Journal of Computers Vol. 27, No. 3, 2016, pp. 81-96
doi:10.3966/199115592016102703009

The Abnormal Behavior Detection Model of the Mobile Internet Based on Selective Semi-supervised Learning

Zhen-Jiang Zhang[1,*] Yu-Wan Wang[1] Zi-Qi Hao[1]

1 School of Software Engineering, Key Laboratory of Communication and Information Systems

2 Beijing Municipal Commission of Education, Beijing Jiaotong Univerrsity, Beijing, 100044, China
zhangzhenjiang@bjtu.edu.cn, 12125046@bjtu.edu.cn, 13120069@bjtu.edu.cn

Received 1 April 2015; Revised 15 April 2015; Accepted 24 April 2015

Abstract. The mobile Internet has higher safety requirements than the traditional Internet in terms of the protection of users' privacy and behavior. In this paper, we study the key technique of the behavior audit, then combine semi-supervised learning with the selective integration technique and propose an abnormal behavior detection model based on selective semi-supervised learning. The simulation results have proven the effectiveness of the detection model.

Keywords: behavior audit, machine learning, mobile Internet

1 Introduction

In recent years, the rapid development of the mobile communication technique has resulted in the mobile Internet gradually becoming the primary information transport platform over the traditional Internet. However, safety problems with the mobile Internet have caused widespread concerns.

At the present time, the means of the information supervision technique of the traditional Internet primarily uses the techniques of intrusion detection and firewalls as defense to detect network attacks; however, these means are difficult to achieve with mobile Internet coverage. Hence, the mobile Internet currently falls short of effective protection and control measures. Moreover, firewall and intrusion detection techniques are the primary detection and protection of external attacks of system; however, it is difficult for them to provide security to the information system with more comprehensive protection. Security threats to the mobile Internet mostly come from the systems interior; protection requirements to user privacy and behaviors are extremely strict, which is on the basis of traditional Internet threats and has a security threat to mobile characteristics. Therefore, it is difficult to apply the safety monitoring technique of the traditional Internet to the mobile Internet.

As one of the important means of today's mobile Internet security supervision, the security audit technique of the mobile Internet is a kind of network security technique used to solve mobile Internet security problems. Different from the traditional Internet security audit, it mainly carries on the security audit according to the online behavior of the mobile terminal and has high real-time requirements. Therefore, it is very important to present an effective security detection model of the mobile Internet for mobile network security problems.

Aimed at the deficiency of the traditional monitoring technique, the machine learning method is introduced to the mobile Internet. Through the combination of semi supervised machine learning methods and selective integrated techniques, we put forward a kind of abnormal behavior detection model for the mobile Internet. Thus, to realize the security audit of the network behavior of mobile users, aimed at different operational behaviors of mobile users, we determine an effective classification and identify users' behavior to implement the accurate and rapid detection of abnormal behavior; thus, it can ensure the safe operation of the mobile Internet.

The rest of this paper is organized as follows. Section 2 introduces the key techniques of the existing Internet security audit. Section 3 briefly introduces machine learning and ensemble learning. Section 4

* Corresponding Author

details the proposed mobile Internet abnormal behavior detection model. Section 5 analyzes and evaluates the performance of the proposed model by using the simulation software. Finally, in Section 6, we summarize the contributions of this paper and recommend future research directions.

2 Related work

The accurate classification for users' application behavior and the correct identification of abnormal behavior is one of the important contexts of network users' behavior auditing. The essence of classifying the users' behavior is the analytical processing of network data flows from different applications.

Research conducted on the classification of network data flows initially focused on the classification techniques based on port scanning, deep packet inspection techniques (DPI) [1] and flow classification techniques [2] based on transmission behavior. However, these methods mainly rely on manual to analysis the network behavior and obtain the characteristics. Because the number of users in the mobile Internet is increasing and users' behavior is becoming more and more complex, relying on a manual analysis method is becoming more and more infeasible. With the development of intelligent techniques, machine learning methods are increasingly used for the automatic calculation of users' behavior patterns and the extraction of related characteristics from the network data, in order to automatically generate the test rules of users' behavior, which greatly reduces the development cost [3]. According to the need to mark the training samples used for the construction of traffic classification models in advance, machine learning is divided into unsupervised learning, supervised learning and semi-supervised learning.

In the literature [4], Erman et al. proposed a network core traffic classification method based on a K-Means algorithm, according to the data packet protocol type (such as HTTP, P2P, FTP), identifying and analyzing the network traffic in the entrance of the core position of the network. In the literature [5] Zhang et al. [6] proposed an online traffic identification solution method based on the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm. Regarding some initial data packets in a data flow, such as the sub-flows, they calculated the statistical characteristics of the sub-flow, proposed an online noise tolerance space based on the density to cluster the feature vector and used the advantages of the probability of the business and the application types to develop the map.

The above studies investigated flow classification methods based on unsupervised learning, realizing the classification of clusters by a clustering algorithm, and located new application types with a high probability. However, the clusters cannot identify the corresponding category of the applications. Therefore, in order to make each cluster correspond to a specific application, we need to mark the data flow in the clusters.

Supervised learning is a classification method. Its core idea is to use labeled training sample features to construct a classification rule and then map the unknown samples to the corresponding categories. In the literature [6], Nguyen et al. based their investigation on the naive Bayes algorithm and proposed the real-time traffic classification algorithms for the N data packet (near the sliding window); when near to the data flow, this algorithm does not need to grab the start of the data flow, which can identify the type of application at any time of the flow life cycle. In the literature [7] Li et al. selected various flow characteristics which had the biggest influence on classification as the characteristic value, divided the data flow into multiple types of applications by using the optimization support vector machine and established the classification model. The experimental results illustrated that selecting a different number of flow characteristics will not cause a decline in the classification accuracy. In the literature [8], Li and Zhang used the decision tree algorithm (C4.5) for the real-time identification of P2P traffic, ensured the modeling efficiency and achieved a high accuracy rate of classification. In the literature [9], Auld, Moore and Gull used the neural network technique on 246 statistical features to classify the data flow; the results revealed an accuracy rate that was higher than the naive Bayes method. Flow classification methods of supervised learning require a large amount of labeled samples to establish a classification model. However, marking the training samples with types of applications is not only very difficult, it is also costly. Consequently, supervised learning methods cannot be directly applied to practice.

