Principal Component Analysis Based Network Traffic Classification

Ruoyu Yan
College of Computer and Information Engineering, Henan University of Economics and Law, Zhengzhou, China
Email: hn_yry@163.com

Ran Liu
Software Institute, North China University of Water Resources and Electric Power, Zhengzhou, China
Email: liuran@ncwu.edu.cn

Abstract—At present traffic classification is widely concerned in many research fields such as network security, traffic scheduling and traffic accounting. How to identify network traffic fast and accurately is a very meaningful thing. But most machine learning based methods have a lower speed and efficiency, and can not guarantee their stability and usability. For this reason a Principal Component Analysis (PCA) based method is proposed in the paper. At first the method use Fast Correlation-Based Filter (FCBF) algorithm to filter training data set to obtain suitable flow attributes. Then these flow attributes are processed by PCA to build feature subspace for each flow class. After that a nearest neighbor rule is used to accurately identify flow class of testing traffic sample. In the end some experiments on public data sets are done to compare performance with some existing methods. The experimental results show that the PCA based method has higher accuracy, stability and faster speed than Naive Bayes (NB) estimation method and Naive Bayes Kernel (NBK) estimation method.

Index Terms—traffic classification, Principal Component Analysis, network flow, Naïve Bayes, machine learning

I. INTRODUCTION

With the increasing of Internet scale network traffic classification is more and more important in network security, traffic scheduling and traffic accounting etc [1-3]. Due to the emerging new network applications and application layer load encryption, the traditional traffic classification methods, such as port number match [4-6] and packet payload features analysis [7-8] can not well meet the needs of various network managements. In order to overcome the deficiencies of the two kinds of methods, researchers begin to study how to use machine learning method to classify traffic fast and accurately. At present there are many machine learning methods to classify network traffic [9-19]. In order to classify network traffic efficiently most of them have to solve two problems: one is how to select suitable traffic attributes set; the other is how to select suitable machine learning algorithm to build classification model. To extract suitable traffic attributes network flow is a common used object. The so-called network flow is a unidirectional stream of packets with five tuples: destination IP, source IP, destination port, source port and layer 3 protocol type. After traffic attributes vector is determined, from the point of view of machine learning, the traffic classification problem can be described as: when the collection of network flow classes \( T = \{T_1, T_2, ..., T_k\} \) and the collection of network flows \( X = \{X_1, ..., X_n\} \) belonging to some of known classes are given, how to use machine learning method to process traffic features vector to construct traffic classification model \( f : X \rightarrow T \) and then use the model to identify unknown class of traffic.

In [8] authors propose a traffic classification method based on network flow attributes. It mainly analyzes application layer load and application traffic class is identified by attributes extracted from application layer. Although the method is effective, analysis of application layer load not only consumes computation, but also may potentially cause user privacy disputes. Besides when the application layer load or attribute fields are encrypted, the method is usually useless. In [9] authors apply one-class SVMs to traffic classification and present a simple optimization algorithm for each set of SVM working parameters. In [10] authors apply SVM method to payload-based traffic classification. Although it is more accurate to classify traffic by searching application attributes in payload content, obtaining the attributes manually is very time consuming. In [11] authors then introduce Naive Bayes (NB) method based on probability model. The method requires that network flow attributes for classification must be conditional independence and follow a Gaussian distribution. But the original flow attributes hardly satisfy the above-mentioned conditions and overall accuracy of the method is only about 65%. In order to overcome the negative effects of conditional independence assumption and Gaussian distribution assumption, in [12] authors use FCBF algorithm and Kernel Estimation (KE) algorithm to improve on primitive Naïve Bayes method. Their experimental results have shown that the classification accuracy can increase to 90% or more.
In addition, in [13] authors firstly introduce K Nearest Neighbor (K-NN) and Linear Discriminant Analysis (LDA) to solve flow classification problem. But when K-NN method is used to process testing sample, the similarity between testing sample and training sample must be calculated one by one. This would not only lead to a larger processing overhead, but also the classification performance is easily affected by noisy data. In [14] authors put forward a method which combines Gaussian mixture model and spectral clustering to process network flow attributes. Their experiments show that the method has accuracy over 90%. To achieve this goal the quality of flow attributes must be guaranteed by the arrival sequence of data packets. But the dynamic routing and network congestion stop data packets arriving in order in real network environment [15]. So the stability and usability of the method can not be guaranteed. In [16] authors propose a traffic classification method based on Rough Set Theory and Genetic Algorithm. Due to limited conditions they only do some small-scale experiments on 2254 flow records and don’t give a further comparison with the existing methods.

