Computers and Electrical Engineering 000 (2018) 1–8

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Network traffic classification based on transfer learning[☆]

Guanglu Sun[a,b], Lili Liang[a], Teng Chen[a], Feng Xiao[a], Fei Lang[b,c,*]

[a] School of Computer Science and Technology, Harbin University of Science and Technology, Harbin, 150080, China
[b] Research Center of Information Security & Intelligent Technology, Harbin University of Science and Technology, Harbin, 150080, China
[c] School of Foreign Languages, Harbin University of Science and Technology, Harbin, 150080, China

ARTICLE INFO

Article history:
Received 3 September 2017
Revised 4 March 2018
Accepted 5 March 2018
Available online xxx

Keyword:
Traffic classification
Machine learning
Transfer learning
Domain adaptation
Maximum entropy model

ABSTRACT

Machine learning models used in traffic classification make the assumption that the training data and test data have independent identical distributions. However, this assumption might be violated in practical traffic classification due to changes of traffic features. The models trained by existing data will be ineffective in classifying new traffic. A transfer learning model without making the above assumption is proposed in the present study. The maximum entropy model (Maxent) was adopted as the base classifier in the transfer learning model. To examine the efficacy of the proposed method, the traffic dataset collected at the University of Cambridge was used in the condition that the training and test dataset were not identical. Experimental results showed that good classification performance was obtained based on the transfer learning model.

1. Introduction

With the increasing number of threats to networks, it is critical for network managers to have deeper understanding about the applications running in their networks. Traffic classification is a vital aspect of network management which enables to make classification of all network traffic [1].

In the last few years, many new network protocols tried to escape from monitoring by means of the disguised methods, such as dynamic port, encapsulation and encryption. It rendered port-based and payload-based methods unreliable. The researchers were motivated to use the application types as categories, the traffic statistical properties generated by network communication as features. Then machine learning models were utilized. It is widely used for machine learning based methods in traffic classification, including supervised learning model, unsupervised learning model and semi-supervised learning model [2]. However, there are still two big challenges in traffic classification. The first is being in line with the rapid growth of new applications. The second is that different training models are required as network topology and time change. In unsupervised learning model, it is hard to build a practical traffic classifier with the clustering results and no regard for the instruction of the real traffic classes [3]. As for semi-supervised learning model, it utilizes a small set of labeled traffic flows and a large set of unlabeled traffic flows during the training process. And for supervised learning model, a bigger set of labeled traffic flows are applied. Both of them achieved satisfying results, when the training and test dataset are identical. However, the changes of time, locations and traffic types make traditional machine learning models less effective than expected, because the training dataset collected in the past and the newly generated test data are not identical [4]. Although the model trained by the outdated data, cannot work well on the new network traffic, the outdated data should not be given

☆ Reviews processed and recommended for publication to the Editor-in-Chief by Guest Editor Dr. S. Liu.

* Corresponding author.
E-mail address: langfei@hrbust.edu.cn (F. Lang).

https://doi.org/10.1016/j.compeleceng.2018.03.005
0045-7906/

up, because it is difficult to acquire the gold-standard labeled traffic, which has the ability to describe the whole network traffic for machine learning methods. Thus, the outdated data with valuable knowledge should be reused for new tasks of traffic classification. It is unnecessary for transfer learning to be with the independent identical distribution (iid). And even for different tasks, transfer learning still has advantages to overcome the above obstacles [5]. It is assumed that a learning task and corresponding data are in a source domain, and a learning task and corresponding data are in a target domain, the objective of transfer learning is to improve rate of target prediction functions by learning the knowledge from the source domain. In contrast to traditional machine learning models, transfer learning does not need to make the iid assumption on the data between two domains [6].

Considering the situation that the data distribution will become different with the change of time, locations and traffic types, TrAdaBoost model [6] are introduced for traffic classification task, which is a distinguished inductive transfer learning model based on instance. Maxent is meanwhile utilized as a base classifier. Aiming at transferring the valuable knowledge in the source domain to the target task, TrAdaBoost uses few labeled data from the target domain to evaluate the availability of the data in the source domain. After that, the valuable auxiliary data are extracted from the source data and combined with the above labeled data in the target domain to train the classifier. By means of transferring the useful knowledge from the source domain to the target domain, TrAdaBoost helps the learning task in the target domain. The key contributions made by the present study are shown as the following.

- In the new traffic classification task, TrAdaBoost is presented to utilize the labeled traffic data extracted from different network traffic sources.
- Maxent model is applied as a base classifier in TrAdaBoost. The proposed method implements the transfer of traffic knowledge from the source domain to the target domain.
- The experiments are conducted to evaluate the performance of TrAdaBoost. The proposed method is compared with traditional machine learning methods, in the condition that there is no enough labeled data to train a learning model effectively in the target domain.

The structure of the present study is as the following. Section 2 briefly introduces related work of traffic classification and transfer learning. Section 3 summarizes the notations and tasks definition. Section 4 proposes the algorithms in detail. Then the datasets and the experimental results are given in Section 5. Section 6 concludes the present study and discusses future work.

2. Related work

In the past decade, a vast outpouring of research is carried out on various methods to deal with internet traffic classification, such as port-based methods, payload-based methods, behavior-based methods and machine learning based methods. Most of the methods are summarized in several surveys chronologically [1,7]. Among these methods, machine learning methods have been paid increasing attention to for its good performance in traffic classification task. Nguyen and Armitage reviewed the related studies based on machine learning models till 2008 in detail [2]. Thus, we mainly review the previous work of machine learning based methods after 2008.