In the literature [10], Erman et al. proposed a traffic classification method based on semi-supervised learning. This method used the K-Means clustering algorithm to cluster a small amount of labeled data flow samples and a large number of unlabeled samples, according to similarities with different clusters. They then calculated the mapping relationship between the cluster and the corresponding application types by using labeled samples of clusters. This method found the mapping relationship between labeled

samples and the cluster; it belonged to the maximum likelihood estimation. Experiments proved that the method had higher precision and could identify the unknown application type flow. At the same time, it changed the known application traffic of the behavior pattern. Semi-supervised learning method synthesizes the advantages of unsupervised and supervised learning methods, which only need a small number of labeled samples, but can realize a high accuracy rate traffic classification by effectively using the useful information of unlabeled samples. Therefore, this paper chose to use the semi-supervised learning method as the processing method for training samples in the system.

3 A brief introduction of the semi-supervised learning method and selective integration

This section briefly introduces the semi-supervised learning method and selective integration, which are used for the network users' abnormal behavior detection mode proposed in this paper.

3.1 Semi-supervised learning

Semi-supervised Learning is the combination of unsupervised and supervised learning methods, whose training set contains labeled and unlabeled samples. In the practical application, due to the fact that obtaining a large number of labeled samples is not only expensive, but also difficult to obtain, the number of the training samples for machine learning is often not wide and can't reflect the flow characteristics of various categories, which makes the traditional supervised learning method not classify some of the new application flows accurately.

In order to solve these problems, a semi-supervised learning method can make full use of a large amount of hidden information of the unlabeled samples. This reduces the demand of the number of labeled samples, by only using a small number of labeled samples and a large number of unlabeled samples to achieve the classification of the data flow. This is a good compromise between the classification accuracy and the number of labeled samples, thus effectively reducing the labeling cost, while improving the performance of machine learning.

3.1.1 Self-training models

The self-training model method is widely used in semi-supervised learning. The approach is based on supervised learning. More specifically, the steps are as follows. The unlabeled data is classified by using the training classifier of labeled samples. Then the data of higher classification confidence is chosen and the classification results are regarded as class labels. Next, the classifiers, together with the labeled data, are trained. The unlabeled data are then classified again. After iterative iterations, the final classification model is built.

Among the self-training model method, the nearest neighbor propagation algorithm (PNN) is a typical self-training model. It is based on 1-NN algorithm. The specific training process is as follows:

Algorithm $PNN(\{(x_i, y_i)\}_{i=1}^{l}, \{x_i\}_{i=l+1}^{l+u}, d)$

(1) Input: Labeled dataset $L=\{(x_i, y_i)\}_{i=1}^{l}$, unlabeled dataset $U=\{x_i\}_{i=l+1}^{l+u}$, and distance function $d(\bullet)$;

(2) Select an x which is satisfied with $x=\arg\min d(x,\tilde{x})$, from $U(U\neq\varphi)$, among which $\tilde{x}\in L$;

(3) Set the class label $f(x)=\tilde{y}$ of x;

(4) Add $(x, \tilde{y})$ to the labeled dataset L, and delete x from U.

3.1.2 Collaborative training

The collaborative training method is a classical semi-supervised learning method. The method assumes that the dataset has two mutually independent views and that each view can individually train a classifier, which do not interfere with each other. Transfer the unlabeled data of the higher classification confidence and the classification results of a classifier to the training samples of another classifier when carrying on the collaborative training to increase the next round of the training sample set size of another classifier. The specific process is as follows:

Algorithm $CT(\{(x_i, y_i)\}_{i=1}^{l}, \{x_i\}_{i=l+1}^{l+u}, k)$

(1) Input：Labeled dataset $L=\{(x_i, y_i)\}_{i=1}^{l}$ and unlabeled dataset $U=\{x_i\}_{i=l+1}^{l+u}$.Each sample data has

two views where $U = \{x_i\}_{i=l+1}^{l+u}$ and k is a constant;

(2) Divide the labeled dataset L into two training datasets, L_1 and L_2, through the views $x_i^{(1)}$ and $x_i^{(2)}$;

(3) Make a repetitive cycle to $U = \varphi$;

(4) Use the dataset L_1 to train the classifier f_1 and use the dataset L_2 to train the classifier f_2;

(5) Use f_1 and f_2 to classify the unlabeled data.

3.2 Selective integration

The basic idea of selective integration is to generate some basic classifiers by training samples and then combine them together in an effective way to complete the prediction of new samples. Thus, this technique can obtain better classification results than a single classifier.

Selective integration adds a new stage between the base classifier's tectonic stage and the ensemble learning integration stage in the process of integrated learning: the basic classifier selection. The primary research content's focus on the selection method of the basic classifier. The basic idea of selective integration is that multiple basic classifiers can be used; through the appropriate choice of these basic classifiers, including choosing a part to be integrated, better prediction results can be obtained than when integrating all the basic classifiers. Fig.1 illustrates the visual expression of the basic idea of selective integration.

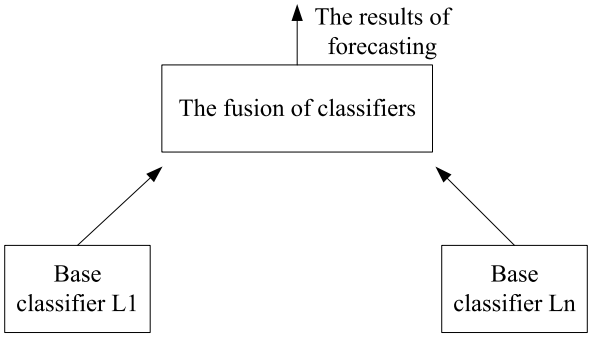


Fig. 1. The basic idea of the selective ensemble

4 The abnormal behavior detection model based on selective semi-supervised learning

This section introduces the abnormal behavior detection model of the mobile Internet selective semi-supervised learning in detail.

4.1 The relevant definition of the concept

Definition 1. Network users' abnormal behavior is a particular type of network behavior of users which influences the normal operations of the network. This paper uses the term abnormal to indicate the network users' abnormal behavior which requires the detection.

