value of a given feature locates at $H_{1,2}$, then we can't identify the network traffic type. Furthermore, the larger the $H_{1,2}$ is, the higher the probability of the feature locating at $H_{1,2}$ is. But if $H_{1,2}$ is 0, then we can identify the network traffic type using the given feature, it is shown as Fig.1.*b*



(*a*) $H_{1,2}$>0



(*b*) $H_{1,2}$=0

Figure 1. The overlapped area between the two given traffic types

Now an important task is how to measure the given feature's classification ability using $H_{1,2}$. Suppose the sample number of two kinds of network traffic is S_1 and S_2, and a feature is selected to classify the two kinds of network traffic. Let the probabilistic distributing area of the given feature of the two kinds of network traffic be p_1 and p_2, respectively. Then $H_{1,2}$ of the given feature is the overlapped area between the two given traffic types. If the sample number of the input locating at the $H_{1,2}$ is $S_{1,2}$, then we have

$$\theta_{1,2} = \frac{2S_{1,2}}{S_1 + S_2} \tag{11}$$

The above formula means the given feature will locate at the $H_{1,2}$ with the probability $\theta_{1,2}$. The conclusion can be drawn that he larger the $\theta_{1,2}$ is, the higher the probability locating at $H_{1,2}$ is. Now the feature's contribution degree can be defined as

$$t_{1,2} = 1 - \theta_{1,2} \tag{12}$$

If the $t_{1,2}$ is 1, the feature can classify the network traffic type completely. If the $t_{1,2}$ is 0, the feature can't classify the network traffic type completely. Then the matrix of the contribution degree of the features is

$$R_{1,2}=\begin{bmatrix} t_{1,2,1} & & \\ & \mathrm{O} & 0 \\ & 0 & t_{1,2,z} \end{bmatrix} \tag{13}$$

where $t_{1,2,i}$, i=1,…,z, means the contribution degree of the i^{th} feature. To suppress the weak feature, each feature has a weight value according to its contribution degree. So the kernel function can be written as $K(\mathrm{x}_i^{\mathrm{T}}R_{1,2}, \mathrm{x}_j^{\mathrm{T}}R_{1,2})$.

III. Improved Possibilistic Gustafson -Kessel Algorithm

A. Original Possibilistic GK Algorithm

The reason that FCM can only work well for the spherical shaped clusters is because in the objective function the distances are calculated by Euclidian distance.

To take into the cluster shape into consideration, one obvious choice is to incorporate the covariance matrix of each cluster into the distance calculation. So the distance is calculated as

$$d_{i,k}^2(\mathrm{x}_k,\mathrm{v}_i)=(\mathrm{x}_k-\mathrm{v}_i)^{\mathrm{T}}M_i(\mathrm{x}_k-\mathrm{v}_i) \tag{14}$$

where x_k is the value of the k^{th} sample, v_i is the i^{th} cluster centre, M_i is the covariance matrix of cluster c_i, and M_i is calculated as

$$F_i=\frac{\sum_{k=1}^{N}u_{i,k}^m(\mathrm{x}_k-\mathrm{v}_i)(\mathrm{x}_k-\mathrm{v}_i)^{\mathrm{T}}}{\sum_{k=1}^{N}u_{i,k}^m} \tag{15}$$

$$M_i=\det(F_i)^{\frac{1}{N+1}}F_i^{-1} \tag{16}$$

where $u_{i,k}\in[0,1]$ is the degree of membership of the k^{th} sample to the i^{th} cluster, and m is the fuzziness of the clusters, which determines the soft margins between clusters. Let m be 2 in this paper. Then the objective function is defined as

$$J_m(U,V,X)=\sum_{i=1}^{C}\sum_{j=1}^{N}u_{i,k}^m d_{i,k}^2 \tag{17}$$

The original GK clustering algorithm is realized as follows:

Step 1: Set c, m, and initialize U.

Step 2: Update cluster centre v_i using the formula: $\mathrm{v}_i=\dfrac{\sum_{j=1}^{N}u_{i,j}^m\mathrm{x}_j}{\sum_{j=1}^{N}u_{i,j}^m}$.

Step 3: Update F_i and M_i by formula (15) and (16), respectively.

Step 4: Update the distance by formula (14).