To sum up, most of these methods mentioned above are complicated, time consuming and inefficient, and especially the stability and usability can not be guaranteed. Although the Naive Bayes method proposed by Moore et al. [12] has higher overall accuracy and is easily implemented, the method cannot guarantee the stability of the classification result, which is verified in the following experiments. For this reason a Principal Component Analysis (PCA) based method is proposed in the paper. The method also use FCBF to filter training data set to obtain suitable flow attributes. Then PCA is applied to the processed training data set to build feature subspace for each flow class. Finally a nearest neighbor rule is used to accurately judge which class the testing traffic sample belongs to.

II. TRAFFIC CLASSIFICATION METHOD BASED ON PCA

A. The Proposed Traffic Classification Model

The detection algorithm model based on PCA is shown in Fig.1. Note that testing data set and training data set are already preprocessed by FCBF method which is introduced in [5].

![Flow classification model](image)

Figure 1. Flow classification model

PCA module: Principal Component Analysis is applied to each flow class of training data set, then for each flow class, the feature profile, namely the set of eigenvectors (U) and the set of eigenvalues (λ), is obtained. The subspace spanned by the eigenvectors is then regarded as the eigenspace of each class of flow data. Detailed steps to calculate U and λ are given in the next.

SPE module: Given a new data vector y to represent a testing flow, we project it onto a k-dimensional subspace U which represents a class of flow behavior and its reconstruction onto the subspace is written as $\hat{y} = U U^T y$. As PCA seeks a projection that best represents the original data in a least-square sense, the squared prediction error (SPE) is used in the experiments to measure the distance between original vector and its reconstruction vector: $\varepsilon = \|y - \hat{y}\|^2$. Then the SPE is a good metric to decide which class the testing flow belongs to.

Q-statistic module: This module applies a statistical test for the residual vector $\hat{y} = y - \hat{y}$ to get a threshold namely $\delta^2_{tr}$ of SPE at the $1-\alpha$ confidence level. Detailed steps to compute $\delta^2_{tr}$ are given in the next.

Flow identification module: This module identifies flow class according to the feature profile of each flow class. Detailed steps to classify flow are given in the next.

B. Using PCA to Obtain Eigenvectors and Eigenvalues

Principal Component Analysis is one of the most widely used dimensionality reduction techniques to analyze and compress data. It transforms lots of variables into less uncorrelated variables by finding a few orthogonal linear combinations of the original variables with the largest variance. In the transformation, the first principal component is the linear combination of the original variables with the largest variance; the second principal component is the linear combination of the original variables with the second largest variance and orthogonal to the first principal component, and so on. The first several principal components always contribute most of the variance in the original data set, so that the rest can be discarded with minimal loss of the variance for dimension reduction of the data [20]. The transformation works as follows.

Step 1) Given a set of observations $X_1, X_2, \ldots, X_n$ in the training data set, where each observation is represented by a vector of length p, the training data set is represented by a matrix $X_{n \times p}$

$$
X = \begin{bmatrix}
x_{11} & x_{12} & \cdots & x_{1p} \\
x_{21} & x_{22} & \cdots & x_{2p} \\
\vdots & \vdots & \ddots & \vdots \\
x_{n1} & x_{n2} & \cdots & x_{np}
\end{bmatrix}
$$

Step 2) Calculate the mean observation:
\[ \mu = \frac{1}{p} \sum_{i=1}^{p} X_i \]  

(2)

**Step 3)** Calculate the deviation from the mean:

\[ \Phi_i = X_i - \mu \quad 1 \leq i \leq n \]  

(3)

**Step 4)** Calculate the sample covariance matrix of the training data set:

\[ C = \frac{1}{n} \sum_{i=1}^{n} (X_i - \mu)(X_i - \mu)^T = \frac{1}{n} \sum_{i=1}^{n} \Phi_i \Phi_i^T = AA \]  

(4)

where \( A = [\Phi_1, \Phi_2, \ldots, \Phi_n] \)

**Step 5)** Suppose \((\lambda_1, u_1), (\lambda_2, u_2), \ldots, (\lambda_p, u_p)\) are \( p \) eigenvalue-eigenvector pairs of the sample covariance matrix \( C \), which can be computed by the Singular Value Decomposition (SVD) \([21]\). Then the \( k \) eigenvectors having the largest eigenvalues are chosen. It implies that \( k \) is the inherent dimensionality of the subspace governing the signal. The dimensionality of the subspace \( k \) can be determined by

\[ \sum_{i=1}^{k} \lambda_i \geq \alpha \sum_{i=1}^{p} \lambda_i \]  