Definition 2. Detecting the characteristic factor set is the characteristic factor set which may contain the differences between the normal and abnormal behavior and can be used for statistical investigations, such as the duration of the data flow, the arrival time interval of the message and so on. This characteristic factor set is indicated by the vector $\mathbf{F}_{abnormal} = \{C_1, C_2, ..., C_n\}$, in which C_i is the i characteristic factor.

Definition 3. The network behavior data is according to the gained specific data of the network behavior after the pretreatment for the network traffic collection by the detected characteristic factor set. As for the characteristic factor set $\mathbf{F}_{abnormal} = \{C_1, C_2, ..., C_n\}$, make the collected i network row be indicated by the vector $\mathbf{Behaviour}_i = \{x_{i1}, x_{i2}, ..., x_{in}\}$, in which $x_{ij} (j \in [1, n])$ indicates the gained specific measured value with the characteristic factor C_j.

Definition 4. The mark is implementing the classification marking for the gained network behavior data through artificial or other analytical methods, indicated by t.

This paper adopts a detection method based on the SVM, because the SVM is a detection method divides dataset into two classifications: the marked value $t \in \{-1,0,1\}$, where -1 indicates the normal behavior of network users, 0 indicates that the behavior classification of the data is unknown, and 1 indicates the abnormal behavior of the network users.

Definition 5. Training sample is the partially marked network behavior data X which is used to train the basic classifier. For example, to partially mark the m network behavior data, the formed training sample X is:

$$X = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_m \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} & t_1 \\ x_{21} & x_{22} & \cdots & x_{2n} & t_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} & t_m \end{bmatrix}_{m\times(n+1)}$$

where each row consists of users' network behavior data, $x_{ij}(i \in [1,m], j \in [1,n])$, and its corresponding classification mark, $t_i \in [-1,0,1](i \in [1,m])$.

4.2 The users' abnormal behavior detection model

The abnormal behavior detection model proposed in this paper mainly consists of selective semi-supervised learning and abnormal behavior detection. The specific work process is illustrated in Fig.2.

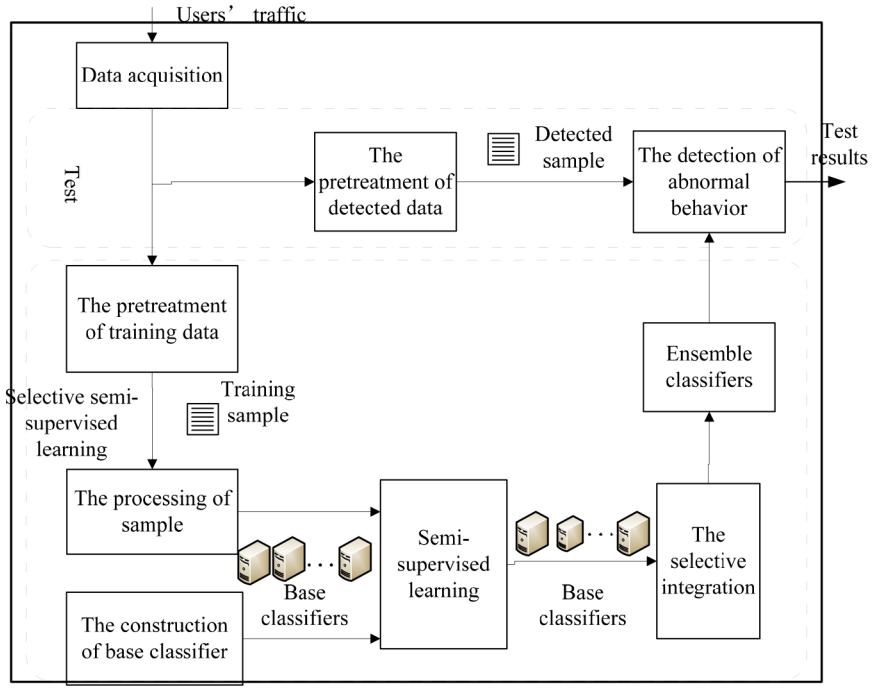


Fig. 2. Abnormal behavior detection model of the network users

The details are as follows. Collect the real-time traffic data of networks and make the pretreatment for the collected data, including the statistics and the measures to the data based on the detection of the characteristic factors. Construct the users' network behavior data. Mark on the part of the network behavior data by an artificial analysis or software tool and obtain the partially labeled training samples. Then use the improved simple integration method to deal with the training samples, including the distribution of the training samples obtained by using the clustering methods of the feature subspace. Divide the unbalanced training samples into multiple balanced training sample subsets, which reserve the original distribution information, and are trained on the base classifier. Use the training sample subspace and training

sample subset, as well as the mixed perturbation method of the base classifier parameter combination to construct the base classifiers of mutual differences. Finally, respectively train the base classifiers through selective semi-supervised learning methods and select the base classifiers based on the accuracy to construct the integrated classifier, through which we can detect the pretreated raw traffic data according to the detection features, thus achieving recognition for the users' behavior.

4.3 Selective semi-supervised learning

The anomaly detection based on selective semi-supervised learning is primarily divided into the treatment of training samples, the construction of a base classifier and selective semi-supervised learning.

4.3.1 The processing of training samples

Semi-supervised learning usually assumes that the training samples belong to the uniform distribution, which does not apply to the real network environment which exists via enormous unbalanced data. If it is used for the detection of network users' abnormal behavior directly, the generated base classifiers may be fitting. As such, it will have a negative impact on the accuracy rate of the anomaly detection.

The simple integration method can effectively deal with the unbalanced data. This method uses the integrated classification theory, based on the degree of unbalanced data. The training samples are divided into a plurality of balanced sample subsets used for training and generating a plurality of different base classifiers. The simple integration technique is not suitable for the direct introduction of an anomaly detection method based on selective semi-supervised learning.

Consequently, this paper has several objectives. It introduces the clustering technique to the simple integration method by clustering the unlabeled data to determine the ownership of the category. It analyzes the distribution information of the majority class data, so as to divide the sample data and be apt to the low proportion of data, as far as possible. This makes the number of sample subsets with a low proportion of data increase and reduces the number of high proportion data to generate a balance. The paper then tries to cover some training sample subsets for all of the category distributed information. In addition, the sample data belongs to high dimensional data, so we can use the subspace clustering algorithm in the clustering algorithms and divide the characteristic factor sets into several feature subspaces.