Step 5: Update the membership matrix U: $u_{i,j}=\dfrac{d^2(\mathrm{x}_j,\mathrm{v}_i)^{-1/(m-1)}}{\sum_{i=1}^{c}d^2(\mathrm{x}_j,\mathrm{v}_i)^{-1/(m-1)}}$. Let $u_{i,k}$ be 1 when $d^2(\mathrm{x}_j,\mathrm{v}_i)=0$, i=k, or else $u_{i,j}=0$ when $d^2(\mathrm{x}_j,\mathrm{v}_i)=0,\forall i\neq k$.

Step 6: Repeat steps 2 through 5 above until $\|U_l-U_{l-1}\|<\varepsilon$, where ε is the iterative termination index, and l is the iteration number.

B. Kernel Possibilistic GK Algorithm

The input space can be mapped into a high dimensional feature space using the theory Mercer kernel, and we have

$$\mathrm{X}=(\mathrm{x}_1,\mathrm{K},\mathrm{x}_N)\rightarrow\Phi(\mathrm{X})=(\Phi(\mathrm{x}_1),\mathrm{K},\Phi(\mathrm{x}_N)) \tag{18}$$

Then the equation (15) can be transformed as follows

$$F_i=\frac{\sum_{k=1}^{N}u_{i,k}^m(\Phi(\mathrm{x}_k)-\Phi(\mathrm{v}_i))(\Phi(\mathrm{x}_k)-\Phi(\mathrm{v}_i))^{\mathrm{T}}}{\sum_{k=1}^{N}u_{i,k}^m} \tag{19}$$

In addition to this, we have

$$\begin{aligned}(\Phi(\mathrm{x}_k)-\Phi(\mathrm{v}_i))^2=&K(\mathrm{x}_k,\mathrm{x}_k)+K(\mathrm{v}_i,\mathrm{v}_i)\\&-2K(\mathrm{x}_k,\mathrm{v}_i)\end{aligned}. \tag{20}$$

The Gaussian kernel function is used in this paper, and the equation (20) can be written as [17]

$$\|\Phi(\mathrm{x}_k)-\Phi(\mathrm{v}_i)\|^2=2-2K(\mathrm{x}_k,\mathrm{v}_i). \tag{21}$$

Then the equation (19) can be written as

$$F_i=\frac{\sum_{k=1}^{N}u_{i,k}^m(2-2K(\mathrm{x}_k,\mathrm{v}_i))}{\sum_{k=1}^{N}u_{i,k}^m}, \tag{22}$$

and the equation (14) can be written as

$$d_{i,k}^2(\Phi(\mathrm{x}_k),\Phi(\mathrm{v}_i))=M_i(2-2K(\mathrm{x}_k,\mathrm{v}_i)). \tag{23}$$

Furthermore, the cluster centre v_i is transformed as

$$\Phi(\mathrm{v}_i)=\frac{\sum_{j=1}^{N}u_{i,j}^m\Phi(\mathrm{x}_j)}{\sum_{j=1}^{N}u_{i,j}^m}, \tag{24}$$

and the membership function can be written as follows

$$\begin{aligned}u_{i,j}&=\frac{d^2(\Phi(\mathrm{x}_j),\Phi(\mathrm{v}_i))^{-1/(m-1)}}{\sum_{i=1}^{c}d^2(\Phi(\mathrm{x}_j),\Phi(\mathrm{v}_i))^{-1/(m-1)}}\\&=\frac{(M_i(2-2K(\mathrm{x}_k,\mathrm{v}_i)))^{-1/(m-1)}}{\sum_{i=1}^{c}(M_i(2-2K(\mathrm{x}_k,\mathrm{v}_i)))^{-1/(m-1)}}.\end{aligned} \tag{25}$$

Likewise, let $u_{i,k}$ be 1 when $d^2(\Phi(\mathrm{x}_j),\Phi(\mathrm{v}_i))=0$, i=k, or else $u_{i,j}=0$ when $d^2(\Phi(\mathrm{x}_j),\Phi(\mathrm{v}_i))=0,\forall i\neq k$.

The realization of the kernel possibilistic GK algorithm is the same as the original possibilistic GK algorithm.

