Soft Comput (2017) 21:1963–1975
DOI 10.1007/s00500-015-1892-1

METHODOLOGIES AND APPLICATION

A novel semi-supervised learning method for Internet application identification

Zhenxiang Chen[1] · **Zhusong Liu**[2] · **Lizhi Peng**[1] · **Lin Wang**[1] · **Lei Zhang**[1]

Published online: 4 November 2015

Abstract Several methods based on port, payload, and transport layer features have been proposed to detect, identify, and manage Internet traffic. The diminished effectiveness of port-based identification and overheads of deep packet inspection methods motivated us to identify Internet traffic by combining distinctive flow characteristics with the machine learning method. However, the abundant ground truth Internet traffic, which is important for building a supervised classifier, is difficult to be obtained in real conditions. In this study, we propose a semi-supervised learning method that combines further division of recognition space technique with data gravitation theory. The further division of recognition space classifier is a powerful multi-classification tool that can be helpful for multi-application identification. The data gravitation may reveal the underlying data space structure from unlabeled data, and thus, it is integrated into the classification to develop a better classifier. The experimental results on the real Internet application traffic datasets demonstrate the advantages of our proposed work.

Keywords Semi-supervised learning · Recognition space · Data gravitation · Internet traffic classification

Communicated by V. Loia.

✉ Zhenxiang Chen
czx.ujn@gmail.com; czx@ujn.edu.cn

Zhusong Liu
25421944@qq.com

Lizhi Peng
plz@ujn.edu.cn

Lin Wang
ise_wangl@ujn.edu.cn

Lei Zhang
zhanglei@ujn.edu.cn

[1] Shandong Provincial Key Laboratory of Network Based Intelligent Computing, University of Jinan, Jinan 250022, Shandong, People's Republic of China

[2] School of Computer Science and Technology, Guangdong University of Technology, Guangzhou 510006, Guangdong, People's Republic of China

1 Introduction

In recent years, the growth of bandwidth-hungry Peer-to-Peer (P2P) applications and quality-of-service guarantees sensitive applications motivate the demand for bandwidth management and network performance optimization as a hot research topic. Going by measurement studies in the literature and estimates by industry experts (Gan et al. 2013), P2P now accounts for 60–70 % of the Internet traffic. It is, therefore, unsurprising that many network operators are interested in tools to manage traffic so that traffic critical to business or traffic with realtime constraints will be given higher priority service on their network. Critical for the success of any such tool is its ability to accurately identify and categorize each network flow by the application responsible for the flow Erman et al. (2007a, b).

Monitoring network and management depends on the network traffic classification and application identification. Traditional traffic classification methods based on the port and the application payload are unable to meet the requirement of practical traffic classification. The reason for this inability is that there is an increasing number of applications using port disguise, which is the data encryption of the application layer and other circumvention technology. The traffic classification method based on the Internet traffic statistic and behavior features has become widely approved. In the traffic classification method that is based on the traffic statistics, the traditional machine learning method includes

the supervised learning (Auld et al. 2007; Roughan et al. 2004) and the cluster learning (McGregor et al. 2004; Zander et al. 2005a, b; Erman et al. 2006) without supervision. While some results were obtained, the traffic classification based on the supervised learning algorithm depends on several labeled samples to train the classifier for effective work. However, the abundant ground truth Internet traffic, which is important for building a supervised classifier, is difficult to be obtained in real conditions for the plenty of time and space resources costs. Unsupervised learning method does not demand labeled samples, but the accuracy of the classifier toward the idiographic application is not ideal.

Identifying and categorizing network traffic by application type is challenging because of the continued evolution of applications, especially of those with a desire to be undetectable. The diminished effectiveness of the port-based identification and the overheads of deep packet inspection approaches motivate us to classify traffic by exploiting the distinctive flow characteristics of applications when they communicate on a network. Considering the Internet traffic characters and the requirements of application identification, in this paper, we proposed a novel semi-supervised classification method combined with Further Division Recognition Space (FDRS) (Chen et al. 2009) and Data Gravitation (DG) theory (Peng et al. 2009). This paper studies and implements a semi-supervised learning model that is applied to traffic identification. Experiment results show that the method can obtain higher identification capability. Meantime, the method has greater ability of discovering new applications, which shows effectiveness in Internet application identification.

The remainder of the paper is organized as follows. In Sect. 2, some related works about the current research achievement of traffic classification and semi-supervised learning method are reviewed. The detailed description of the proposed semi-supervised scheme combined with the data gravitation and the further recognition space method is presented in Sect. 3. In Sect. 4, the experiment and evaluation criterion are addressed. The semi-supervised classification result and analysis are presented in Sect. 5. We devote the final section to some key concluding remarks and future works.

2 Related work

With the development of port disguise, payload encryption, and traffic-ensconced technologies, the traffic identification methods based on applications port, payload, and obvious static features have already been outdated (Karagiannis et al. 2004; Ohzahata et al. 2005; Karagiannis et al. 2005; Shi et al. 2010). The traffic identification method, which is based on transmission and application layer features (Karagiannis et al. 2004; Ohzahata et al. 2005; Karagiannis et al. 2005; Fbrega et al. 2011; Beitollahi and Deconinck 2014) is more effective. With the development of data mining technology, traffic identification based on the machine learning (Nguyen and Armitage 2008; Upadhyaya 2013; Ye and ChoKyungsan 2014; Chiou et al. 2014) has become a widely researched topic at present. The statistic features of traffic (Este et al. 2009; Lakhina et al. 2005) are dynamic and difficult to be disguised. In this light, the traffic identification method (Zhang et al. 2013; Iliofotou et al. 2011; Gmez et al. 2013; Imai et al. 2013), which combines the traffic statistic feature with machine learning, is worthy of investigation. However, the traditional supervised learning method needs numerous "labeled" samples to build a classifier. Obtaining several actual traffic samples with classified "label" for this method in real condition is difficult. Meanwhile, the unsupervised learning (clustering) (McGregor et al. 2004; Lu et al. 2011) can work without the support of "labeled" samples. However, it can only distinguish different sorts and does not have the ability to identify an application.

The semi-supervised learning method (Yu et al. 2010; Gan et al. 2013) can utilize some unknown knowledge of sorting samples to improve the effect of machine learning. This method plays an important role in the areas such as speech identification (Yu et al. 2010; Li et al. 2012), text paper identification (Yan et al. 2013), Smart cloud service (Esposito et al. 2015) and natural language disposal (Tur et al. 2005; Ficco et al. 2015). In cases when obtaining a sort labeled traffic sample is difficult, the effect of traffic identification can be effectively improved by adopting the semi-supervised learning to utilize the unlabeled traffic sample knowledge. Erman et al. (2007a, b) firstly imported the semi-supervised learning method into the research of traffic identification for good consultation effect. However, this kind of research adopts the k-means method, which utilizes partial feature information to clustering. Application classified matching adopts simple probability, but it does not have the ability to identify a new application. Other researches (Qian et al. 2008; Lin et al. 2014; Chen et al. 2013; Zhang et al. 2014) showed that the semi-supervised learning method has great ability for identifying an Internet application.