We elaborate the simple integration method proposed in this paper, based on clustering, which is used for the processing of the training samples.

Algorithm 1. The simple integrated training sample processing algorithm based on clustering.

Input: Training sample

$$\mathbf{X}=\begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_m \end{bmatrix}=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} & t_1 \\ x_{21} & x_{22} & \cdots & x_{2n} & t_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} & t_m \end{bmatrix}_{m\times(n+1)},$$

The detecting feature set is $\mathbf{F}_{abnormal}=\{C_1,C_2,...,C_n\}$, the feature number of detecting feature subspaces is s(s < n) , and the number of feature subspaces is N (N is the odd number).

Output: Training sample subset $\{X_1,X_2,...,X_M\}$, detecting feature subset $\{F_1,F_2,...,F_N\}$

Step 1. Randomly extract s characteristic factors from $\mathbf{F}_{abnormal}$. Repeat the operation N times and generate N feature subsets $\{F_1,F_2,...,F_N\}$, in which $F_i=\{C_1,C_2,...,C_s\}\left(i\in[1,N]\right)$ and $C_1,C_2,...,C_s\in\mathbf{F}_{abnormal}$; feature subsets are different from each other.

Step 2. Carry out the projection of the training sample X, according to every $F_i(i\in[1,N])$, and obtain the sample of the corresponding feature subspace:

$$\mathbf{X}_{F_i}=\begin{bmatrix} d'_1 \\ d'_2 \\ \vdots \\ d'_m \end{bmatrix}=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1s} & t_1 \\ x_{21} & x_{22} & \cdots & x_{2s} & t_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{ms} & t_m \end{bmatrix}_{m\times(s+1)}$$

Step 3. Carry out with the two classification clusterings for $\{d_1', d_2', ..., d_m'\}$ by the CURE algorithm[39], which uses a hierarchical clustering method, where the number K of clusters is equal to 2 and the shrinkage factor is $\alpha = 0.3$.Take out 10% from the two labeled data as the representative point. Calculate the number of elements, $t = 1$, in the two categories and let the category with the larger number be the results of the minority class $Small(F_i)$.

Step 4. Carry out a vote for all the minority class elements, $t = 0$, of $Small(F_i)$ $(i \in [1, N])$, according to the simple majority rule1 to decide whether this element belongs to the minority class or not. Combine the results of the vote with the element $t = 1$ of the training sample X to generate the minority class sample sets, $Small(Small \subset X)$.Let this set have m_1 data, thus $Small = \{d_1, d_2, ..., d_{m_1}\}$, and divide the other elements into a majority class sample set $Majority(Majority \subset X)$.Let it have m_2 data and $m_1 + m_2 = m$.

Step 5. Carry out the projection of the minority class sample set $Small$, according to every F_i, and obtain the minority class sample of the corresponding feature subspace:

$$Small'(F_i) = \begin{bmatrix} d_1' \\ d_2' \\ \vdots \\ d_{m_1}' \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1s} & t_1 \\ x_{21} & x_{22} & \cdots & x_{2s} & t_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{m_1 1} & x_{m_1 2} & \cdots & x_{m_1 s} & t_m \end{bmatrix}_{m \times (s+1)}$$

Then calculate the center position of the minority class sample:

$$\overline{Small'(F_i)} = (\overline{x}_1, \overline{x}_2, ..., \overline{x}_s), \overline{x}_i = \sum_{j=1}^{m_1} x_{ji} \Big/ m_1$$

Step 6. Carry out the projection of the majority class sample set, $Majority$, according to every F_i, and obtain the majority class sample of the corresponding feature subspace:

$$Majority'(F_i) = \begin{bmatrix} d_1' \\ d_2' \\ \vdots \\ d_{m_2}' \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1s} & t_1 \\ x_{21} & x_{22} & \cdots & x_{2s} & t_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{m_2 1} & x_{m_2 2} & \cdots & x_{m_2 s} & t_m \end{bmatrix}_{m \times (s+1)}$$

Then use the high dimensional data processing algorithm CLIQUE[13] to carry out the majority of the class clustering. Let the subspace with the largest number of categories be $Majority'(F_{\max})$.Calculate the center position for each of the subspace categories:

$$\overline{Majority'(F_{\max})} = (\overline{x}_1, \overline{x}_2, ..., \overline{x}_s), \overline{x}_i = \sum_{j=1}^{m_1} x_{ji} \Big/ m_1$$

Determine the data which has the nearest distance with the center position as a voting reference point. The distance formula is:

$$D_{d_j'} = Dist(\overline{Small'(F_i)}, d_i') = \sqrt{(\overline{x}_1 - x_{i1})^2 + (\overline{x}_2 - x_{i2})^2 + \cdots + (\overline{x}_{n'} - x_{is})^2}$$

Assume the subspace has t' categories. As such, the relevant reference point dataset is $R = \{d_1, d_2, ..., d_{t'}\}$.

Step 7. Carry out the fusion of the classification results of the other subspace. Determine the data which does not belong to the reference point dataset R from the classification results of each subspace. If the data and the reference point $x_t (x_t \in R)$ belong to the same class, then mark the data the same class as x_t.If it has multiple reference points x_t that belong to the same category, then carry out more class labels on it. After the completion of the class mark operations on the data, then carry out the vote for the data markers of the majority class sample, $Majority$.Select the class mark with the most votes as the category which the data belongs to. Then check the classification results and delete the categories which contain only a few points.

Step 8. Assume the results of the majority class are $C_1, C_2, ..., C_t$ and the numbers of the data in every

category are $Count(C_1), Count(C_2), ..., Count(C_t)$, so the sampling weights of $C_1, C_2, ..., C_t$ are:

$$Sweight(C_i) = 1 - Count(C_i) / \sum_{i=1}^{t} Count(C_i)$$

Therefore, the sample of category C_i is:

$$Size(C_i) = m_1 \bullet \frac{Sweight(C_i)}{\sum_{i=1}^{t} Count(C_i)}$$

Step 9. Calculate the number of sample subspaces $M = m_2 / m_1$ .Generate the M majority class sample set $\{X'_1, X'_2, ..., X'_M\}$. Assume the initial value of every subset is null, carry out the sampling without the replacement for every category C_i of the majority class elements. The number of the sampling is $Size(C_i)$ .If the data belongs to the category, C_i has been exhausted in the sampling process. As such, reset this class data as the initial state. Then continue to sample for M times, put respectively to each $X'_j (j \in [1, M])$.

