Mobile Networks and Applications (2018) 23:789–796
https://doi.org/10.1007/s11036-018-0999-x

Internet Traffic Classification Based on Incremental Support Vector Machines

Guanglu Sun[1,2] · Teng Chen[1] · Yangyang Su[1] · Chenglong Li[3]

Published online: 10 February 2018

Abstract
Machine learning methods have been deployed widely in Internet traffic classification, which identify encrypted traffic and proprietary protocols effectively based on statistical features of traffic flows. Among these methods, support vector machines (SVMs) have attracted increasing attention as it achieves the state of art performance in traffic classification compared with other machine learning methods. However, traditional SVMs-based traffic classifier also has its limitations in real application: high training complexity and computation cost on both memory and CPU, which leads to the frequent and timely updating of traffic classifier being impractical**.** In this paper, incremental SVMs (ISVM) model is first introduced to reduce the high training cost of memory and CPU, and realize traffic classifier's high-frequency and quick updates Besides, a modified version of ISVM model with attenuation factor, called AISVM, is further proposed to utilize valuable information in the previous training data sets. The experimental results have proved the effectiveness of ISVM and AISVM models in traffic classification.

Keywords Internet traffic classification · Incremental learning · Support vector machines · Attenuation factor

1 Introduction

Internet traffic classification has attracted lots of research interests in the recent years. The ability to identify flows and their relevant protocols is required for many applications, such as security and Quality of Service (QoS). The traditional methods of traffic classification are based on well-known port numbers and deep packets identifications [1]. However, these methods became ineffective when dealing with dynamic port numbers, encrypted payloads, unknown protocols, and even variants of known protocols.

Later on, machine learning methods were introduced to tackle the above problems, which mainly exploited the network behaviors and statistical characteristics of traffic flows [2, 3]. Many supervised machine learning models have been applied in traffic classification, such as Naive Bayes (NB) [4], SVMs [5, 6], Bayesian networks [7], k-nearest neighbor (k-NN) [8], C4.5 decision tree [9] and neural networks [10], etc. Among these methods, SVMs have attracted considerable interests for its high accuracy. Proved by the experiments, SVMs obtain an average accuracy over 95%, 2.3% higher than the best performance of other machine learning methods on the same data sets [5]. In this paper, SVMs are adopted as the baseline method in traffic classification.

Although SVMs are able to achieve satisfactory performance, it has two main limitations in practice in traffic classification task.

(1) **High training cost on both memory and CPU**: As SVMs have a high training complexity, the training of SVMs itself is time-consuming, especially in dealing with noisy data [11]. In addition, the more statistical features and training data adopted to achieve higher accuracy, the more occupations of memory and CPU resources needed.
(2) **Hardly realizing the frequent and timely updating:** In real scenario of traffic classification, when adding the new training data, the entire traffic classifier has to be retrained in order to adapt to the new change by combining the old training data with the new training data. This updating method is costly and time-consuming which causes the frequent and timely updating of traffic classifier to be impractical. But in real application, the timely updating of the model is critical for traffic classification.

✉ Chenglong Li
lichenglong@cert.org.cn

[1] School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China

[2] Research Center of Information Security & Intelligent Technology, Harbin University of Science and Technology, Harbin 150080, China

[3] National Computer Network Emergency Response Technical Team/ Coordination Center of China (CNCERT/CC), Beijing 100029, China

In order to tackle these problems, ISVM model is firstly introduced in traffic classification, which realizes the incremental updating of the classification model by utilizing the new training data set and the Support Vectors (SVs) produced in the last training process. The thought of incremental learning based on SVMs was first presented by Syed et al. in 1999 [11]. The method is known as the SV-incremental model in some situations. The application of ISVM model is due to the following reasons: first, SVs are the only parameters in SVMs model that the resulting decision depends on. That is, results got by training SVMs model with SVs are the same as those with the whole training data set. Therefore, an incremental result is equivalent to non-incremental result, when all the SVs in each non-incremental training case are included.

When being used in practice, however, ISVM model also has its limitations. As it only utilizes the new training data set and the SVs produced in the last updating process, other previous training data will be discarded directly. It could result in the loss of potential valuable information contained in the previous training data, which plays an important role in determining the classification boundary. AISVM is proposed for traffic classification to address this issue. The key point of AISVM model is to give per SV a weight value. If a training sample is SV in the current updating process, but not SV in the next updating process, it will not be discarded directly. Its weight value will be attenuated instead, until the weight value declines below a certain value. Then, the sample will be discarded from the set of the SVs.