A novel semi-supervised classification method combined with Data Gravitation (DG) theory (Peng et al. 2009) and Further Division Recognition Space (FDRS) (Chen et al. 2009) model was proposed in our research. "Constriction", a kind of data disposal technology, was proposed in Shi and Zhang (2004) Indulska and Orlowska (2002). This technology is engendered by the elicitation of the concept of universal gravitation of Newton. The technology utilizes the universal gravitation to optimize the inside structure of data. Meanwhile, a previous study employed GRAVI clustering (Indulska and Orlowska 2002), which is

a kind of space clustering algorithm that utilizes the calculation of clustering center gravitation to achieve the best clustering effect. Data classification and evaluation were performed by using the data gravitation theory (Indulska and Orlowska 2002). Our study proposes further division of recognition space to decrease the classification error rate of blurry space. A study (Auld et al. 2007) reports that the improvement in the neural network classified method by theory is effective. This proposed scheme can significantly improve classification accuracy. The present study combines the theory method based on cluster in actualizing the further division of recognition space (Chen et al. 2009). We obtain the classified result of clustering to develop an entire semi-supervised learning model for real Internet traffic identification.

3 Classification method

To address the above challenges, we design a semi-supervised learning scheme that combines an unsupervised and a supervised methods. The process is illustrated in Fig. 1, and the detail steps is outlined in the following. Firstly we use our designed TL (Peng et al. 2014) system to label the application traffic (Fig. 1a, b). The TL system captures all user socket calls and their corresponding application process information in the user mode on a Windows host, and then it sends the information to an intermediate NDIS driver in the kernel mode to modify the TOS field of the IP packets. Secondly, we employ a data gravitation-based clustering algorithm to partition a training data set that consists of scarce labeled flows combined with abundant unlabeled flows (Fig. 1c). Thirdly, we use the available labeled flows to obtain a mapping from the clusters to different known classes (Fig. 1d). This step also allows some clusters to remain unmapped, which is accounting for possible flows that have no known labels. The result of the learning is a set of partitions, and some are mapped to different flow types.

3.1 Data gravitation model

3.1.1 Law of gravity

A force existing between any two objects in the universe is called gravitation in Physics. Gravitation follows the universal law of gravitation. In 1687, Newton published an important study that first illustrated the universal law of gravitation. The law indicates that the strength of gravitation between two objects has a direct ratio to the product of the masses of the two objects, but it has an inverse ratio to the square of the distance between them. The law can be illustrated by formula (1).

$$F = G\frac{m_1 m_2}{r^2} \tag{1}$$

Given that force has direction, the precise description of the law takes the following vector form, which is shown in formula (2):

$$F = G\frac{m_1 m_2 r}{|r|^3}, \tag{2}$$

where F is the gravitation between two objects, G is the constant of universal gravitation, m_1 is the mass of object 1, m_2 is the mass of object 2, r is the distance between the two objects, F is the vector form of F and r is the vector form of r.

3.1.2 Gravitation and data similarity

In the data space, the relationship between data points (samples) is not isolated. When clustering analyzes the data, which can be operated in a computer, the Euclid distance (Hrubeš 2012) in the two data points of the data space becomes farther. When a shorter distance between the two data points is minimal, this finding implies that these data points belong to the same degree of clustering. Most methods of clustering analysis are confined to the local area of the analyzed data, and they have neglected the relationship between the

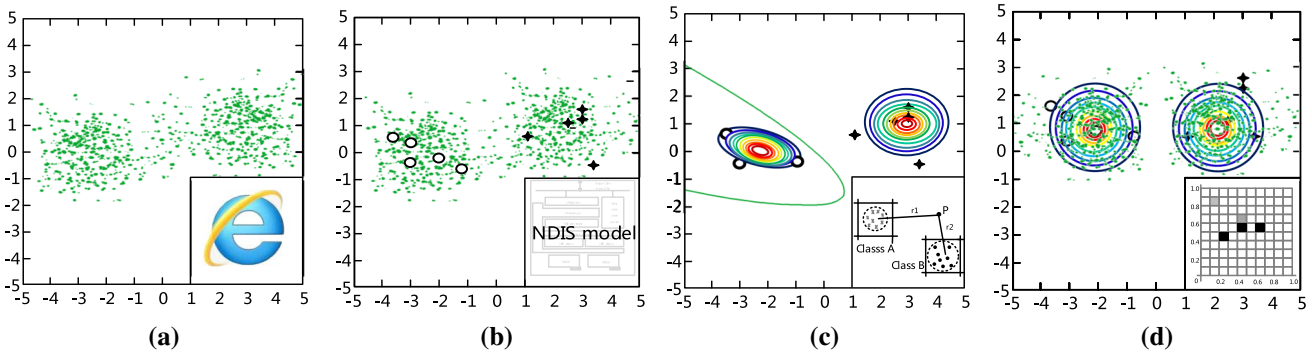


Fig. 1 Schematic description of the proposed Internet traffic framework

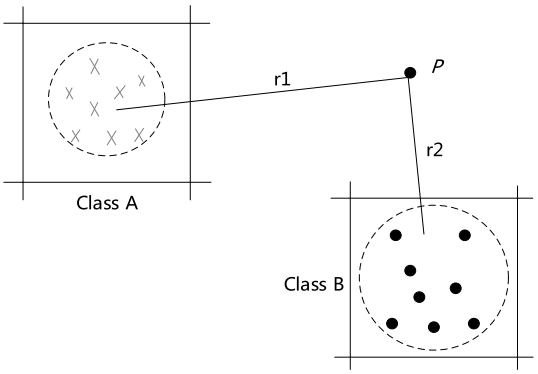


Fig. 2 Relation between data similarity and distance

data and the overall situation. For example, classical k-means method (MacQueen et al. 1967) is a data clustering method based on the distance in the center neighborhood. This kind of brush-fire clustering may fail to represent the entire precision. Nevertheless, most clustering methods utilize a kind of entitative feature between data, which is the similarity of data. For convenience, the "Euclid distance" (hereafter, "distance") compares the similarity of data with the real-world gravitation.

The gravitation between two objects as well as that between two data points is inversely proportional to the distance between them. As presented in Fig. 2, A and B are assumed as two kinds of data in the two-dimensional data space. P is a test data point, which belongs to the unknown data sort. However, the geometry center distance r_1 between P and A is bigger than r_2, which is between P and B. The degree of P for A is shorter than it for B.

3.1.3 Law of data gravitation

Based on the above analysis, the law of universal gravitation can be applied in data classification, and can be known as "the law of universal gravitation" in data space.

Definition 1 Data point is a data unit with "data quality" in a data space. Data point is composed of a group of data elements (point) with special relationship in a data space. Generally, this kind of relationship is a geometry border upon the relationship of the data element in the data space. A data point has two primary attributes, namely, data quality and data center of mass.