The maximum degree of membership of a sample means which cluster it belongs to. When the data is partitioned into clusters using the clustering algorithm, a clustering centroid is calculated as follows

$$\mathrm{x}_0=\frac{\sum_{i=1}^{N_1}\mathrm{x}_i}{N_1}, \tag{26}$$

where x_0 means the clustering centroid, and N_1 means the sample number in the cluster. The average distance of the cluster is calculated as follows

$$D=\left(\frac{\sum_{i=1}^{N_1}(\mathrm{x}_i-\mathrm{x}_0)^2}{(N_1-1)}\right)^{1/2}. \tag{27}$$

Furthermore, the outliers or noises are far away from the normal samples, and they can be found only if one of the following conditions is satisfied:

- There are only a few samples or one sample in a cluster.
- The distance between a sample belong to a cluster and its clustering centroid is several times bigger than the average distance of the cluster.

In this paper, the samples of a cluster will be considered as the outliers or noises if the number of the samples of a cluster is less than 3, and the sample will be considered as the outliers or noises also if the distance between the sample and its clustering centroid is 3 times bigger than the average distance of the cluster.

IV. EXPERIMENTS

A. Experiment Data

The Moore_Set is employed to verify the method proposed in this paper and compare with other methods. There are 10 kinds of network traffic in the Moore_Set, and the total number of the samples is 377526. Table.1 describes the name, number, and the percentage of each kind of network traffic in total samples.

TABLE .1

STATISTICS OF MOORE_SET

class	*Representative Applications*	*Number*	*Ratio(%)*
WWW	www	328091	86.91
MAIL	imap, pop3, smtp	28567	7.567
BULK	ftp	11539	3.056
DATABASE	oracle, mysql	2648	0.701
SERVER	ident, ntp, x11, dns	2 099	0.556
P2P	kazaa, bittorrent	2 094	0.555
ATTACK	worm, virus	1793	0.475
MEDIA	real, media player	1 152	0.305
INT	telnet, ssh, rlogin	110	0.029
GAME	half-life	8	0.002
WWW	www	328091	86.91

The selected data set in this paper was divided equally into two subsets, one subset was the training set, and another was the measuring set. To compare with the results in [16], 0.1% of isolated samples were blent into the training set.

Each sample of the Moore_Set was sampled from a complete bidirectional network flow of TCP, and the statistical feature attribute of network traffic reflects the nature of application flow. The optimal feature attribute set in this paper was obtained through the correlation-based feature selection algorithm [16], such as { the duration time of a flow, the total number of packets of a flow, the total bytes of a flow, the average packet size of a flow, the median of packets' size of a flow, the variance of packets' size of a flow, the maximum of packets' size of a flow, the minimum of packets' size of a flow, the median of the arrival time interval of packets, the average arrival time of packets, the variance of arrival time interval of packets, the maximum of the arrival time interval of packets, the minimum of the arrival time interval of packets, the average packets' payload, the median of packets' payload, the variance of packets' payload, the maximum of packets' payload, the minimum of packets' payload }.

B. Experiment Results

In fact, the network traffic classification in this paper is a multi-class problem. At present there are two most popular methods for multiclass problem. One is one-against-all model which converts an n-class problem into n two-class problems. For example, for the i^{th} two-class problem, the optimal decision function is to separate class i from the remaining classes. Another method is one-against-one model which constructs $n(n\text{-}1)/2$ classifiers, and each one classifier is trained on data from two classes. The one-against-one model is employed to classify the network traffic in this paper. So total of $n(n\text{-}1)/2$ matrix of the feature contribution degree are needed in this paper.

To compare with other methods, the measurement index presented in [16] was used in this paper. The feedback rate expressed as b and accuracy rate q is as follows

$$\begin{cases} b=\dfrac{tp}{tp+fn} \\ q=\dfrac{tp}{tp+fp}, \end{cases} \tag{28}$$

where *tp* means the number of correctly classified samples belong to class *i*, *fn* means the number of the incorrectly classified samples belong to class *i*, and *fp* means the number of samples incorrectly classified to class *i*.

First, the data set was divided randomly into two equal subsets. Then the improved possibilistic Gustafson-Kessel algorithm was applied to eliminate the outliers and noises of the training set. Finally, the measurement set was used to measure the classification accuracy. This process was repeated 10 times, and the average values of the results of all ten consecutive tests were calculated to obtain the average classification accuracy.