Step 10. Construct the M training sample subset. $\{X_1, X_2, ..., X_M\}, X_i = X'_i \bigcup Small (i \in [1, M])$

Step 11. Return training sample subset $\{X_1, X_2, ..., X_M\}$ and detection feature subset $\{F_1, F_2, ..., F_N\}$.

4.3.2 The construction of the base classifier

In the achieved process of training samples by the detailed steps, form more training sample subsets, and then respectively carry out the semi-supervised learning to these sample subsets through the base classifiers of an ample number with mutual differences. In order to be able to generate a plurality of the different base classifiers, based on the SVM classifier, use the mixed perturbation method with the combination of training sample subsets, feature subspaces and SVM classifier parameters to generate base classifiers that are different from each other. As shown in Fig.3, the detailed description of the construction process is as follows.

Algorithm 2: Construction of base classifiers

```
Input:  X_j : training sample subset
F_i: feature subset
W: the parameter of disturbance
Output:  L_all: base classifier set
Y_all : corresponding training sample set
for i←1 to N
  for j←1 to M
Y_ij=ProjectionTrainingSampleSubset(X_j )
    CalculateRegLow();
    Select Parameter ξ and C
  end for
end for
for k = 1 to w
  l = ConstructBaseClassifier(ξ_k,C_k);
  L_all = l;
  Y = Y_ij ;
  Y_all = Y;
end for
```

Fig. 3. Construction of the base classifiers

Algorithm 2. The construction algorithm of the base classifiers based on the mixed disturbance.

Input: The training sample subset $\{X_1, X_2, ..., X_M\}$, feature subset $\{F_1, F_2, ..., F_N\}$, and the parameter disturbance number W .

Output: The base classifier set is $L_{all} = \{l_1, l_2, ..., l_{N \times M \times W}\}$. The corresponding training sample set is $Y_{all} = \{Y(l_1), Y(l_2), ..., Y(l_{N \times M \times W})\}$.

Step 1. For $i=1$ to N

Step 2. For $j=1$ to M

Step 3. Carry out the projection of the training sample subset X_j according to F_i.Obtain the sample Y_{ij} of the corresponding feature subset

Step 4. Analyzesample Y_{ij} and calculate the low deviation region, RegLow, by using the methods in the literature [14]. Select W pairs of the parameters of ξ and C from this region.

Step 5. For $k=1$ to w

Step 6. Construct the base classifier $l\left(l_{i\times j\times k}\right)$ through the parameter ξ_k and C_k.Add it to the base classifier set L_{all}.

Step 7. Let the corresponding training sample be $Y(l_{i\times j\times k})=Y_{ij}$.Add it to the training sample set Y_{all}.

Step 8. Return the base classifier set L_{all} and the corresponding training sample set Y_{all}.

4.3.3 The selective semi-supervised learning of the base classifier

After completing the processing of the training samples and the generation of the base classifiers, carry out the semi-supervised training for these generated base classifiers with mutual differences.

Based on the related knowledge of the selective integration technique in Section 3.3, this paper applies the selective integration technique to semi-supervised learning and the integration of base classifiers respectively to make up for the problems of traditional semi-supervised training methods such as the not-insignificant amount of time spent on the calculation of confidence when the number of base classifiers is increasing; As shown in Fig.4, the specific steps of the semi-supervised learning and the selective integration for the base classifiers are described below.

Algorithm 3. The algorithm of the base classifiers semi-supervised learning and selective integration.

Input: Base classifier set $L_{all}=\{l_1,l_2,...,l_{N\times M\times W}\}$, the corresponding training sample set $Y_{all}=\{Y(l_1),Y(l_2),...,Y(l_{N\times M\times W})\}$, stable threshold value Sa, the default number of ensemble classifiers z, the number of base classifiers in the iterative process TOP, and the threshold value of iterations Tb.

Output: Ensemble classifier $L_{resemble}=\{l_1,l_2,...,l_z\}$

Step 1. Assume the accuracy rate $A(l_i)=1$ of every base classifier $l_i(i\in[1,\ N\times M\times W])$, where the initial value of the stable value is 0.

Step 2. For every base classifier, $l_i\in L_{all}$.

Step 3. Use the labeled data training base classifier l_i of sample $Y(l_i)$.Use l_i to classify the unlabeled data in sample $Y(l_i)$.

Step 4. Carry out the integration for the classification results of the unlabeled data. The confidence of all of the data is:

$$Conf=\frac{\sum_{j=1}^{TOP}a(j)\times A(l_i)}{TOP},$$

where $a(j)$ is the decided result for the categories in which the data belongs to the TOP base classifiers l_j with the highest accuracy. Mark the data which meets $|Conf|>0.6$.Generate the updated set $\mathrm{Re}newal$ of the training sample. Assume its number is R.

Step 5. If $(R=0)$ or (Stable value $>Sa$) or (the number of iterations $>Tb$).

Step 6. Carry out "Step12".

Step 7. Else.

Step 8. Update the entire base classifier sample Y_i by using $\mathrm{Re}newal$.Check up on all the data in $\mathrm{Re}newal$.Update the accuracy of the new classifier. If the base classifier l_i classifies the r data in $\mathrm{Re}newal$ accurately, then $A(l_i)=(A(l_i)+r/R)/2$.