Definition 2 Data quality is the inclusive number of data elements in a data point.

Definition 3 Assuming that $x_1, x_2, \ldots, x_m$ ($x_i = \langle x_{i1}, x_{i2}, \ldots, x_{in}\rangle, i = 1, 2, \ldots, m$) are a group of data elements in the n-dimensional data space S; P is composed of $x_1, x_2, \ldots, x_m$, and thus the data centroid $x_0 = \langle x_{01}, x_{02}, \ldots, x_{0n}\rangle$ of P is the geometry center in data space, which is shown in Formula (3):

$$x_{0j} = \frac{\sum_{i=1}^{m} x_{ij}}{m}, \quad i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, n. \quad (3)$$

Given that a data particle has data mass and a data centroid, and a data particle is described by a paired expression $\langle m; x\rangle$, in which m is the data mass of the data particle and x is the data centroid. After considering the class information (feature y), a data particle is described as a triple expression $h\langle m; x; y\rangle$.

Definition 4 (*Atomic data particle*). An atomic data particle only contains one data element. The data mass of an atomic data particle is 1.

Definition 5 (*Data gravitation*). Data gravitation is defined as the similarity between data particles, and it is scalar. Data gravitation is an important factor that is different from the physical force. For the same data particle, the gravitation from different data classes can be compared. Meanwhile, the data gravitation from the same class follows the superposition principle.

Lemma 1 (Superposition principle). *Assuming that $p_1; p_2; \ldots; p_m$ are data points in a data space and they belong to the same data class. The gravitation they act on another data point is given by $F_1; F_2; \ldots; F_m$. The composition of the gravitation can be obtained by using Formula* (4).

$$F = \sum_{i=1}^{m} F_i \quad (4)$$

Definition 6 The data gravitation field is formed by a data point through the interaction of data gravitation with the field of data space as a whole.

Given that data gravitation may belong to different data sorts, similar kinds of gravitation fields produced by a data point are considered when discussing the data gravitation field. Field strength is a pivotal element of data gravitation field. The field strength of an appointed point is equal to the summation of field strengths produced at this point on all data points of the same kind. At this point, the field strength produced by a single data point is equal to the data gravitation of the atomic data point. Similar to the equal-force-cover in the physical field, the same points of all the field strengths in the data space form a hypersurface, which is called "equal-force-cover" in the data gravitation field.

The data gravitation intensity between two data points in the data space is a plus-rate with the product of their data quality, as indicated in Formula (5), which is the anti-rate with the square of the Euclid distance between them.

$$F = \frac{m_1 m_2}{r^2}, \quad (5)$$

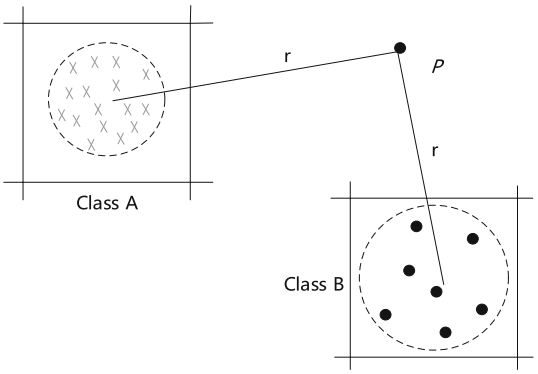


Fig. 3 Relation between data similarity and data density

where F is the data gravitation between two data points; m_1 and m_2 are the data qualities of data points 1 and 2, respectively; and r is the Euclid distance between two data points in the data space.

If the given problem has n-dimensional feature, on the assumption that the data centroid of data point 1 is $\langle x_{11}, x_{12}, \ldots, x_{1n}\rangle$, then the data centroid of data point 2 is $\langle x_{21}, x_{22}, \ldots, x_{2n}\rangle$. Thus, Formula (6) is as follows:

$$r = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2}. \tag{6}$$

If the quality of an object increases, then the gravitation of any other object likewise will increase. In the data space, one data sort includes more data points. The unknown degree of a data points belonging to this kind of class becomes stronger. In Fig. 3, A and B are two data sorts. A has 16 data points, whereas B has only 7 data points. The subordinate sample P is unknown, and the distance of P between A and B is r. Given that the number of data elements in A is greater than that of B, we consider the "quality" of A is also greater than B. Hence, we conclude that the intensity belonging to A is greater than that belonging to B.

3.2 Further division of recognition space

3.2.1 Definition of recognition space

This study defines recognition space as the classification space of identification effect in mapping obtained by using the clustering or classified method. Traditional classified recognition space is a stick segment of space [0C1]. In this study, recognition space is extended from being one-dimensional to two-dimensional, and from three-dimension to n-dimensional, which as shown in Fig. 4.

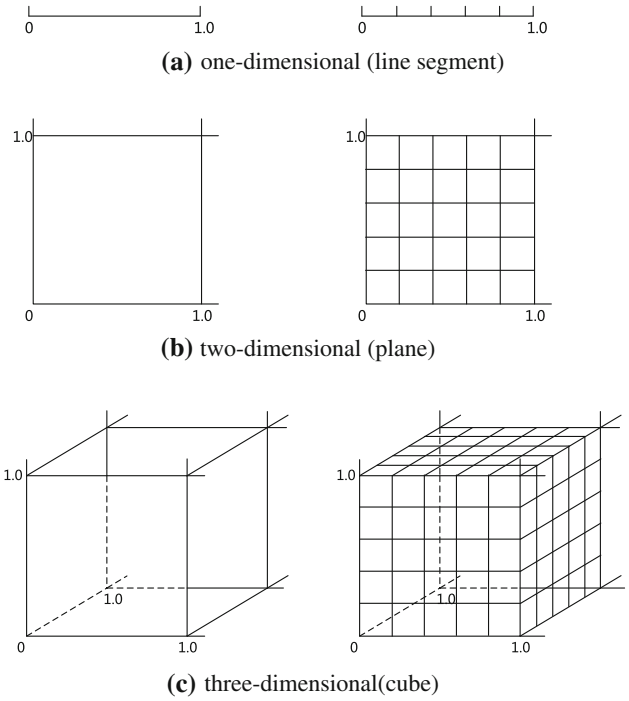


Fig. 4 Class recognition space and its further division

3.2.2 Further division of recognition space

In the new method of this study, recognition space is divided into several small divisions. Assuming that m expresses the number of dimensions in space. If m equals 1, the divisions become small line segments. If m equals 2, the recognition space is divided into rectangular divisions. Analogously, cube divisions are obtained from the recognition space when m equals 3, and so on. The total number of divisions can be obtained by using Formula (7):

$$\text{Total division number} = \prod_{d=1}^{m} \text{Division number}_d \tag{7}$$

in which the division number$_d$ is the number of divisions in the d axis, and the total division number is the number of divisions in the whole recognition space. Figure 4 presents the instances of recognition space with division regulation in different dimensions. On the assumption that the number of further division spaces in each dimension is $n = 10$, when $m = 1$, the total number of further division spaces is 10. When $m = 2$, the total number of further division spaces is 100. When $m = 3$, the total number of further division spaces is 1000. The number n of further division spaces depends on the precision requirement of the application. A greater precision corresponds to a larger n and the higher computation complexity.