The major contributions of this paper include:

- ISVM model is firstly applied to traffic classification, and the experimental results proved its effectiveness in classifying traffic flows.
- The high training cost of memory and CPU is dramatically reduced by deploying ISVM model, in order to realize traffic classifier's high-frequency and quick updates.
- AISVM model is proposed to overcome the drawback of ISVM itself in real application of traffic classification, which further improves the traffic classification accuracy without the updating cost increasing.

The remainder of this paper is organized as follows. Section 2 reviews related work. In Section 3, ISVM model is introduced for traffic classification and AISVM model is further proposed. Section 4 discusses the experimental results. Section 5 concludes the paper and presents the future work.

2 Related work

Internet traffic classification is a task of classifying the network flows which are a mixture of various flows with different application protocols. Traditional technologies of traffic classification mainly incorporate two categories: port-based method and payload-based method [1].

In port-based method, TCP or UDP port numbers are used to inspect and identify the application protocols according to the Internet Assigned Numbers Authority (IANA) list of the registered ports or well-known ports. It is simple and fast. But it became less effective as more new applications (such as P2P) started implementing dynamic port numbers in order to hide their identity [12]. Later on, payload-based methods were developed to inspect packet payloads based on specific string patterns of known applications. Though this method is more accurate, it does not work on encrypted traffic, and causes serious privacy and legal concerns.

More recently, machine learning methods have been applied in many fields [13, 14]. They also have received more attention in traffic classification [15]. Using the statistical features of traffic flows, machine learning techniques are able to avoid the above problems of port-based and payload-based methods. Machine learning models used in traffic classification are divided into two branches: unsupervised learning and supervised learning methods [3]. The former clusters traffic flows into groups that have similar traffic characteristics, while the latter classifies the traffic flow into a predefined category. The drawback of the unsupervised traffic classification is that without the real traffic classes, it is difficult to build an application-oriented traffic classifier by only using the clustering results. In contrast, the supervised learning methods require priori knowledge (also known as pre-labeled data) to train the classification model and the corresponding parameters. Then the trained model can be used in traffic classification [16]. This paper mainly focuses on supervised machine learning models for their popularity and wide usage.

They also have been received more attention in traffic classification. In 2005, Moore and Zuev [17] applied NB model to classify application protocols. In 2006, Williams et al. [18] compared five supervised learning models including NB with discretization, NB with kernel density estimation (NBKDE), C4.5 decision tree, Bayesian network and Naive Bayes tree, from the aspects of classification accuracy and computational performance. In 2009, Este et al. [5] applied SVMs on three well-known data sets and obtained an average accuracy over 95%, which is 2.3% higher than the best performance of other machine learning methods on the same data sets. In 2011, Finamore et al. [19] further presented statistical characterization of payload as features used in traffic classification based on SVMs. In 2012, Nguyen et al. [20] trained the machine learning models with a set of sub-flows and investigated different sub-flow selection strategies. In 2014, Ye and Cho [21] proposed an improved two-step hybrid P2P traffic classification framework with heuristic rules and REPTree model. In 2015, Li et al. [22] utilized logistic regression model to classify the traffic flows with the non-convex multi-task feature selection method. Peng et al. [23] verified that 5-7 packets are

the best packet numbers for early stage traffic classification based on 11 well-known supervised learning models.

3 Traffic classification based on incremental SVMs model

3.1 Problem setting

We first discuss how to transform traffic classification into a classification problem. Consider a set of traffic flows $\{x_1, x_2, \ldots, x_n\}$, each flow $x_i \in R^d$, in which d dimension of features correspond to d attributes of the traffic flow, such as packet size, TCP window size, etc. Each x_i is tagged as one of the application protocols $\{y_1, y_2, \ldots, y_m\}$, such as P2P, WWW, and FTP, etc. By training with the mapping pairs $<x_i, y_j>$ in the training data set, the goal is finding a discriminative function $y = f(x)$, which can make accurate prediction about what application protocol the unlabeled traffic flow belongs to. In this paper, we start with the simple binary classification problem. As the application protocols usually contains more than two types of protocols, the one-against-all strategy is utilized to expand the binary SVMs model into multi-class SVMs model for traffic classification.