(5)

where \( \alpha \) is the ratio of the amount of information in the subspace to the total amount of information in the original space. We arrange the set of eigenvectors (principal components) \( (u_1, u_2, \ldots, u_k) \) as columns of a matrix \( U = (u_1, u_2, \ldots, u_k) \) of size \( p \times k \) where \( k \) denotes the number of normal axes and arrange the set of eigenvalues \( (\lambda_1, \lambda_2, \ldots, \lambda_p) \) as columns of a vector \( \lambda = (\lambda_1, \lambda_2, \ldots, \lambda_p) \).

C. Using Q-statistic to Obtain Threshold

Q-statistic is a useful statistical test for the residual vector \( \tilde{y} \) to calculate threshold \( \delta_\alpha^2 \) for the SPE at the \( 1 - \alpha \) confidence level, which is developed by Jackson and Mudholkar and is given in \([22]\) as:

\[ \delta_\alpha^2 = \phi \left[ c_\alpha \sqrt{\frac{2 \phi h_0^2}{\phi_1}} + 1 + \frac{\phi h_0 (h_0 - 1)}{\phi_1^2} \right] \left( \frac{1}{h_0} \right) \]  

(6)

Where \( \phi = \sum_{j=1}^{m} \lambda_j \quad i = 1, 2, 3 \quad \), and \( h_0 = 1 - 2 \phi \phi_2 / (3 \phi^2) \). \( \lambda_j \) is eigenvalue, and \( c_\alpha \) is the \( 1 - \alpha \) percentile in a standard normal distribution. Jackson and Mudholkar’s result holds regardless of how many principal components are retained in the normal subspace.

Note that in this setting, the \( 1 - \alpha \) confidence limit corresponds to a false alarm rate of \( \alpha \), if the assumptions under which this result is derived are satisfied. An important property of this approach is that it does not depend on the mean traffic amount in the network. Thus, one can apply the same test on networks with different sizes and utilization levels.

D. Flow Identification Method

There are four steps to determine the class of testing flow:

**Step 1)** Use PCA module to profile behavior of each flow class based on training flow data. Suppose there are \( p \) classes of flow behaviors. Given \( n \) vectors \( X_1, X_2, \ldots, X_n \) representing the observations of a class of flow data as an example, the average vector \( \mu \) and each mean-adjusted vector can be computed by (2) and (3). \( p \) eigenvalue-eigenvector pairs \( (\lambda_1, u_1), (\lambda_2, u_2), \ldots, (\lambda_p, u_p) \) of the sample covariance matrix \( C \) of the data set are then calculated. The number of principal eigenvectors \( u_1, u_2, \ldots, u_k \) which are used to represent the distribution of the original data, is determined by (5). Any training data vector belonging to a certain flow class can be approximately represented by a linear combination of \( k \) eigenvectors so that the dimensionality of the data is reduced, hopefully without sacrificing valuable information. The subspace spanned by the eigenvectors is then regarded as the eigenspace of the flow class.

**Step 2)** For each flow class \( i \), use SPE module to calculate a SPE \( \epsilon_i \) between the testing data vector and its reconstruction onto subspace of the flow class.

**Step 3)** For each SPE \( \epsilon_i \), use Q-statistic module to get its threshold \( \theta_i = \delta_\alpha^2 \) at the \( 1 - \alpha \) confidence level.

**Step 4)** Identify the testing flow as a known flow class or a new flow class. For the testing vector to be identified, find the minimum \( \epsilon_i \). If the minimum \( \epsilon_i \) is below its predefined threshold \( \theta_i \), the vector is then identified as flow class \( i \). Otherwise it is identified as a new flow class.

III. COMPARISON AND ANALYSIS

A. Measurement Data Used

Data sets provided in \([12]\) are used in the paper. Theses data sets are collected from a network which hosts several Biology-related facilities. These facilities have about 1,000 users connected to the Internet via a full-duplex Gigabit Ethernet link. Traffic was monitored to generate traffic-set for a full 24 hour period and for both link directions. Because the raw traffic is so huge, traffic sampling is used by Moore et al. In order to construct the sets of flows, the day trace was split into ten blocks of approximately 1680 seconds each and the start of each
sample was selected randomly (uniformly distributed over the whole day trace). The fundamental object classified is a traffic-flow which is represented as a flow of one or more packets between a given pair of hosts. The flow is defined by five tuples consisting of the IP address of the pair of hosts, the protocol type and the port numbers used by the two hosts. In the case of TCP, a flow has a finite duration defined by the semantics of the TCP protocol. Here training and testing sets consist only of TCP and are made-up of semantically 2 complete TCP connections.