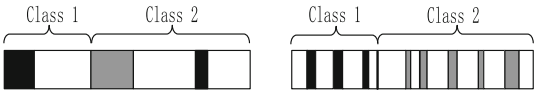


Fig. 5 Selection way of color points

3.2.3 Color recognition space

To map the sample into different sorts, various colors are used to dye the samples in the recognition space. A dyed sample recognition space can be used to classify an unknown test sample.

Choice of color point. Before being colored, all the sample data are mapped after being classified and exported or are directly mapped into the recognition space. During coloring, the samples in the dataset are mapped into the recognition space by a classifier. Therefore, the entire map of data samples can be selected to color the recognition space. Considering the generalization ability, only part of the samples taken out by rate DP ($0 < \mathrm{DP} < 1$) are used to color the divisions. These selected and mapped samples are called color points. After the DP was decided, there are two sequences being used to generate color points in one class, namely, order and random (Fig. 5). Order sequence suits the instance that the sample distribution changes little. Random sequence suits the instance when sample distribution is large.

Coloration rules. Each recognition space is blank after its divisions. The following rules are observed during coloring. R1: Different colors must be applied on different classes. R2: One or more divisions can be colored as one class. R3: If the color points of one class comprise the majority (weighted) of one division, this class will control the division, and the division should be colored according to the color of the related class. R4: After coloring the recognition space, the blank divisions are called "uncolored divisions". Normally, training samples do not fall into these divisions. However, if they do, the division will then be colored. The test points that fall into these divisions are called "unclassified points". The corresponding sample can then be considered unclassified.

Classified weight control However, the dataset is often uneven. Thus, the weight of the color points is an important factor for calculating the number of color points of each class. The category of the majority in one division should control the division which it belongs to. The process in which labels are divided according to the majority is called "division coloring". However, not every class data can color the recognition space with the same weight, because the number of each class data in the dataset is different. A class will control most of the recognition space if its number is considerably larger than any other classes in the same dataset. To solve this problem, a weight needs to be defined. The weight of the first class should be smaller than that of the second class if the number of the former is larger than the latter. The proportion of class c in the entire dataset is defined as R_c in Formula (8).

$$R_c = \frac{\mathrm{Num_c}}{\mathrm{Num_{total}}} \tag{8}$$

in which $\mathrm{Num_c}$ is the number of samples in class c, and $\mathrm{Num_{total}}$ is the total number of samples in the entire dataset. The weight of class c is calculated by using Formula (9).

$$W_c = \frac{1}{R_c} \tag{9}$$

Coloration. The course of coloration conforms to the above descriptive principial and corresponding weight for dying further division of recognition space. The coloration algorithm can be designed as Algorithm 1.

For example, in the sample collection of a clustering effect, assuming that the sample totality of classification 1 is twice as that of classification 2, and W_0 is equal to 3, and W_1 is equal to 1.5. Assuming that the dimension of the recognition space is 2, then the number of divided spaces in single dimension is 10, and the total number of fractionized spaces is 100. Figure 6 describes the division of space and color result.

Algorithm 1 Coloration algorithm

```
1: Majority = 0
2: for C=0 to Number of Classes-1 do
3:     if (number of color points in class c in this division)*(weight
   of class c)>(number of color points of the class majority in this
   division)*(weight of the class majority) then
4:         majority = C
5:     end if
6: end for
```

3.3 Semi-supervised learning classification

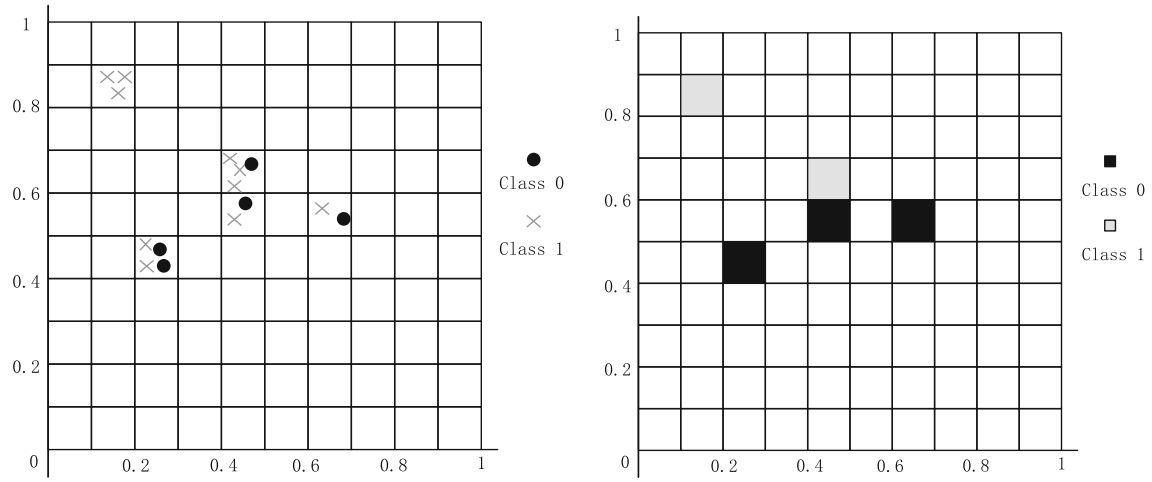
As presented in Fig. 7, in order to achieve the semi-supervised classification scheme which combines the unsupervised clustering and supervised classification method, a recognition space was created to support the data gravitation-based cluster and further division coloring-based classification. The detail shows as follow subsections.

3.3.1 Samples for cluster

The set of the original data samples is assumed to be $S = \{x_1, x_2, \ldots, x_m\}$. This set can calculate the gravitation field strength in the position that each sample takes over as follows:

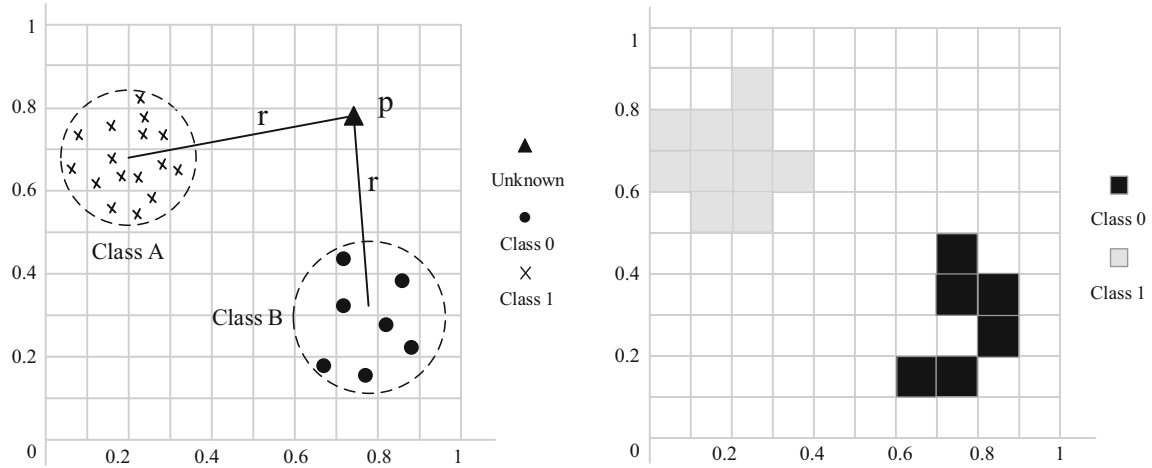
$$F_i = \sum_{j=1}^{m} f_j (j \neq i) \tag{10}$$

in which f_j is the gravitation from the jth data sample to sample i. This calculation method follows Formula (11) of data gravitation:



(a) Distribution of coloring point **(b)** Coloring result

Fig. 6 Coloring result in the recognition space



(a) Clustering **(b)** Classification

Fig. 7 Semi-supervised learning classification

$$f_j = \frac{1}{r_{ij}^2}. \tag{11}$$

The two samples can be considered as a data particle with a data quality of "1" in the course of the calculation. Therefore, the product of their quality of the samples is constantly equal to 1 in Formula (11).

Given the threshold $F_{\min}$ of gravitation field strength, sample set $S^{'} = \{x_{1'}, x_{2'}, \ldots, x_{n'}\}$ can be clustered as each x_i must meet $F_i > F_{\min}$.

3.3.2 Data gravitation-based clustering

The first step of clustering is the confirmation of kinds of samples that can be clustered. The course of clustering needs to solve two basic problems:

- Which kind of samples can be aggregated into one cluster?
- How many clusters can be obtained when the algorithm is converged?

A strictest criterion exists where the enclosure space can be considered as a cluster, which is encircled by the equipotential plane with field strength $F_{\min}$, based on the theory of gravitation field. However, the research and demarcation of the equipotential plane would consume more resources to calculate. Therefore, a rigorous equipotential plane cannot be found to cluster in practical application.

Each cluster is assumed to be an anti-concave in space. Two samples that can be clustered must abide by the following lemma.

Lemma 2 *If the cluster in space is anti-concave, the two samples that can be clustered must belong to different clusters if any one point exists on the connected-line of the samples, where the field strength is less than threshold $F_{\min}$ of the clustering.*

According to this lemma, an approximate validation method could be used to ensure whether any two data samples could aggregate to one cluster by measuring the field strength of the distances between the two samples in the first Quartile ($F_{ij1/4}$), the second Quartile ($F_{ij1/2}$), and the last Quartile ($F_{ij3/4}$).

The above analysis shows that the cluster algorithm can be designed as Algorithm 2:

Algorithm 2 Cluster algorithm

```
    select S' which is samples to cluster
 2: given the S1 as samples set in cluster, which is an empty set
    for i =1 to n do
 4:     for j =1 to n do
            if j = i then
 6:             Continue
            end if
 8:         compute the field strength F_ij1/4 in first Quartiles of the
    distance between sample i and sample j
            compute the field strength F_ij1/2 in second Quartiles of the
    distance between sample i and sample j
10:         compute the field strength F_ij3/4 in last Quartiles of the
    distance between sample i and sample j
            if F_ij1/4>=F_min and F_ij1/2>=F_min and F_ij3>=F_ij3/4 then
12:             cluster(j,i) //j and i are included to the same cluster
                add(S1,j)
14:             delete(S',j)
                count(S'=S'-1)
16:         end if
        end for
18: end for
```

3.3.3 Further divide recognition space-based learning

The input sample in the traditional classified method is mapped on recognition space to obtain the output. In this paper, an unclassified sample needs to be mapped into a patulous and subdivision recognition space based on the result of clustering. The nuclear content of the semi-supervised learning methodology obtains several clusters, including unclassified and classified samples, by clustering the test sample set that includes a part of the samples with a label. However, this kind of cluster could only have an unclassified sample. All the unknown samples are classified to the known categories in a cluster of samples with the known categories. All samples are classified according to the unknown categories in a cluster that only includes the unknown samples. This requirement necessitates special operation. Different clusters with a converged algorithm dye the corresponding sample space by applying the coloration theory. The course of coloration follows certain principia:

- If a cluster only includes one kind of sample with a known category, this cluster should be controlled by this category. All samples in this cluster belong to this category, and they are dyed with the corresponding color.
- If a cluster includes various samples with known categories, all samples in this cluster will be dyed based on the coloration principia (4). The category with the most weight is chosen to control the corresponding cluster, and the samples in this cluster are dyed based on their category.
- If a cluster does not include samples with known categories, the cluster is considered to be controlled by the new unknown application category. The samples are further classified through human intervention. A new application category is analyzed in the clusters. Therefore, the course of clustering requires the preparation of the known standard samples as much as possible.

3.3.4 Identification and classification

The dyed intervals of a converged cluster algorithm can be regarded as practical sample categories, which are dyed by subdividing spaces into the recognition space. The new data sample is input. The samples that are mapped into the dyed interval through interval mapping fall under the category with the corresponding color. The samples are not mapped into interval that is not dyed. It cannot be identified or it belongs to new categories. These intervals should be further confirmed by using other methods.

The above semi-supervised learning model obtained is suited to the identification of samples of no time sequence feature and the data sampling with some time sequence features. These samples could cluster in every interval based on the requirement of traffic identification to form a classifier. Subdividing the recognition space after clustering can be continued after the traffic identification. In the application of traffic identification, the clustering before 10 min can be used to identify the traffic, results after 10 min or half an hour. The result of traffic clustering between 8:00 and 9:00 in the first day is used to identify traffic after a week or more than a week later. In the course of traffic identification, the method can continually collect samples to analyze further clustering identification.

Table 1 Characteristics of Auckland II traces

Type	No. of instances	Total bytes
FTP	251	136,241
FTP-data	463	5,260,804
HTTP	23,721	139,421,961
Imap	193	86,455
POP3	498	98,699
Smtp	2602	1,230,528
Telnet	37	21,171

Table 2 Characteristics of UJN traces

Type	No. of instances	Total bytes (G)
HTTP	2,024,446	25.377
P2P	172,624	41.888
FTP	59,119	1.052
Email	587,423	0.431
Stream	121,794	3.718
Chat	339,587	0.19
Unknown	2,884,243	13.663

4 Experiment and evaluation

4.1 Auckland II traffic datasets

Auckland II is a collection of long GPS-synchronized traces taken using a pair of DAG 2 cards at the University of Auckland which is available at WAND (2009). There are 85 trace files which were captured from November 1999 to July 2000. Most traces were targeted at 24-h runs, but hardware failures have resulted in most traces being significantly shorter. We selected two trace files captured at Feb 14 2000 (20000214-185536-0.pcap and 20000214-185536-1.pcap) for our study. The traces include only the header bytes with a maximum amount of 64 bytes for each frame, while the application payload is fully removed. And all IP addresses anonymized use the Crypto-Pan AES encryption. The header traces were captured with a GPS-synchronized mechanism using a DAG3.2E card connected to a 100 Mbps Ethernet hub interconnecting the University's firewall to their border router.