Algorithm 3: the base classifiers semi-supervised learning and selective integration.

```
Input: L_all: baseclassifier set
Y_all :corresponding training sample set
Sa: stable threshold value
z: the default number of integration classifiers
TOP: the number of base classifiers in the iterative process
Tb: the threshold value of iterations
Output:L_resemble: ensemble classifier
Accuraacy←1;
InitialStableValue←0;
for l_i ∈ L_all do
   Training(l_i);
    ClassifyUnlabeledData(Y);
    IntegrateUnlabeledData();
  Confidence=EveryDataConfidence();
    if(|Confidence|>0.6) then
      MarkTheData();
      GenerateTrainingSampleUpdationSet(Renewal);
    end if
    if((R=0) or (StableValue>Sa) or (the number of iterations >Tb)) then
     Execute 30;
    else
      UpdateAllBaseClassifierSamples(Y_i);
      CheckupAllData(Renewal);
      UpdateNewClassifierAccuracy();
       if(right)then
         ReaccurateAccuracy();
       end if
      end if
   SelectBaseClassifiersOfHighAccuracy(TOP);
   if(this==last)then
     StableValue←AtableValue+1;
   else
     StableValue←0
   return 5
   L_resemble=SelectBaseClassifiersOfHighAccuracy(z);
```

Fig. 4. The base classifiers semi-supervised learning and selective integration

Step 9. Pick out the *TOP* base classifiers with the highest accuracy. If the composition is the same as the last one, add one to the stable value, otherwise set the stable value to 0.

Step 10. Return to "Step2".

Step 11. Pick out the z base classifiers with the highest accuracy to form the ensemble classifier $L_{resemble} = \{l_1, l_2, ..., l_z\}$.

Step 12. Return to $L_{resemble}$.

4.4 The abnormal behavior detection of the network user

In the phase of the network abnormal behavior detection, input the network behavior data generated through the data preprocessing stage into the ensemble classifiers. Because different base classifiers are often established based on different feature subspaces, we need the projection to the network behavior data which will be detected according to the corresponding feature subspace of every base classifier before inputting the data and obtain the samples which will be detected. When each base classifier of the ensemble classifier outputs the classification results of their own, vote for the results by a simple majority and obtain the final detection results.

5 Analysis of the experiments and the results

5.1 Experiment scene

In order to prove the validity of this detection method, we use the real network environment of the campus network LAN of the computer network laboratory to carry out an experimental verification. In the course of the experiment, install attackers on partial terminals, and initiate network abnormal behavior, such as DoS and DDoS attackers in this network environment, through the remote controlled attacker. At the same time, operate all kinds of normal operation behaviors by remote control, such as browsing webpages, downloading P2Ps, chatting and watching videos online. This process can be used to collect abnormal and normal network behavior data for the experiment.

For the class label problems of the training sample data, we mark the normal network behavior by using the traffic analysis tool L7-filter and mark the abnormal network behavior according to the abnormal operation behavior implemented in the experiments. Then we construct the training samples. The experiment captured the network traffic from 9:00 am to 21:00 pm from March 24th, 2014 to March 28th, 2014. We refer to the measurement metrics in the literature [41] to measure the captured network traffic. The used measurement metrics are shown in Table 1.

5.2 Evaluation index

This paper used the semi-supervised learning method, which has the semi-supervised learning ability based on a CoForest algorithm: naive Bayes, SVM and the base classifier construction method of C4.5. The detection method proposed in the paper was compared with the semi-supervised learning method to verify the accuracy of the method.

In the machine learning literature, we usually use Recall (i.e., the recall ratio), Precise (i.e. the precision ratio) and the harmonic mean F-measure of the two to evaluate the classification results. As such, this paper uses three evaluation indicators as the detection method.

Table 1. The setting of the detection feature index set

Feature category	Detection feature index
Features of data packets	The average length of the packets, packet arrival intervals, the rate of sending packets, the maximum length of the packet , the minimum length of the packet, the average length of the packet header, the maximum length of the packet header, the minimum length of the packet header, the maximum number of data packets, the number of uploading data packets, and the number of downloading data packets
Features of data flow	The number of source ports, the number of destination ports, the number of destination addresses, the average flow length, the maximum flow length, and the minimum flow length
Features of application layer	Upload traffic flow, download traffic flow, the upload traffic flow of UDP, the download traffic flow of UDP, the upload traffic flow of TCP, and the download traffic flow of TCP

We let TP be the sample size of the accurately identified variables in the detected network behavior. We let TP be the number of samples which are accurately identified for abnormal behavior in the detected samples of network behavior, FN be the number of samples considered as normal behavior by the error, FP be the number of samples considered as abnormal behavior by the error, and formulate each evaluation index as follows:

$$Recall:\ R = \frac{TP}{TP + FN} \tag{1}$$

$$Precise:\ P = \frac{TP}{TP + FP} \tag{2}$$

$$F\text{-}measure:\ F = \frac{2 \times R \times P}{R + P} \tag{3}$$

5.3 The analysis of the results

This paper confirms whether the proposed detection method can ensure the accuracy rate and reduce the demand of the labeled training sample data with the network data of unbalance. To accomplish this, the paper first verifies the accuracy rate of the proposed method in a different balance degree of network data and then verifies the performance of the proposed method in different proportions of labeled samples.

Experiment A. Verify the accuracy rate of the proposed method in different balance degrees of network data.

The experimental procedures are as follows:

(1) Calculate the percentage of abnormal network behavior for different user behavior datasets;

(2) Use the datasets of different abnormal behavior ratios to construct the training samples and carry out the selective semi-supervised learning; and

(3) Take out the same validation data randomly from all network behavior data obtained in the experiment and then compare the proposed detection method with other comparative methods about the accuracy rate in a different balanced degree of training samples.

The data in Experiment A was set in Table 2. The sources of the experimental datasets are described as the introduction of the 4.4.1 experiment scene. The data are captured from a real network environment of a laboratory small LAN in the campus network.

The comparative analysis results of Experiment A are illustrated in Fig.5. Graphs (a), (b), and (c) are the separate comparison results of each assessment index when the abnormal behavior data ratios in the training datasets are 1.10%、4.77%、9.78%、20.62%、31.56% and 41.76%. We can see from the figures that the detection accuracy rates of the detection methods proposed in this paper that are based on the selective semi-supervised learning, such as recall, precise and the harmonic mean F-measure, are higher than other comparison methods in different degrees of unbalanced conditions. This is especially true in the low ratio of abnormal behavior data, such as the ratio of abnormal behavior that are 1.10% and 4.77%, where the method proposed in this paper still has an accuracy rate of 80%, while the top rate of the other three methods is only around 75%.