Table.2 and Table.3 show that compared with the FWSVM presented in [15] and the FW-FSVM presented in [16], respectively, the improve SVM (ISVM) proposed in this paper has higher classification accuracy.

TABLE .2

COMPARISON OF CLASSIFICATION ACCURACY BETWEEN ISVM AND FWSVM WITH MOORE_SET

class	**FWSVM**		**ISVM**	
	Feedback rate	accuracy rate	Feedback rate	accuracy rate
WWW	81.25	80.67	86.24	87.18
MAIL	80.17	79.52	86.33	85.84
BULK	81.74	80.91	85.83	84.91
DATABASE	78.36	78.65	83.51	84.26
SERVER	76.98	77.42	83.79	82.57
P2P	74.83	73.15	81.66	80.51
ATTACK	75.62	74.83	80.44	81.29
MEDIA	76.64	77.56	82.13	82.52
Average values	78.199	77.839	83.74	83.63

TABLE .3

COMPARISON OF CLASSIFICATION ACCURACY BETWEEN ISVM AND FW-FSVM WITH MOORE_SET

class	***FWSVM***		***ISVM***	
	Feedback rate	accuracy rate	Feedback rate	accuracy rate
WWW	85.47	85.32	86.24	87.18
MAIL	84.18	85.79	86.33	85.84
BULK	83.26	84.67	85.83	84.91
DATABASE	80.94	81.34	83.51	84.26
SERVER	80.23	81.28	83.79	82.57
P2P	79.17	79.93	81.66	80.51
ATTACK	79.85	80.81	80.44	81.29
MEDIA	80.57	81.71	82.13	82.52
Average values	81.709	82.606	83.74	83.63

The results demonstrate the method proposed in this paper can give the proper weight to the features according to their real classification ability, and depresses the weak features.

V. CONCLUSIONS

An improved SVM algorithm is proposed in this paper to classify the network traffic. The main contribution in this paper is as follows:

- The probabilistic distributing area of a feature is employed to represent the nature of the feature.
- The overlapped area of the probabilistic distributing area of a feature between two given traffic types is applied to calculate the contribution degree of the feature.
- The input data is mapped into a high-dimensional data space, and the Gustafson-Kessel clustering algorithm is employed to cluster the input data. Then we can deal with the outliers or noises existing in the input samples.

Finally, experiment results show that the method proposed in this paper can improve the classification accuracy.

REFERENCES

[1] IANA,Port Numbers, http://www.iana.org /assignments/port-numbers.

[2] G. Cheng, S. Wang, "*Traffic classification based on port connection pattern*," *in 2011 International Conference on Computer Science and Service System (CSSS)*, 2011, pp.914 – 917.

[3] S. SEN, O. Spatscheck, D. Wang, "Accurate scalable in-network identification of P2P traffic using application signatures," *in Proceedings of ACM WWW'*04, New York, 2004, pp.512-521.

[4] D.C. Sicker, P. Ohm, D. Grunwald, "Legal Issues Surrounding Monitoring During Network Research, (invited paper)," *in Proceedings of the 7th ACM SIGCOMM conference on Internet Measurement*, San Diego, USA, 2007, pp.141-148.

[5] R. Alshammari, A.N Zincir-Heywood, "Machine Learning Based Encrypted Traffic Classification: Identifying SSH and Skype," *in IEEE Symposium on Computational Intelligence for Security and Defense Applications*, Ottawa, ON, 2009, pp. 1 – 8.

[6] T.T.T. Nguyen, G. Armitage, "A survey of techniques for internet traffic classification using machine learning," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 4, pp. 56 – 76, 2008.

[7] Murat Soysala, Ece Guran Schmidt, "Machine learning algorithms for accurate flow-based network trafficclassification: Evaluation and comparison," *Performance Evaluation*, vol. 67, no. 6, pp. 451–467, 2010.

[8] R.Y. Sun, B. Yang, L.Z. Peng, Z.X. Chen, L. Zhang , S. Jing, "Traffic classification using probabilistic neural networks," *in 2010 Sixth International Conference on Natural Computation* (*ICNC*), Yantai, Shandong, 2010, pp. 1914–1919.

[9] A.W. MOORE, D. ZUEV, "Internet traffic classification using Bayesian analysis techniques," *in International Conference on Measurement and Modeling of Computer Systems*, Alberta, Canada, 2005, pp. 50-60.