There are 377,526 flow samples altogether in data sets, which are classified into ten classes. Each flow has 249 attributes parameterizing its behavior, among which the last attribute marks its class. In the paper we only use five classes of flows as experimental data, as these flows account for about 98.8% of the entire flows. The five flow classes include: WWW, MAIL, BULK, DATABASE, SERVICE. The detailed application name, flow number and proportion of each class are shown in Table I. It can be seen that WWW and MAIL flow class has much larger flow number than BULK, DATABASE and SERVICE flow class, so WWW and MAIL flow class are called big flow class, and BULK, DATABASE and SERVICE flow class are called small flow class.

### TABLE I. INFORMATION ABOUT EACH FLOW CLASS

<table>
<thead>
<tr>
<th>Flow class</th>
<th>Application</th>
<th>Flow number</th>
<th>Percentage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>WWW</td>
<td>Http,Https</td>
<td>328,091</td>
<td>86.91</td>
</tr>
<tr>
<td>MAIL</td>
<td>Imap, Pop2/3, Smtp</td>
<td>28,567</td>
<td>7.567</td>
</tr>
<tr>
<td>BULK</td>
<td>Ftp</td>
<td>11,539</td>
<td>3.056</td>
</tr>
<tr>
<td>DATABASE</td>
<td>Postgres, Sqlnet, Oracle</td>
<td>2,648</td>
<td>0.701</td>
</tr>
<tr>
<td>SERVICE</td>
<td>X11, Dns, Ident, Ldap</td>
<td>2,099</td>
<td>0.556</td>
</tr>
</tbody>
</table>

B. Evaluation Criteria

In this paper two metrics are used to assess the performance of different classification methods. In particular, refinements to those methods will be assessed on the basis of the two evaluation criteria.

**Accuracy:** The accuracy is the raw count of flows which are correctly classified by the total number of flows. This metric can be used to describe the classification accuracy for the whole system and can also provide an accuracy of classification on a per-class basis.

**Trust:** This is a per-class measure and it is an indication of how much the classification can be trusted. In other words, this measure is a probability that a flow that has been classified into some class, is in fact from this class. The trust value is higher for a flow class, the other class of flow samples is lesser classified as the class of flow by classification model.

Accuracy and trust of a particular flow class reflect the ability of the model to classify the flow class.

C. Experimental Results Under Stratified Sampling

At first, in order to compare the classification stability of PCA and Naïve Bayes method, the data set collected by Moore (called Moore_Set) is equally divided into two data sub-set. They are Set_1 and Set_2. In Set_1 and Set_2, the proportion of each class of flow sample is consistent with that in Moore_Set. For each flow class, 0.1% of the flow samples are taken from Set_1 to compose training set. In Moore_Set, each flow samples includes 249 network flow attributes, in which there are many redundancy attributes and irrelevant attributes. Due to the lower classification accuracy and heavier computation load of the classification model caused by theses attributes, FCBF method is used to filter training set. Then after NB, NBK and PCA method are applied to the preprocessed training set, for the three machine learning algorithms, flow classification model can be learned respectively. At last, for performance verification, each classification model is applied to testing set Set_2.

For the purpose of analyzing the sensitivity of classification model to training data size, training data set is constructed respectively with the sampling scale gradually increasing to 0.5%, 1%, 5%, 10%, 50%. For each scale of training data set, the classification experiment process described before is repeated ten times. The final experimental results are shown in Fig. 2.

In Fig. 2, X-axis is Logarithmic coordinates, which represents the number of training flow sample. Y-axis represents overall accuracy of classification model. It can be shown that NB method has significantly poorer classification results. That is because NB method directly uses Gaussian distribution hypothesis which can not effectively fit the distribution of network flow attributes. Different from NB method NBK method and PCA method are able to maintain higher classification accuracy. But the classification accuracy of the two methods vibrates slightly with the increase of training data. This is mainly because FCBF algorithm filters flow attributes according to local information of the training data set, namely FCBF algorithm selects proper flow attributes under local optimality, which leads to the unstability of classification results. Besides, on the one hand NBK method needs the attribute filtering mechanism to satisfy conditional independence assumption, on the other hand the local optimality of NBK method can lead to the unstability of classification results. Therefore how to avoid local optimality and optimize flow attribute selection is still a need for further in-depth study.
Although Fig. 2 has shown the changes of overall accuracy of classification methods with the increasing of training data, in order to compare the classification accuracy and trust of each flow class, Table II and Table III show the classification results under stratified sampling when the training data size is 93236.