Table 2. Data setting of Experiment A

Datasets	The number of normal behavior data (bar)	The number of abnormal behavior data (bar)	The ratio of abnormal behavior (%)	The ratio of labeled samples (%)
I	148	13380	1.10	30
II	3818	80030	4.77	30
III	5180	52941	9.78	30
IV	20105	97498	20.62	30
V	20793	65891	31.56	30
VI	34821	83381	41.76	30

The results show that the selective semi-supervised learning was an effective treatment on the imbalanced data through the sample processing stage and carries out the selective integration to the base classifiers by using the selective integration technique. All of these techniques improved the accuracy of the detection method.

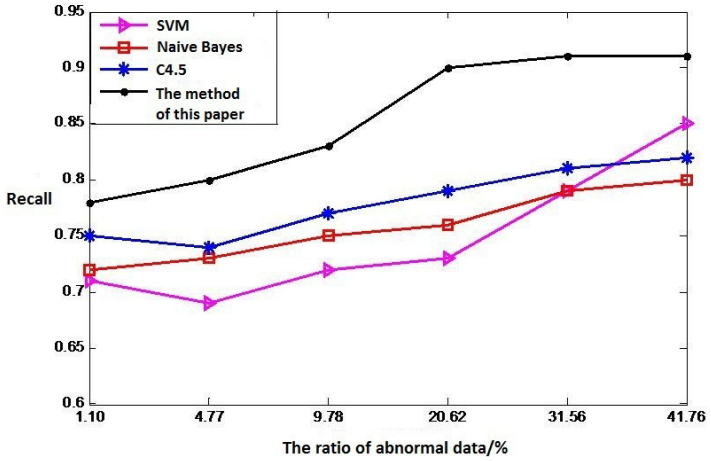
Experiment B. Verify the performance of the proposed method in different proportions of labeled samples

The experimental procedures are as follows:

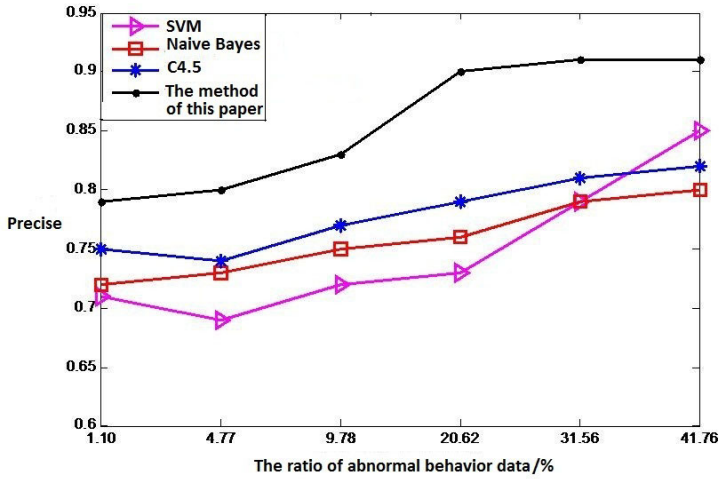
(1) Choose a certain dataset to construct the training samples. The dataset with an abnormal behavior data rate of 20.62% was chosen in this paper.

(2) Mark the partial data of this training sample, construct a training sample of some different labeled data ratios, and carry out the selective semi-supervised learning.

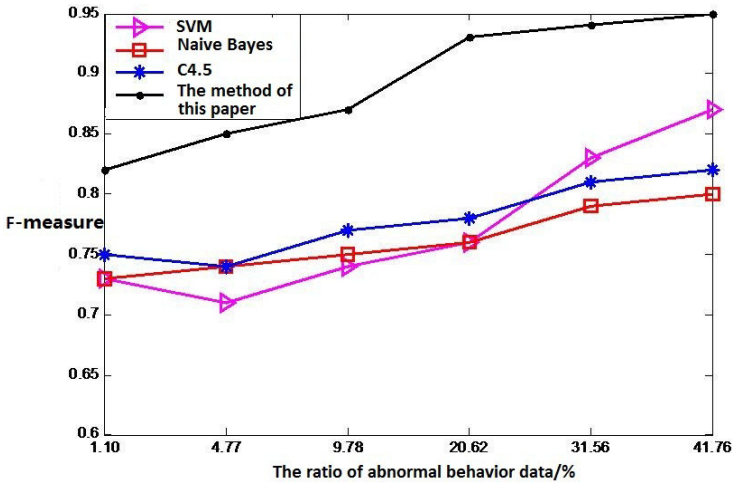
(3) Through the same validation data, compare the accuracy of this detection method with other comparison methods. The dataset sources of Experiment B are the same as Experiment A. The following data set of experiment: B shown in Table 3.



(a) The comparison results of Recall



(b) The comparison results of Precise



(c) The comparison results of the F-measure

Fig. 5. The results of Experiment A

Table 3. Data setting of Experiment B

Datasets	The number of labeled sample data (bar)	The number of unlabeled sample data (bar)	The ratio of the labeled sample (%)	The ratio of the abnormal behavior (%)
I	5870	111700	5	20.62
II	11760	105830	10	20.62
III	17642	99951	15	20.62
IV	23513	94064	20	20.62
V	17641	99951	25	20.62
VI	35280	82314	30	20.62

The comparative analysis results of Experiment B are illustrated in Fig.6. Graphs (a), (b), and (c) are the separate comparison results of each assessment index when the labeled data ratios in the training datasets are 5%、10%、15%、20%、25% and 30%. We can see from the figure that the detection accuracy rates of the detection method proposed in this paper was based on selective semi-supervised learning, such as recall, precise and the harmonic mean F-measure. The results are higher than those of the other comparison methods in the different labeled sample data ratios. This is especially true for the low ratio of labeled data, such as the ratio of labeled data being only 5%; the method proposed in this paper still has an accuracy rate of 80%.

In addition, from the figure accuracy change trend, we can see that the detection accuracy of the comparison method increases linearly with the increasing proportion of the labeled data. This is especially true of the proportion of labeled samples that is 20% to 30%.Its accuracy is less than 80%.The accuracy of the ratio interval in the method proposed in this paper tends to be stable. The performance is stable when the ratio of the labeled data is increased to more than 30%, with an accuracy of about 90%. The results show that the accuracy of the comparison methods depend on the number of labeled data in the training samples. The detection method is more suitable for the unbalanced training samples, as it can make use of fewer labeled samples to train and can achieve a better detection performance.