[10] Vladimir N. Vapnik, *Nature of Statistical Learning Theory*. New York: Springer-Verlag l, 1999, ch. 5.

[11] E. Alice, G. Francesco, S. Luca, "Support Vector Machines for TCP traffic classification," *The International Journal of Computer and Telecommunications Networking*, vol. 53, no.14, pp. 2476-2490, 2009,
[12] X.S. Zhou, "A P2P Traffic Classification Method Based on SVM ," *in Proceedings of the 2008 International Symposium on Computer Science and Computational Technology*, Washington, DC, USA, 2008, pp. 53-57.
[13] M. Dusi, A. Este, F. Gringoli, L. Salgarelli, "Using GMM and SVM-Based Techniques for the Classification of SSH-Encrypted Traffic," *IEEE International Conference on Communications*, Dresden, 2009, pp. 1– 6.
[14] S. Kumar, S. Nandi, S. Biswas, "Peer-to-Peer Network Classification Using nu-Maximal Margin Spherical Structured Multiclass Support Vector Machine," *Data Engineering and Management*, vol. 6411, no. 2012, pp. 80-84, 2012.
[15] X. Z. WANG,M. HAN,J. WANG, "Applying input variables selection technique on input weighted support vector machine modeling for BOF endpoint prediction," *Engineering Applications of Artificial Intelligence*, vol. 23, no. 6, pp. 1012-1018, 2010.
[16] C.J. Gu, S.Y. Zhang, "Network traffic classification based on improved support vector machine," *Chinese Journal of Scientific Instrument*, , vol. 32, no. 7, pp.1507-1513, 2011
[17] X.H. Wu, J.J. Zhou, "An Improved Possibilistic C-Means Algorithm Based on Kernel Methods," *Structural, Syntactic, and Statistical Pattern Recognition, LNCS*, vol. 4109, pp. 783-791. 2006,
[18] X.W. Yang, G.Q. Zhang, J. Lu, J. Ma, "A kernel fuzzy c-means clustering-based fuzzy support vector machine algorithm for classification problems with outliers or noises," *IEEE TRANSACTIONS ON FUZZY SYSTEMS*, vol.19, no.1,pp. 105-115, 2011.
[19] L. Teslic," Nonlinear System Identification by Gustafson–Kessel Fuzzy Clustering and Supervised Local Model Network Learning for the Drug Absorption Spectra Process," *IEEE Transactions on Neural Networks*, vol. 22, no.12, pp.1941 – 1951, 2011.
[20] J.H. Wang, J.M. Liu, Y. Zhao, B. Li, " A New Method of Eliminating Noise Based on Clustering," *International Conference on Machine Learning and Cybernetics*, Hong Kong, 2007, pp.3956– 3960.

Lei Ding was born in 1972. He received the Ph.D. degree in 2012 from central south university. Now he is an associate professor in school of information science and technology of Jishou University. His research interests include computer network, artificial intelligence and industrial process control.

Fei Yu was born in Ningxiang, China, on February 06, 1973. Before Studying in Peoples' Friendship University of Russia, Russia, He joined and worked in Hunan University, Zhejiang University, Hunan Agricultural University, China. He have taken as a guest researcher in State Key Laboratory of Information Security, Graduate School of Chinese Academy of Sciences, Guangdong Province Key Lab of Electronic Commerce Market Application Technology, Jiangsu Provincial Key Lab of Image Processing and Jiangsu Provincial Key Laboratory of Computer Information Processing Technology.

He has wide research interests, mainly information technology. In these areas he has published above 90 papers in journals or conference proceedings and a book has published by Science Press, China (Fei Yu, Miaoliang Zhu, Cheng Xu, et al. Computer Network Security, 2004). Above 70 papers are indexed by SCI, EI. He has won various awards in the past.

He served as many workshop chair, advisory committee or program committee member of various international ACM/IEEE conferences, and chaired a number of international conferences such as IITA'07, IITA'08;ISIP'08, ISIP'09, ISIP'10, ISIP'11;ISECS'08, ISECS'09, ISECS'10, ISECS'11; WCSE'08, WCSE'09, WCSE'10, WCSE'11, WCSE'12 and ISISE'08, ISISE'09, ISISE'10, ISISE'12.