3.2 Traditional SVMs model

SVMs are first introduced in a binary classification task with batch learning setting, assuming the training data and their labels are given as follows: $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}, x_i \in R^d, y_i \in \{+1, -1\}$.

SVMs separate the training samples by maximizing the margin between the SVs and the classification hyperplane. The hyperplane is defined by the equation: $w \cdot x + b = 0$, where w is a coefficient vector, b is a scalar offset, and the symbol "$\cdot$" denotes the inner product in R^d, defined as:

$$f(x) = w \cdot x = \sum_{i=1}^{n} w_i x_i \tag{1}$$

Samples lying on each side of the hyperplane are labeled as −1 or 1, respectively. The training samples falling on the margin of classification hyperplane are called SVs.

Through the Mercer kernel function $K(x_j, x_k) = \Phi(x_j) \cdot \Phi(x_k)$, e.g. linear, polynomial and RBF kernel, SVMs map the original training samples in space X to a higher dimensional space F to make them be separated. The new discriminative function is:

$$f(x) = sign(w \cdot \Phi(x) + b) \tag{2}$$

where $w = \sum_{i=1}^{n} \alpha_i y_i \Phi(x_i)$, $b = -1/2 \left(\sum_{x_a, x_b \in \{x_i\}} \sum_{i=1}^{n} \alpha_i y_i \Phi(x_a) \Phi(x_b) \right)$, $\Phi(x)$ is the mapping function.

SVMs optimize the coefficients w and b by applying the sequential minimal optimization algorithm, which is the direct reason why the training cost of SVMs model is so high. However, not all the samples but SVs (whose coefficients are not equal to zero) decide the classification hyperplane and the discriminative function. SVs can absolutely replace the whole training samples to present the characteristics of the data distribution, when kernel function and other coefficients are confirmed [24]. That is why ISVM model could get an incremental result that is equal to the non-incremental result, by utilizing the new training data set and the SVs produced in the last updating process.

3.3 Incremental SVMs model for traffic classification

As the new training data occur, the data distribution also changes with the time going. The classification model is required to be updated, in order to fit the new data and to make the accurate prediction. The traditional updating method of SVMs-based traffic classifier is to retrain the model by combining the old training data with the new one. The main drawback of this strategy is the cost of the model updating is too high, both in the occupation of updating time and computational resources. In this paper, ISVM model is implemented to tackle this problem, which realizes the incremental updating of the traffic classifier.

ISVM model discards the original training data, and retains the SVs which are produced in the last updating process of SVMs model [25]. When a new training data join, ISVM classifier trains a new discriminative function by combining the new data with the retained SVs [26, 27]. The updating procedure will be repeated, when an additional new data join. Figure 1 shows the primary updating steps of ISVM model.

3.4 ISVM model with attenuation factor for traffic classification

Although ISVM model could accelerate the speed of model updating, it decreases the traffic classification accuracy. In the original setting of ISVM model, if the SVs in the last updating

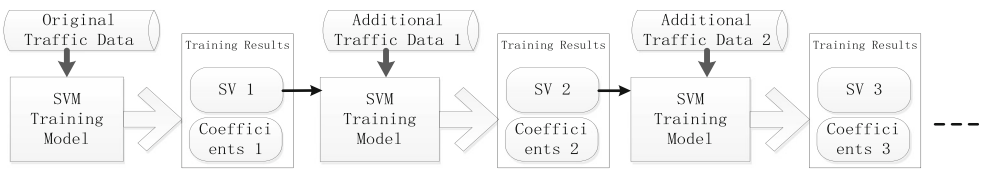

Fig. 1 The sketch map of ISVM updating steps

process cannot maintain its status and become the member of Not-SVs, it will be discarded directly in the next updating process. From the global perspective, this discarding strategy will result in the loss of potential valuable information contained in the previous SVs. Further, these SVs should play a critical role in determining the classification hyperplane continuously.