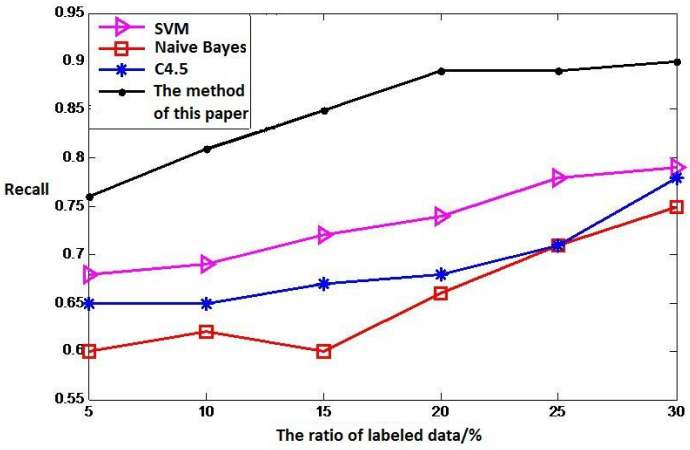
6 The Summary and the Outlook

This paper proposed an abnormal behavior detection model based on selective semi-supervised learning, according to the problem of the mobile Internet security audit. The model, which chose the method of semi-supervised learning, was used for the processing of training samples in the system. At the same time, in view of the defects of integration learning in the fusion process of base classifiers, this paper chose the selective integration learning method whose performance was better than the integration learning to make further improvements in the prediction performance of the decision model. The simulation results proved that the abnormal behavior detection scheme proposed in this paper could ensure the accuracy rate when reducing the demand of the labeled training sample data.

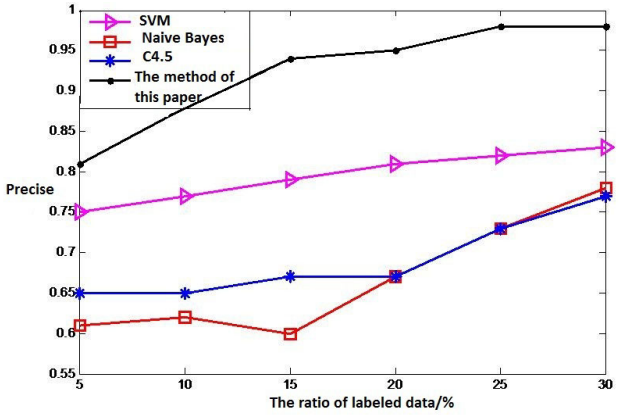
In terms of improvement, we consider the following aspects for future research:

(1) In this paper, the effectiveness and superiority of the abnormal behavior detection model were verified in a relatively simple network environment with less range, whose performance also needs to be further validated and compared with the more complex and larger real network environment.

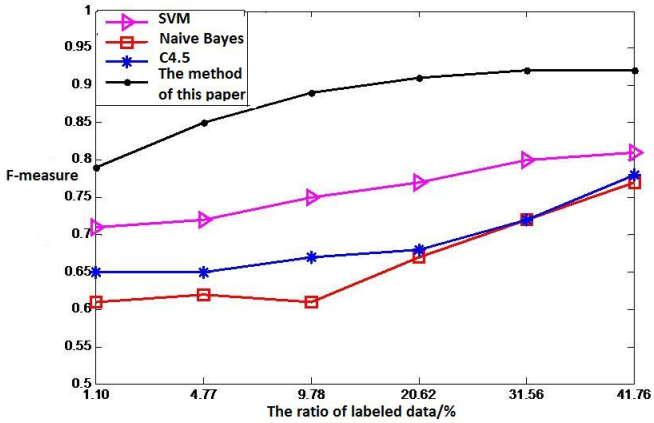
(2) This paper focuses on the detection and analysis of the correct classification and abnormal behavior of network users in the mobile Internet, which is mainly for the audit of mobile Internet behavior. In many cases, an audit of the mobile Internet content security should also be carried out to timely discover the illegal information content in a mobile network and effectively prevent its transmission, so as to ensure the development of a mobile network that is more healthy and civilized. As such, we recommend the research of the security audit of the mobile Internet content as important future work.



(a) The comparison results of Recall



(b) The comparison results of Precise



(c) The comparison results of the F-measure

Fig. 6. The results of Experiment B

References

[1] Wu, J., & Du, Z.-H. (2014). A network flow identification scheme based on DPI. *Software Guide, 1*(1), 23-26.

[2] Yan, X.-J., & Yu, G.-H. (2011). The management of campus network anomaly traffic based on OpenFlow. *Network and Communication, 32*(24), 53-55.

[3] Chauhan, A., Mishra, G., & Kumar, G. (2011). Survey on data mining techniques in intrusion detection. *International Journal of Scientific & Engineering Research, 2*(7), 1-4.

[4] Erman, J., Mahanti, A., & Arlitt, M. (2007, May). *Identifying and discriminating between web and peer-to-peer traffic in the network core*. Paper presented at WWW'07: Proceedings of the 16th international conference on World Wide Web. Banff(Canada), New York, NY.

[5] Zhang, J., Qian, Z.-J., & Shou, G.-C. (2011). The network traffic identification of online clustering. *Journal of Beijing University of Posts and Telecommunications, 34*(1), 103-106.

[6] Nguyen, T., & Armitage, G. (2006, July). *Training on multiple sub-flows to optimise the use of Machine Learning classifiers in real-world IP networks*. Paper presented at the IEEE 31st Conference on Local Computer Networks, Tampa, FL.

[7] Li, Z., Yuan, R., & Guan, X.-H. (2007, June). *Accurate classification of the internet traffic based on the SVM method*. Paper presented at the Proceedings of the 42th IEEE International Conference on Communications, Glasgow, Scotland.

[8] Li, J., Zhang, S.-Y., & Lu, Y.-Q. (2008, November). *Real-time P2P traffic identification*. Paper presented at the Global Telecommunications Conference, San Diego, CA.

[9] Auld, T., Moore, A. W., & Gull, S. F. (2007). Bayesian neural networks for machine learning techniques. *IEEE Trans. Neural Networks, 18*(1), 223-239.

[10] Erman, J., Mahanti, A., Arlitt, M., Cohen, I., & Williamson, C. (2007). Semi-supervised network traffic classification. *ACM SIGMETRICS Performance Evaluation Review, 35*(1), 369-370.