Since the application payloads were not recorded in Auckland II, DPI tools are invalid to obtain ground truths. The only way to pick out the original application type is using port numbers. In this study, we only accounted TCP case since TCP is the predominant transport layer protocol. Each flow is thus assigned to the class identified by the server port. We selected eight main types from Auckland II traces and filtered mouse flows with no more than six non-zero packets. Table 1 lists all selected types and their instance and the total bytes distributions.

4.2 UJN traffic datasets

Classifying traffic aims to match the traffic with the application of generated traffic, which is described by the feature vector with different applications model (traffic classification). The network traffic is defined as a series of packets sequences that are transferred between the two ends by the source address, the source port, the objective address, the objective port, the transmission protocol and the finished time or other terminative flags for distinguishing various traffic. The traffic feature in this paper concerns mainly about the number of packets in the forward and backward direction, the average packet size in the forward and backward direction, and the variance of packet size in the forward and backward direction. In the interface of the mirror image, the corresponding label is examined after obtaining traffic. A real "sign" of traffic classification is also obtained.

The data set is collected in a laboratory network of University of Jinan (UJN) by using Traffic Labeler (TL) (Peng et al. 2014). TL system captures all user socket calls and their corresponding application process information in the user mode on a Windows host, and it sends the information to an intermediate NDIS driver in the kernel mode. The intermediate driver writes application type information on the TOS field of the IP packets which match the 5-tuple. By this means, each IP packet sent from the Windows host carries their application information. Therefore, traffic samples collected on the network have been labeled with the accurate application information and they can be used for training effective traffic classification models. We deployed 10 TL instances on Windows user hosts in the laboratory network of Provincial Key Laboratory for Network Based Intelligent Computing. A mirror port of the uplink port of the switch was set, and a data collector was deployed at the mirror port. The deployed TL instances ran at work hours every day. The data collecting process lasted 2 days in May 2015. Again, flows with no more than six non-zero payload packets are also filtered. And Table 2 shows the instance and the total bytes distributions of each type. The collected 86.259G Bytes datasets was divided into eight groups. Six kinds of application traffic exist in the experiment, web browser (HTTP), P2P (Maze, Web Thunder), the FTP, email, stream media traffic, and the instant chat traffic. The unknown data are some real unknown traffic, besides the above six kinds. Table 2 describes the flow number, the number of bytes, and the corresponding percentage from collecting the traffic samples.

The data of the traffic samples collected in the experiment are the original data that comprised all data packets. The machine learning method must be adopted to initialize this kind of data packet through traffic classification, and it must

Table 3 Features description in the experiment

No.	Features
01	Number of packets of a flow in forward direction
02	Number of bytes of a flow in forward direction
03	Number of bytes in header of packet of a flow in forward direction
04	Number of bytes in payload of packet of a flow in forward direction
05	Number of packets of a flow in backward direction
06	Number of bytes of a flow in backward direction
07	Number of bytes in header of packet of a flow in backward direction
08	Number of bytes in payload of packet of a flow in backward direction
09	Mean packet size of a flow in forward direction
10	Mean packet size of a flow in backward direction
11	Variance in packet size of a flow in forward direction
12	Variance in packet size of a flow in backward direction
13	During time of a flow
14	Inter-arrival time among packets of a flow
15	Total number bytes of a flow
16	Total number in packet header of a flow

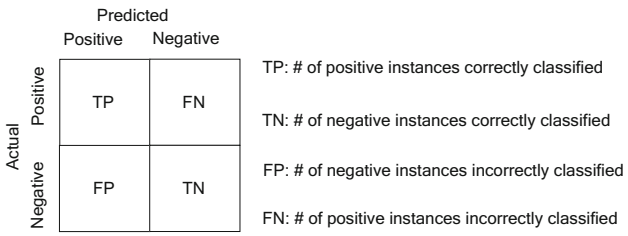


Fig. 8 Confusion matrix

also be corresponding to the character of the traffic statistic. The features of the traffic statistic and the initialized original data of collection are described in Table 3. All features are extracted from the header of packets in each flow. A group of sample collection with a corresponding traffic register of 6,189,236 is obtained. The succeeding experiment can use any quantitative sample for analysis.

4.3 Accuracy performance

The confusion matrix is the basis in measuring a classification task, wherein the rows denote the actual class of the instances and the columns denote the predicted class. Figure 8 shows a typical confusion matrix of a binary classification. The simplest method to evaluate a classifier is using the classification accuracy (acc) which is defined as the rate between the number of samples correctly classified and the total number of samples in testing set, but the accuracy can only express the overall level of hitting ratio, and it does not contain particular information of incorrectly classified sample ratio of each class. Therefore, a more sophisticated and widely used evaluating method is applied in this research. This method uses true positive rate (TPR) and false positive rate (FPR) to evaluate the performance of a classifier. TPR and FPR are deduced from the confusion matrix as Fig. 8 shows. For each class, a confusion matrix can be obtained according to the classification results. And then its TPR and FPR are defined as:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{12}$$

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{TN} + \mathrm{FP}}. \tag{13}$$

It can be seen from these two equations that TPR is in fact the ratio of the correctly classified positive samples and the total positive samples, and the FPR is the ratio of the incorrectly classified negative samples and the total negative samples. It can be inferred easily that:

$$\mathrm{acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}. \tag{14}$$

5 Semi-supervised classification and analysis

5.1 Classification based on labeled traffic

Based on open Auckland II dataset and collected UJN dataset, we adopted the method of 10 cross-validations for contrast and analysis. We randomly chose 50,000 sort samples with the "label" and randomly divided these samples into 10 symmetrical subclasses. In each experiment, we united nine subclasses for training and left one for classification testing. We obtained the exact ratio of average identification to be the

Table 4 Experiment results on UJN dataset–updated

APP	ANN		SVM		DGC		Semi-supervised	
	TPR(%)	FPR(%)	TPR(%)	FPR(%)	TPR(%)	FPR(%)	TPR(%)	FPR(%)
HTTP	87.19	2.18	90.82	21.07	92.15	8.34	92.05	1.58
P2P	95.11	1.89	97.69	18.12	97.47	10.27	98.01	2.48
FTP	95.73	3.39	96.10	28.01	96.62	7.52	95.58	3.89
Email	82.94	4.22	85.63	22.01	94.47	12.08	92.32	3.76
Stream	98.52	1.89	97.71	24.12	96.04	6.88	97.85	1.99
Chat	96.78	1.97	96.21	19.06	95.90	7.58	97.10	2.81