AISVM model gives each SV a weight value. In the continuous updating process of the traffic classifier, AISVM model retains all the previous SVs until its weight value is reduced below a certain value. Compared with the simply strategy of direct discarding, this method tries best to make use of the previous training samples. The algorithm is as the following:

Algorithm1 Incremental SVM algorithm with attention factor

Input: M_{i-1} // the AISVM model
RSV_{i-1} // the remaining set of the support vectors
$B_{i-1}=\{\beta_{RSV_{i-1}}\}$ // the set of weights
NTS_i // new additional training set

Output: M_i // the AISVM model updated with NTS_i
SV_i // new set of the support vector
RSV_i // the remaining set of the support vectors
$B_i=\{\beta_{RSV_i}\}$ // the set of the weights

1: if $M_{i-1} = null$
2: training M_{i-1} with NTS_i, generating SV_i // initialization
3: $M_i = M_{i-1}$, $RSV_i = SV_i$
4: for each $x_j \in RSV_i$
5: set $\beta_{x_j} = 1$, insert β_{x_j} into B_i
6: end
7: else
8: updating M_{i-1} with $(NTS_i + RSV_{i-1})$, generating SV_i
9: $M_i = M_{i-1}$
10: for each $x_j \in RSV_{i-1}$
11: if $x_j \in SV_i$
12: reset $\beta_{x_j} = 1$, delete x_j from SV_i
13: else
12: $\beta_{x_j} = \beta_{x_j} - \alpha^*$ // α^* is the attenuation factors
13: if $\beta_{x_i} < 0$
14: delete x_j from RSV_{i-1}, delete β_{x_i} from B_{i-1}
15: $RSV_i = RSV_{i-1} + SV_i$, $B_i = B_{i-1}$
16: for each $x_t \in SV_i$
17: set $\beta_{x_t} = 1$, insert β_{x_t} into B_i
18: end

From the algorithm, the initial SVMs model is trained in Step1 to Step6, when the first training data occurs. All the SVs are assigned with 1. From Step7, the updating process of SVMs model is described. The new model is trained by the new data and the remaining SVs. The new SVs are generated after the model updating in Step8. In Step10 to Step 15, the set of the remaining SVs is updated. In Step16 to Step17, the value set of the remaining SVs is updated.

4 Experimental results and discussions

4.1 Experimental data sets, metrics and setting

(1) Data sets

The traffic data set of Cambridge's Nprobe Project is used to train and test SVMs in this paper. The data set has been widely deployed in traffic classification such as the experiments based on Bayesian methods by Moore and Zuev [17]. It provides a wide variety of features to characterize traffic flows, which includes simple statistics about packet length, inter-packet timing, and information derived from traffic flows [28]. The total data set contains ten subsets, in which per subset has 256 attributes (seen in Table 1). In all the experiments of this paper, Entry1 data set is adopted as the training data, and Entry2 to Entry10 data sets as the test data.

(2) Evaluation metrics

The main metric used to evaluate the performance of traffic classification is accuracy value, which is the percentage that the number of traffic flows correctly classified accounts for of the total number of traffic flows. The accuracy values are obtained for the whole system instead of per class. Besides, the training time in each updating process is taken into account.

(3) Experimental setting

Table 1 The statistics of Cambridge traffic data set

Cambridge data set	Data duration	Samples number
1	1821.8 s	24,863
2	1696.7 s	23,801
3	1724.1 s	22,932
4	1784.1 s	22,285
5	1794.9 s	21,648
6	1658.5 s	19,384
7	1739.2 s	55,835
8	1665.9 s	55,494
9	1664.5 s	66,248
10	1613.4 s	65,036

Table 2 The statistics of each training part of Entry1

Training data set	WWW	MAIL	FTP-CONTROL	FTP-PASV	ATTACK	P2P	DATA-BASE	FTP-DATA	MULTI MEDIA	SERVICES	INTER ACTIVE	Total number
Part01	1006	1480	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	2486
Part02	2486	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	2486
Part03	2486	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	2486
Part04	1694	792	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	2486
Part05	1336	22	3	3	82	160	34	846	N/A	N/A	N/A	2486
Part06	1709	231	N/A	N/A	5	116	N/A	425	N/A	N/A	N/A	2486
Part07	1915	369	62	N/A	6	3	43	7	81	N/A	N/A	2486
Part08	1665	494	17	9	2	3	63	35	2	195	1	2486
Part09	1830	461	25	16	22	30	88	4	N/A	8	2	2486
part 10	2084	297	42	15	5	27	10	2	4	3	N/A	2489

In order to verify the fact that the traditional SVMs achieve the best performance in traffic classification, a comparative experiment is firstly conducted which adopts NB and NBKDE as the comparison methods. Then, the traditional SVMs model is directly deployed in incremental learning setting. To observe the variation trend of the accuracy value, training time and the SVs number during the incremental updating process of traffic classifier, Entry1 data set is divided into ten parts by the sequence of generating time. The statistics of each training part is listed in Table 2. N/A means no sample of this application protocol occurs. Then, ten parts of Entry1 data sets are added to the training process one by one. ISVM model and AISVM are further utilized in traffic classification. The division of training data set and the adding strategy of training data are the same as the above experiments.