From Table II and Table III, it can be seen: when comparing the accuracy and trust of small flow class, such as SERVICE and BULK flow class, PCA method is significantly better than NB and NBK methods. This is because NB and NBK methods are all dependent on sample priori probability. That means bigger training set can improve the classification performance. However PCA method is independent on priori probability distribution, which can efficiently avoid this situation. In summary, PCA method is better than NB and NBK methods under stratified sampling when measuring from these performance indicators.

<table>
<thead>
<tr>
<th>Method</th>
<th>WWW</th>
<th>MAIL</th>
<th>DB</th>
<th>BULK</th>
<th>SERV</th>
</tr>
</thead>
<tbody>
<tr>
<td>NB (%)</td>
<td>97.95</td>
<td>87.52</td>
<td>1.35</td>
<td>22.50</td>
<td>8.24</td>
</tr>
<tr>
<td>NBK (%)</td>
<td>98.83</td>
<td>90.64</td>
<td>8.76</td>
<td>14.33</td>
<td>0.01</td>
</tr>
<tr>
<td>PCA (%)</td>
<td>99.27</td>
<td>96.38</td>
<td>91.42</td>
<td>89.10</td>
<td>65.17</td>
</tr>
</tbody>
</table>

TABLE III.
TRUST OF ALL CLASSES WITH STRATIFIED SAMPLING

<table>
<thead>
<tr>
<th>Method</th>
<th>WWW</th>
<th>MAIL</th>
<th>DB</th>
<th>BULK</th>
<th>SERV</th>
</tr>
</thead>
<tbody>
<tr>
<td>NB (%)</td>
<td>94.15</td>
<td>81.68</td>
<td>20.10</td>
<td>65.33</td>
<td>1.05</td>
</tr>
<tr>
<td>NBK (%)</td>
<td>94.60</td>
<td>76.82</td>
<td>48.70</td>
<td>95.68</td>
<td>9.21</td>
</tr>
<tr>
<td>PCA (%)</td>
<td>96.53</td>
<td>97.47</td>
<td>93.46</td>
<td>91.85</td>
<td>66.27</td>
</tr>
</tbody>
</table>

D. Experimental Results Under Uniform Sampling

In order to analyze the dependence degree of Naive Bayes to sample priori probability, and to further study the classification stability of PCA and Naïve Bayes method, some experiments are done under uniform sampling. Firstly, for each flow class 100 flow samples are randomly selected from Set_1 to compose a training set. That means in training set each class of flow numbers are equal. Likewise FCBF method is used to filter training data set. Then after NB, NBK and PCA method are used to process training data set respectively, flow classification model of each method can be built. Finally, for performance verification, each classification model is applied to testing data set Set_2. Again for each different flow sample number, such as 300, 500, 700 and 900, training data set is constructed as mentioned before. For each training data set, the experiments are repeated also ten times. The average experimental results are shown in Fig. 3.

In Fig. 3, X-axis represents the sample number of each flow class. Y-axis represents overall accuracy of classification model. It can be shown that, among the three methods, with the increasing of training data only the overall accuracy of PCA method improves in a relatively stable manner. However the overall accuracy of NB and NBK method does not increase, on the contrary decreases with the increasing of training data. This is not only due to the vibration caused by local optimality of FCBF, but the more important factor is the larger flow distribution differences between testing data set and training data set. Because NB method and NBK method, which are all based on Bayes' theorem, assume that a priori probability remains unchanged. When this assumption is unsatisfied, NB method and NBK method become invalid.