Table 5 Experiment results on Auckland II dataset–updated

APP	ANN		SVM		DGC		Semi-supervised	
	TPR(%)	FPR(%)	TPR(%)	FPR(%)	TPR(%)	FPR(%)	TPR(%)	FPR(%)
FTP	96.13	2.75	97.01	14.01	95.14	7.53	97.55	1.97
FTP-data	92.12	2.23	92.69	15.12	92.36	9.38	93.01	2.89
HTTP	88.12	3.73	91.10	18.24	92.62	7.57	92.81	1.97
Imap	91.09	1.89	93.63	22.49	93.98	10.47	92.89	3.91
POP3	91.15	2.59	92.72	24.71	94.34	6.86	95.77	1.99
Smtp	89.01	1.38	94.28	14.01	95.70	7.77	94.31	2.83
Telnet	93.12	3.11	95.21	12.52	95.15	7.00	98.71	3.85

estimating parameter of the effect from the experiment. Artificial neural network (ANN) (Yaghini et al. 2013; Prieto et al. 2013), Support vector machine (SVM) (Zhu et al. 2012; Shao et al. 2013), and data gravitation classifier (DGC) (Galperin 2011) were used for abundant contrast to perform the contrastive experiment. As result lists in Tables 4 and 5, our newly designed semi-supervised learning method has nice manifestation in classification accuracy performance with high TPR and low FPR. It shows greater performance contrast to ANN, SVM and DGC methods on both Auckland II and UJN datasets for the most part.

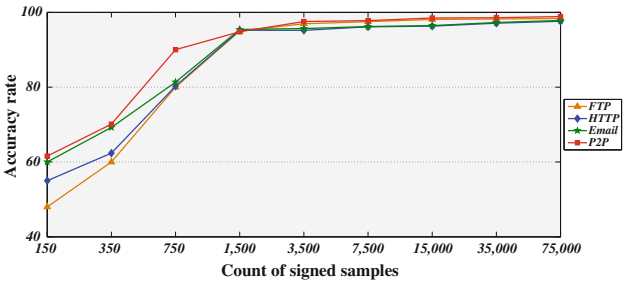


Fig. 9 Classification accuracy rate affected by labeled samples

5.2 Classification based on mixture traffic

Adopting the semi-supervised learning method adequately to utilize the knowledge of the data sample without "label" to improve the effect of the classification identification. We utilized the semi-supervised learning method based on data gravitation and the theory of further division recognition space to test and to analyze the same kind of traffic sample collection.

The requirement of the sample quantity with "label" is an important guideline in validating the semi-supervised learning method. The sample with "label" generally increases, and the accurate ratio of the identification is higher. The sample with "label" decreases, and the effect of the identification is worse. The effect of the experiment with the expression indicates that this classification method has a negative effect in the instance of owning a little sample with "label". The effect of classification was improved obviously. The effects of the experiment (Fig. 9), in which the 75,000 samples are the samples with "label", show that the accurate ratio of identification with the semi-supervised learning method has proximity to that of the accurate ratio of identification in the supervisory learning method. The influence from which the durative increasing in the number with the classified sample toward the semi-supervised learning method decreasing is most important. The experiment expresses that this method could obtain a better classification effect for a sample with minimal "label". An analogical classified effect in the instance of a sample with "label" could be gained in a large classification from having a supervised learning method. The advantage of the semi-supervised learning method is embodied in the results of the paper.

Table 6 Detection of unknown application

Cluster	Labeled class	Count of samples	Count of labeled samples	Accuracy (%)
Cluster 1	Stream	30,161	26,152	93.96
Cluster 2	Stream	72,828	70,743	
Cluster 3	Stream	21,295	19,884	
Cluster 4	Chat	95,345	92,256	94.92
Cluster 5	Chat	26,031	22,958	
Cluster 6	Unknown	219,314	#	#
Cluster 7	Unknown	1,008,146	#	#
Cluster 8	Unknown	98,222	#	#
Cluster 9	Unknown	247,391	#	#
Cluster 10	Unknown	16,677	#	#
Cluster 11	Unknown	7690	#	#

5.3 Unknown traffic sort discovery

Another important feature of the semi-supervised learning method is the ability of new sort discoveries. We labeled the stream and chat traffic, which is the sample without "label", for validation. The given sample of the semi-supervised learning method classification can be used to test the ability for unknown sort discovery. The data shown in Table 6 are the sample data of the classified "label" from the collection in the experiment. The "#" is expressed to the known or the unknown traffic and the effect of the identification from stream and chat. This system will validate the effect towards the discovery ability of the data sort in two kinds of data from stream and chat. Table 6 shows that the correct ratio of discovery from the two kinds of "unlabeled" and "fake" have reached a higher level, which shows that the method has an ideal discovery ability for new applications.

6 Conclusions and future work

The semi-supervised learning method provides new ideas in the supporting machine learning method, which lacks "labeled" samples. In this paper, we analyzed data gravitation and the further division of recognition space based on the semi-supervised learning method. The method is used for identification application in real Internet traffic identification. The experiment results show that the method is a powerful multi-classification tool that could aid in multi-application identification. The data gravitation may reveal the underlying data space structure from the unlabeled data, which is integrated into the classification to help train a better classifier. The experimental results on the real datasets demonstrate the advantages of the proposed method.

Acknowledgements This work was supported by the National Natural Science Foundation of China No. 60903176 and No. 61472164, the Natural Science Foundation of Shandong Province No. ZR2014JL042 and the Program for Youth science and technology star foundation of Jinan No. TNK1108.

Compliance with ethical standards

Conflict of interest None

References

Auld T, Moore AW, Gull SF (2007) Bayesian neural networks for internet traffic classification. IEEE Trans Neural Netw 18(1):223–239

Beitollahi H, Deconinck G (2014) Connectionscore: a statistical technique to resist application-layer ddos attacks. J Ambient Intell Hum Comput 5:425–442

Chen X, Zhang J, Xiang Y, Zhou W (2013) Traffic identification in semi-known network environment. In: Proceedings of the 2013 IEEE 16th international conference on computational science and engineering, IEEE, pp 572–579

Chen Z, Wang H, Abraham A, Grosan C, Yang B, Chen Y, Wang L (2009) Improving neural network classification using further division of recognition space. Int J Innov Comput Inf Control 5(2)

Chiou TW, Tsai SC, Lin YB (2014) Network security management with traffic pattern clustering. Soft Comput 18:1757–1770

Erman J, Arlitt M, Mahanti A (2006) Traffic classification using clustering algorithms. In: Proceedings of the 2006 SIGCOMM workshop on mining network data, ACM, pp 281–286

Erman J, Mahanti A, Arlitt M, Cohen I, Williamson C (2007a) Offline/realtime traffic classification using semi-supervised learning. Perform Eval 64(9):1194–1213