4.2 Experimental results and discussions

(1) The batch learning based on the traditional SVM model

From Table 3, the performance of Naive Bayes model is much lower than the other two models. Besides, although the NBKDE model achieves a fairly high accuracy, it is impractical in real application because its classification process is very time-consuming. For example, it spends more than thirty minutes in identifying the flows in Entry9 data set, while SVMs only use 69.364 s. Considering the average accuracy, the SVMs model also achieves the accuracy over 2.3% higher than that of the NBKDE model. Therefore in this paper, SVMs are adopted as the baseline model in traffic classification.

(2) The incremental learning results based on the traditional SVMs model

Table 4 presents the experimental results of incremental learning based on the traditional SVMs model by progressively increasing the training data. The result in the 10th line is the point that all parts of the Entry1 are added to the training process. The Entry1 column belongs to the close test, as Entry1 itself is training data set. The other columns are the results of the open test. The Average column is calculating the average accuracy value from Entry2 to Entry10 column.

From Table 4, the accuracy value is not always increasing along with the increasing size of training data. The performance of the SVMs model does not absolutely depend on the size of training data, but on the number of SVs [11]. In general, the more SVs are, the higher classification accuracy that the SVMs model can achieve. From the global sense, the performance of traffic classification based on SVMs will increase with the size of the training data set growing.

However, with the increasing of the training data, the computing complexity and the occupation of the computing resources will also grow significantly, which posts a great challenge to the frequent and timely updating of the model in

Table 3 The batch learning results based on NB, NBKDE and SVMs

ML model	Training data set	Test data sets									
		Entry2	Entry3	Entry4	Entry5	Entry6	Entry7	Entry8	Entry9	Entry10	Average
NB	Entry1	57.5%	57.7%	51.9%	57.3%	57.8%	46.8%	50.5%	44.4%	45.7%	49.7%
NBKDE	Entry1	92.5%	96.5%	96.3%	96.8%	96.3%	94.3%	94.3%	92.1%	90.7%	93.6%
SVMs	Entry1	94.4%	96.6%	97.9%	96.4%	98.0%	97.8%	97.7%	96.2%	91.5%	95.9%

Table 4 The incremental learning results based on the traditional SVMs

	Train time H:M:S	SV number	Entry1	Entry2	Entry3	Entry4	Entry5	Entry6	Entry7	Entry8	Entry9	Entry10	Average
1	0:11:36	20	87.8%	78.0%	72.1%	83.7%	91.7%	79.3%	81.4%	84.3%	79.8%	89.8%	82.9%
2	0:26:06	86	88.1%	77.9%	71.7%	81.9%	90.7%	83.0%	84.1%	77.7%	73.6%	90.5%	81.3%
3	0:40:02	86	87.8%	79.3%	74.3%	83.2%	91.6%	84.8%	87.5%	84.5%	80.2%	90.9%	84.8%
4	1:10:09	130	88.6%	74.7%	70.2%	80.4%	87.2%	78.7%	89.8%	83.2%	80.3%	88.5%	83.1%
5	4:31:29	263	94.8%	76.7%	77.7%	73.8%	92.2%	62.6%	94.6%	74.2%	72.9%	90.3%	81.0%
6	8:33:34	363	97.0%	78.0%	78.2%	77.5%	84.0%	71.9%	94.8%	76.7%	74.0%	90.1%	81.9%
7	19:48:24	540	98.1%	88.8%	90.6%	93.3%	93.3%	79.3%	95.9%	89.0%	87.6%	89.0%	89.9%
8	39:13:48	657	99.2%	83.6%	89.0%	89.8%	92.1%	84.9%	91.4%	77.8%	79.0%	86.2%	84.9%
9	62:57:09	887	99.6%	77.2%	93.7%	86.2%	93.5%	91.6%	97.2%	72.9%	75.7%	63.5%	80.3%
10	86:44:51	1071	99.8%	94.4%	96.6%	97.9%	96.4%	98.0%	97.8%	97.7%	96.1%	91.5%	96.0%