<table>
<thead>
<tr>
<th>Method</th>
<th>WWW</th>
<th>MAIL</th>
<th>DB</th>
<th>BULK</th>
<th>SERV</th>
</tr>
</thead>
<tbody>
<tr>
<td>NB (%)</td>
<td>43.94</td>
<td>81.85</td>
<td>37.94</td>
<td>16.45</td>
<td>87.75</td>
</tr>
<tr>
<td>NBK (%)</td>
<td>55.16</td>
<td>86.70</td>
<td>64.68</td>
<td>12.73</td>
<td>95.36</td>
</tr>
<tr>
<td>PCA (%)</td>
<td>89.07</td>
<td>92.82</td>
<td>97.50</td>
<td>90.70</td>
<td>96.28</td>
</tr>
</tbody>
</table>

It can be shown that, under uniform sampling, the accuracy of identifying WWW flow all decreases compared to that under stratified sampling when the three classification models process flow sample respectively. Among them, the accuracy of PCA method identifying WWW flow decreases about 10%. This is because the number of WWW flow in training data set is only 900, which is much smaller compared to 200000 WWW flows in whole data set. Namely the relative deficiency of sampling causes the decrease of accuracy. However the accuracy of NB and NBK method identifying WWW flow decreases about 50%. This significant decrease is
not only because under-sampling losses many information, but also because uniform sampling losses priori probability information. In summery, PCA method is better than NB and NBK methods under uniform sampling when measuring from accuracy and trust.

E. Classification Speed Comparison

In the paper, FCBF algorithm is used to select suitable flow attributes before each flow classification experiment, but each time the selected attributes are different owing to the difference of training data sets. So it is hard to measure the speed of classification method. For this reason a group of fixed flow attributes should be selected. Hence FCBF algorithm is run on whole Moore_Set data to assure the selected attributes typical and seven flow attributes are obtained. Then 10% of Moore_Set data are randomly selected as training data set, and the rest of Moores_Set data belong to testing data set. Afterwards NB, NBK and PCA methods are run to process training data set respectively to obtain flow classification model. Finally for performance verification, each classification model is applied to testing data set. Above experiment is repeated ten times. The experiments results are shown in Table VI. The testing time is average identification time for one testing sample. Comparing the experimental results of the three methods, it is clear that PCA method not only has a shorter training time, but also has an absolute advantage in testing speed. The main reason is that when PCA method classifies flow samples, it only computes SPE value and executes a simple threshold comparison. However for NB and NBK methods, they need compute flow sample probability of belonging to each flow class, which is relatively complicated. In practical application, a good classification method should have fast speed to classify flow and need not build classification model frequently in large scale network. Compared to NB and NBK methods, PCA is clearly suitable for massive network traffic.

<table>
<thead>
<tr>
<th>TABLE VI. AVERAGE TRAINING TIME AND TESTING TIME OF 3 METHODS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
</tr>
<tr>
<td>Training Time (S)</td>
</tr>
<tr>
<td>Testing Time (μS)</td>
</tr>
</tbody>
</table>

IV. CONCLUSIONS

This paper proposes a method based on PCA and flow concept to classify network traffic. FCBF algorithm firstly processes training data set to select suitable flow attributes. Next PCA algorithm analyzes these flow attributes in training data set and builds feature subspace for each flow class. When classifying testing flow sample, SPE and Q-statistic algorithm compare the distance between testing flow sample and each feature subspace to judge flow class. Compared to NB and NBK method, experiments under stratified sampling and uniform sampling demonstrate that the method has higher accuracy, trust and faster speed. Furthermore the method has better classification stability under the changing in the training data set size. For the purpose of improving on network traffic online identification, in the future some research work should be done to collect only a smaller portion of data packets in one network flow to extract flow attribute. This will identify flow class in advance greatly.

ACKNOWLEDGMENT

This work is supported mainly by National Natural Science Foundation of China (61101211, 61202285), Foundation of He’nan Educational Committee (12A520005, 13B520901), Research Programme of He’nan Fundamental and Advanced Technology of China (132300410337) and The Heng Yang Joint Funds of The Hunan Provincial Natural Sciences Foundation of China (11JJJ010).

REFERENCES


Ruoyu Yan was born in Shaoyang, China, on April, 1974. He received a M.S. degree from Beijing Jiaotong University in computer science, China, in 2004 and received a Ph.D. from Xi’an Jiaotong University in computer science, China, in 2010. Now he is a lecturer in Henan University of Economics and Law in China. His research interests include computer network, network security and database system.

Ran Liu was born in Xingyang city, Henan, P.R. China in December 23, 1979. He received master’s degree in engineering from North China University of Water Resources and Electric Power, Zhengzhou, P.R. China. He is lecturer of computer science, and the director of multimedia technology department, the Software Institute, North China University of Water Resources and Electric Power, Zhengzhou, P.R. China. He has published more than 10 papers in Journal of Science and Technology. His research interests include data mining, distributed database, and multimedia technology.