Erman J, Mahanti A, Arlitt M, Cohen I, Williamson C (2007b) Semi-supervised network traffic classification. ACM SIGMETRICS Perform Eval Rev 35:369–370

Esposito C, Ficco M, Palmieri F, Castiglione A (2015) Smart cloud storage service selection based on fuzzy logic, theory of evidence and game theory. IEEE Trans Comput. doi:10.1109/TC.2015.2389952

Este A, Gringoli F, Salgarelli L (2009) On the stability of the information carried by traffic flow features at the packet level. ACM Sigcomm Comput Commun Rev 39(3):13–18

Fbrega L, Jov T, Vil P, Marzo JL (2011) A network scheme for tcp elastic traffic with admission control using edge-to-edge per-aggregate measurements in class-based networks. J High Speed Netw 18:15–32

Ficco M, Palmieri F, Castiglione A (2015) Modeling security requirements for cloud-based system development. Concurr Comput Pract Exp 27:2107–2124

Galperin EA (2011) Information transmittal, relativity and gravitation. Comput Math Appl 61(6):1517–1535
Gan H, Sang N, Huang R, Tong X, Dan Z (2013) Using clustering analysis to improve semi-supervised classification. Neurocomputing 101:290–298
Gmez J, Gil C, Banos R, Mrquez AL, Montoya FG, Montoya MG (2013) A pareto-based multi-objective evolutionary algorithm for automatic rule generation in network intrusion detection systems. Soft Comput 17:255–263
Hrubeš P (2012) On the nonnegative rank of distance matrices. Inf Process Lett 112(11):457–461
Iliofotou M, Hc Kim, Faloutsos M, Mitzenmacher M, Pappu P, Varghese G (2011) Graption: a graph-based p2p traffic classification framework for the internet backbone. Comput Netw 55(8):1909–1920
Imai S, Leibnitz K, Murata M (2013) Energy efficient data caching for content dissemination networks. J High Speed Netw 19:215–235
Indulska M, Orlowska ME (2002) Gravity based spatial clustering. In: Proceedings of the 10th ACM international symposium on advances in geographic information systems, ACM, pp 125–130
Karagiannis T, Broido A, Faloutsos M, Claffy K (2004) Transport layer identification of p2p traffic. In: Imc '04 proceedings of ACM Sigcomm conference on internet measurement
Karagiannis T, Papagiannaki K, Faloutsos M (2005) Blinc: multilevel traffic classification in the dark. Proc ACM Sigcomm 35(4):229–240
Lakhina A, Crovella M, Diot C (2005) Mining anomalies using traffic feature distributions. ACM Sigcomm 35(4):217–228
Li H, Zhang T, Qiu R, Ma L (2012) Grammar-based semi-supervised incremental learning in automatic speech recognition and labeling. Energy Proc 17:1843–1849
Lin G, Xin Y, Niu X, Jiang H (2014) Network traffic classification based on semi-supervised clustering. J China Univ Posts Telecommun 17:1257–1270
Lu W, Rammidi G, Ghorbani AA (2011) Clustering botnet communication traffic based on n-gram feature selection. Comput Commun 34(3):502–514
MacQueen J et al (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, vol 1, Oakland, pp 281–297
McGregor A, Hall M, Lorier P, Brunskill J (2004) Flow clustering using machine learning techniques. In: Passive and active network measurement. Springer, Berlin, pp 205–214
Nguyen TTT, Armitage G (2008) A survey of techniques for internet traffic classification using machine learning. IEEE Commun Surveys Tutor 10(4):56–76
Ohzahata S, Hagiwara Y, Terada M, Kawashima K (2005) A traffic identification method and evaluations for a pure p2p application. Lecture Notes in Computer Science, pp 55–68
Peng L, Yang B, Chen Y, Abraham A (2009) Data gravitation based classification. Inf Sci 179(6):809–819
Peng L, Zhang H, Yang B, Chen Y, Wu T (2014) Traffic labeller: collecting internet traffic samples with accurate application information. China Commun 11:69–78
Prieto A, Atencia M, Sandoval F (2013) Advances in artificial neural networks and machine learning. Neurocomputing 121
Qian F, Gm Hu, Xm Yao (2008) Semi-supervised internet network traffic classification using a gaussian mixture model. AEU Int J Electron Commun 62(7):557–564
Roughan M, Sen S, Spatscheck O, Duffield N (2004) Class-of-service mapping for qos: a statistical signature-based approach to ip traffic classification. In: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement, ACM, pp 135–148
Shao YH, Wang Z, Chen WJ, Deng NY (2013) A regularization for the projection twin support vector machine. Knowl Based Syst 37:203–210
Shi L, Li W, Liu B (2010) Flow-based packet-mode load-balancing for parallel packet switches. J High Speed Netw 17:97–128
Shi Y, Zhang A (2004) A shrinking-based dimension reduction approach for multi-dimensional analysis. In: Proceedings of 16th international conference on scientific and statistical database management, IEEE, pp 427–428
Tur G, Hakkani-Tür D, Schapire RE (2005) Combining active and semi-supervised learning for spoken language understanding. Speech Commun 45(2):171–186
Upadhyaya SR (2013) Parallel approaches to machine learningła comprehensive survey. J Parallel Distrib Comput 73(3):284C292
WAND (2009) Wits: Waikato internet traffic storage. http://www.wand.net.nz/wits
Yaghini M, Khoshraftar MM, Fallahi M (2013) A hybrid algorithm for artificial neural network training. Eng Appl Artif Intell 26(1):293–301
Yan Y, Chen L, Tjhi WC (2013) Fuzzy semi-supervised co-clustering for text documents. Fuzzy Sets Syst 215:74–89
Ye W, Kyungsan C (2014) Hybrid p2p traffic classification with heuristic rules and machine learning. Soft Comput 18:1815–1827
Yu D, Varadarajan B, Deng L, Acero A (2010) Active learning and semi-supervised learning for speech recognition: a unified framework using the global entropy reduction maximization criterion. Comput Speech Lang 24(3):433–444
Zander S, Nguyen T, Armitage G (2005a) Automated traffic classification and application identification using machine learning. In: The IEEE conference on local computer networks, 30th anniversary, IEEE, pp 250–257
Zander S, Nguyen T, Armitage G (2005b) Self-learning ip traffic classification based on statistical flow characteristics. In: Passive and active network measurement. Springer, Berlin, pp 325–328
Zhang J, Xiang Y, Zhou W, Wang Y (2013) Unsupervised traffic classification using flow statistical properties and ip packet payload. J Comput Syst Sci 79(5):573–585
Zhang J, Chen X, Xiang Y, Wu J (2014) Robust network traffic classification. IEEE/ACM Trans Netw 24:84–88
Zhu Z, Zhu X, Guo Y, Ye Y, Xue X (2012) Inverse matrix-free incremental proximal support vector machine. Decis Support Syst 53(3):395–405