Table 5 The incremental learning results based on ISVM model

	Train time H:M:S	SV number	Entry 1	Entry2	Entry3	Entry4	Entry5	Entry6	Entry7	Entry8	Entry9	Entry10	Average
1	0:11:40	20	87.8%	78.0%	72.1%	83.7%	91.7%	79.3%	81.4%	84.3%	79.8%	89.8%	82.9%
2	+ 0:01:23	21	88.2%	71.9%	67.4%	76.8%	86.8%	78.5%	78.3%	72.1%	68.0%	81.3%	75.2%
3	+ 0:01:20	20	87.8%	74.0%	69.4%	79.6%	90.4%	77.4%	81.0%	81.2%	76.7%	87.2%	80.4%
4	+ 0:06:13	41	88.8%	82.8%	80.9%	90.4%	92.1%	84.0%	90.0%	85.2%	83.9%	91.2%	87.7%
5	+ 0:26:16	223	91.2%	85.2%	87.3%	90.6%	91.7%	88.4%	92.1%	91.8%	88.9%	87.7%	89.5%
6	+ 0:34:19	299	96.5%	87.4%	91.4%	92.%	94.0%	82.4%	94.2%	93.1%	87.4%	84.5%	89.5%
7	+ 0:55:43	526	97.0%	90.0%	96.1%	96.0%	93.6%	60.3%	95.4%	95.4%	91.9%	88.5%	91.2%
8	+ 1:07:56	560	99.0%	90.4%	95.3%	96.9%	97.3%	94.0%	96.3%	94.8%	92.0%	92.7%	94.1%
9	+ 1:16:09	711	98.0%	91.7%	95.9%	97.8%	97.1%	95.0%	97.3%	96.2%	93.8%	92.5%	95.0%
10	+ 1:25:25	768	99.4%	91.3%	96.9%	97.8%	98.0%	97.9%	97.7%	94.2%	91.3%	94.5%	94.2%

traffic classification. For example, in the 10th line, when the size of training data set grows to 24,863, its corresponding training time is 86 h, 44 min and 51 s, which reaches the peak value of training time during the entire incremental training process.

(3) The incremental learning results based on ISVM models

From the experiments above, it is confirmed that it is hard to deploy the traditional SVMs model in incremental learning setting, because it leads to a long updating period and high occupation of memory and CPU. So ISVM model is applied to tackle this problem. Table 5 lists the classification results based on ISVM model.

Table 6 The incremental learning results based on AISVM model

	Train time H:M:S	SV number	Entry 1	Entry2	Entry3	Entry4	Entry5	Entry6	Entry7	Entry8	Entry9	Entry10	Average
1	0:11:34	20	87.8%	778.0%	72.1%	83.7%	91.7%	79.3%	81.4%	84.3%	79.8%	89.8%	82.9%
2	+0:01:25	21	88.2%	71.9%	67.4%	76.8%	86.8%	78.5%	78.3%	72.1%	68.0%	81.3%	75.2%
3	+0:01:23	21	88.2%	71.9%	67.4%	76.8%	86.8%	78.5%	78.3%	72.1%	68.0%	81.2%	75.2%
4	+0:06:16	33	89.0%	78.6%	70.5%	83.3%	89.5%	72.2%	91.2%	81.8%	80.1%	89.8%	83.5%
5	+0:26:03	227	91.4%	86.9%	86.3%	90.9%	91.0%	88.7%	92.1%	92.5%	89.9%	87.3%	89.8%
6	+0:34:51	289	96.6%	74.1%	80.0%	76.6%	88.9%	69.8%	93.4%	77.2%	73.0%	83.0%	80.3%
7	+0:58:05	456	97.6%	89.7%	95.1%	95.0%	95.0%	69.9%	95.6%	94.8%	89.5%	86.6%	90.7%
8	+1:09:59	588	99.1%	89.4%	95.5%	97.0%	96.7%	95.9%	91.0%	86.4%	85.8%	86.8%	89.7%
9	+1:27:57	735	99.3%	91.3%	96.0%	97.2%	98.0%	97.9%	97.6%	95.5%	93.9%	92.9%	95.1%
10	+1:38:12	901	99.6%	91.4%	96.2%	97.7%	97.7%	98.0%	97.8%	95.9%	94.6%	92.5%	95.4%

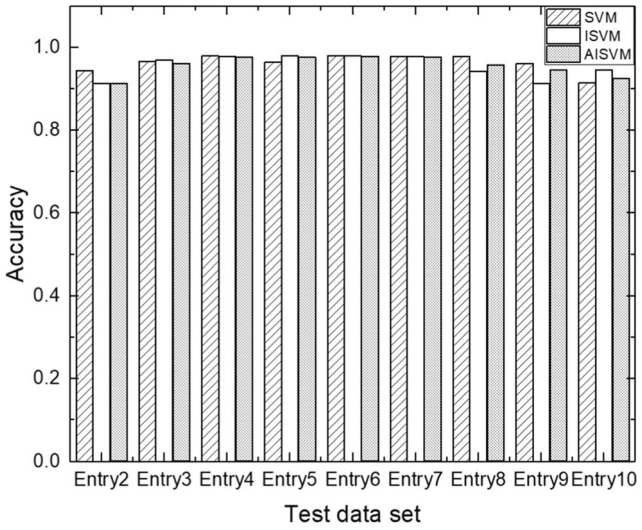

Fig. 2 The final results based on the three Models

In the continual process of model updating, the training time are reduced dramatically in each step compared with the traditional SVMs model in the same incremental learning pattern, which powerfully proves the effectiveness of ISVM model in realizing the frequent and timely updating of the traffic classifier. But in the 10th line which is the end step of incremental updating process, the average accuracy value is over 1.5% lower than the traditional SVMs in the same phrase. The decreasing of the traffic classification accuracy is due to that ISVM model ignores the previous data except the SVs produced in the last updating process. This method could result in the loss of potential valuable information contained in all the previous training data.

(4) The incremental learning results based on AISVM models

ISVM model only retains the SVs in each updating process, resulting in the loss of much information. AISVM model is proposed to tackle this problem, in order to make use of more information in previous training data. Table 6 shows the results based on AISVM model with all data sets. The parameter β is set as 1, the attenuation factor as 0.4. That is, if a training sample is not selected as SV in three continual updating steps, it will be discarded. Compared with ISVM model, AISVM model improves the classification accuracy by 1.2 percentages without the obvious increment of updating time.

(5) Comparison with three models

To intuitively show the effectiveness of ISVM and AISVM models in traffic classification, Fig. 2 shows the accuracies of the final results with the test data sets. Table 7 shows the average performance comparison of these three models. The final result is the point that all the training data were used during the entire incremental learning process.

As Fig. 2 shown, three models have got the similar trends in the accuracy with test data sets from Entry2 to Entry10. But from Table 7, SVMs consume more than 86 h during the entire increasing updating process, 14 times more than ISVM and AISVM models. It significantly is proved that ISVM and AISVM models are effective to reduce the updating time. Compared the proposed AISVM model with ISVM model, the former one achieves 1.2 percentages improvement of the average classification accuracy without the obvious increment of the updating time. Besides, the number of SVs has a significant impact on the classification performance of the traffic classifier. More SVs usually mean high classification accuracy.

Table 7 The average performance comparison of the three models

Model	Train time H:M:S	SV number	Average
SVMs	86:44:51	1071	96.0%
ISVM	6:06:24	768	94.2%
AISVM	6:35:45	901	95.4%

5 Conclusions

SVMs model is an effective discriminative learning model with high classification performance. However, traditional SVMs do not adapt to incremental updating in traffic classification, because of its high requirement of both time and computing resources. Aiming to tackle the problem, this paper first uses ISVM model in traffic classification. To overcome the drawback of ISVM model, AISVM model is further proposed to effectively make use of potential valuable information contained in the previous training data. The experimental results have proved that the proposed models are more effective than the traditional SVMs model in reducing the high training cost of memory and CPU with high classification performance in the same incremental updating setting.

Future work will include further implementation in online traffic classification. Extensions to applying more incremental learning modes in traffic classification, such as learning for un-supervised and semi-supervised learning, are being considered.

Acknowledgements This work was partly financially supported through grants from the National Natural Science Foundation of China (No. 60903083 and 61502123), Scientific planning issues of education in Heilongjiang Province (No. GBC1211062), and the research fund for the program of new century excellent talents (No. 1155-ncet-008). The authors thank the anonymous reviewers for their helpful suggestions.

References

1. Kim H, Claffy K C, Fomenkov M, et al (2008) Internet traffic classification demystified: myths, caveats, and the best practices[C]//Proceedings of the 2008 ACM CoNEXT conference. ACM, 1-12
2. Karagiannis T, Papagiannaki K, Faloutsos M (2005) BLINC: multilevel traffic classification in the dark[C]//ACM SIGCOMM Computer Communication Review. ACM 35(4):229-240
3. Nguyen T T T, Armitage G (2008) A survey of techniques for internet traffic classification using machine learning[J]. IEEE Commun Surveys Tutor 10(4):56-76
4. Wang Y, Yu S Z (2008) Machine learned real-time traffic classifiers[C]. Intelligent Information Technology Application, 2008. IITA'08. Second International Symposium on IEEE, 3 p 449-454
5. Este A, Gringoli F, Salgarelli L (2009) Support vector machines for TCP traffic classification[J]. Comput Netw 53(14):2476–2490
6. Yuan R, Li Z, Guan X et al (2010) An SVM-based machine learning method for accurate internet traffic classification[J]. Inf Syst Front 12(2):149–156
7. Bashar A, Parr G et al (2014) Application of Bayesian networks for autonomic network management. J Netw Syst Manag 22(2):174–207
8. Zhang J, Xiang Y et al (2013) Network traffic classification using correlation information. IEEE Trans Parallel Distrib Syst 24(1): 104–117
9. Monemi A, Zarei R, Marsono MN (2013) Online NetFPGA decision tree statistical traffic classifier[J]. Comput Commun 36(12): 1329–1340
10. Auld T, Moore AW, Gull SF (2007) Bayesian neural networks for internet traffic classification[J]. IEEE Trans Neural Netw 18(1): 223–239
11. Syed NA, Liu H, and Sung KK (1999) Handling concept drifts in incremental learning with support vector machines. In Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM Press, New York, NY, pages 272-276
12. A. Moore and K. Papagiannaki. Toward the Accurate Identification of Network Applications. In Proceedings of PAM Workshop, 2005
13. Pan Z, Liu S, Fu W (2017) A review of visual moving target tracking[J]. Multimed Tools Appl 76(16):16989–17018
14. Liu S, Cheng X, Fu W et al (2014) Numeric characteristics of generalized M-set with its asymptote[J]. Appl Math Comput 243: 767–774
15. Dainotti A, Pescape A, and Claffy KC (2012) Issues and future directions in traffic classification. IEEE Netw 26(1):35-40
16. Perera P, Tian Y C, Fidge C, et al (2017) A Comparison of Supervised Machine Learning Algorithms for Classification of Communications Network Traffic[C]//International Conference on Neural Information Processing. Springer, Cham, 445-454
17. Moore A and Zuev D (2005) Internet traffic classification using Bayesian analysis techniques[C]. ACM SIGMETRICS-2005, Banff, Alberta, p 50-60
18. Williams N, Zander S, Armitage G (2006) A preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification[J]. ACM SIGCOMM Comput Commun Rev 36(5):5–16
19. Finamore A, Mellia M, Meo M, et al (2010) Kiss: Stochastic packet inspection classifier for udp traffic[J]. IEEE/ACM Trans Netw (TON) 18(5):1505-1515
20. Nguyen TTT, Armitage G, Branch P, et al (2012) Timely and continuous machine-learning-based classification for interactive IP traffic[J]. IEEE/ACM Trans Netw (TON) 20(6):1880-1894
21. Ye W, Cho K (2014) Hybrid P2P traffic classification with heuristic rules and machine learning[J]. Soft Comput 18(9):1815–1827
22. Li D, Hu G, Wang Y et al (2015) Network traffic classification via non-convex multi-task feature learning[J]. Neurocomputing 152:322–332
23. Peng L, Yang B, Chen Y (2015) Effective packet number for early stage internet traffic identification[J]. Neurocomputing 156:252–267
24. Ruping S. (2001) Incremental learning with support vector machines[C]. Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on. IEEE, p 641-642
25. Laskov P, Gehl C, Kruger S, Muller K (2006) Incremental support vector learning: analysis, implementation and applications[J]. J Mach Learn Res 7:1909–1936
26. Shilton A, Palaniswami M, Ralph D et al (2005) Incremental training of support vector machines[J]. IEEE Trans Neural Netw 16(1): 114–131
27. Tsai C H, Lin C Y, Lin C J. (2014) Incremental and decremental training for linear classification[C]. Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, p 343-352
28. Moore AW and Zuev D (2005) Discriminators for use in flow-based classification [R]. Technical report, Intel Research, Cambridge