The machine learning methods used for traffic classification task are classified into supervised learning methods, semi-supervised learning methods and unsupervised learning methods. In supervised learning methods, the traffic flows are manually labeled according to the generating application categories, as the gold-standard benchmark dataset. The supervised machine learning model is built based on the labeled flows, and its features are statistical patterns extracted from the flows. The new traffic flows are classified by the model that adjusts its parameters during the training process. In this way, many models were implemented in traffic classification. Moore and Zeuv [8] put forward Bayesian method to identify application protocols. They further made the accuracy higher by refining the variants. Five supervised models were compared in their accuracy of classification and performance of computation, which includes Naive Bayes with discretization, Naive Bayes with kernel density estimation, C4.5 decision tree, Bayesian network and naive Bayes tree [9]. Este et al. [10] applied Support Vector Machines model (SVMs) on three types of well-known datasets and obtained an average accuracy over 95%, 2.3% over the best performance of Bayesian methods and other methods on the same datasets. Finamore et al. [11] further presented statistical characterization of payload as features and used SVM to conduct traffic classification. Nguyen et al. [12] trained the ML models with a set of sub-flows and investigated various strategies of sub-flow selection. The accuracy of their model would be maintained when the traffic mixed up bi-directional flows. A hybrid method with heuristic rules and REPTree model was proposed to classify P2P traffic with different levels of features [13]. Li et al. [14] utilized logistic regression model to classify the flows using non-convex multi-task feature selection. They tried a Capped $-\ell_1,\ell_1$ to learn the features of flows as the regularizer. Peng et al. [15] verified 5–7 packets are the best numbers of packets for early step traffic classification based on 11 well-known supervised learning models.

As for unsupervised methods, the main application is clustering-based methods. The clustering-based model automatically groups unlabeled traffic flows into a set of clusters. The clusters mapping to the different applications are utilized to train a new traffic model. The expectation maximization model was utilized to cluster traffic flows and label each cluster to an application manually [16]. The k-means, DBSCAN and AutoClass models were verified and summarized that the clustering methods got the good-quality clusters when the number of clusters reached a certain scale [17]. The method derived signatures from unidentified traffic flows automatically. However, the difference between the number of clusters and applications

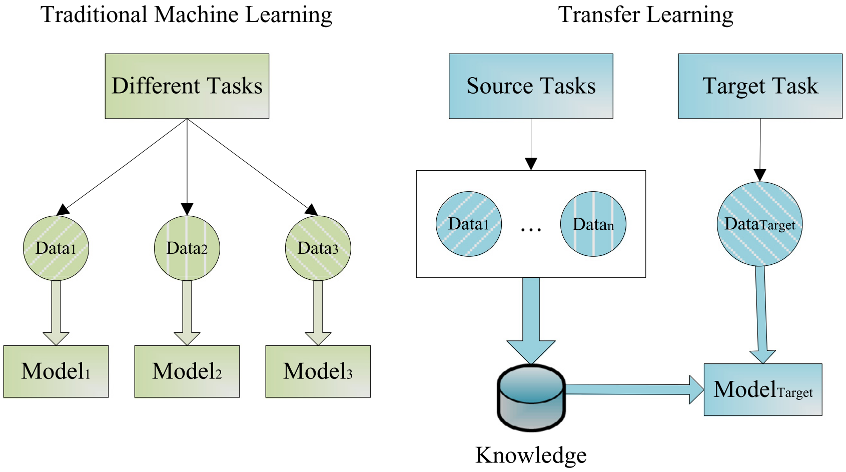

Fig. 1. Comparison between transfer learning and traditional learning methods.

resulted in hard mapping. To tackle this problem, Keralapura et al. [18] introduced a two-stage self-learning architecture for classifying P2P traffic, which develops the temporal correlation of flows and identifies P2P traffic. Zhang et al. [19] utilized a bag-of-words model to stand for the clusters content which is built with statistical features. The latent semantic analysis was then implemented to merge similar clusters on the basis of payload content. Wang et al. [20] presented a constrained clustering framework that realizes decision making considering equivalence set constraints and the background knowledge, and used Gaussian mixture density to model the observed data.

In semi-supervised learning methods, the learning model leverages unlabeled flows to enhance the initial supervised learning model. The labeled and unlabeled flows are selected to obtain better accuracy in the semi-supervised learning model. A set of labeled flows were applied to mapping the clusters to real applications [21]. The ensemble clustering method was present to improve the semi-supervised method, based on the combination of sub-space clustering, evidence accumulation, and hierarchical clustering [22]. Aiming at making unknown flows detection perform better, Zhang et al. extended the former semi-supervised work to utilize the correlation information in the compound classification model, the nearest-neighbor (NN) based model, and the bag-of-flow model, respectively [23].

Different types of learning models tackle different problems with different assumptions. But all the above methods are with the iid assumption, which cannot always be satisfied in real application. Transfer learning is put forward to resolve the above issue without the training and test data being identical. Dai and Yang [6] proposed transfer learning model and verified the effectiveness of the model on three text datasets and one none-text dataset of the UCI machine learning repository. Pan and Yang [4] categorized transfer learning into three types, inductive transfer learning, transductive transfer learning and unsupervised transfer learning on the basis of various definitions of transfer learning, diverse conditions between source and target domain, source and target tasks. Lu et al. [24] concluded several transfer learning methods and applications in their survey paper. Recently, transfer learning has been successfully applied into practical uses, such as computer vision, image processing, biology, natural language processing, and text data mining [25].

3. Problem statement

The purpose of traffic classification is to map each flow to an application protocol in the network. Traditional machine learning methods used for traffic classification task implement the training dataset $(X^{train}, Y^{train}) = \{(x_1, y_1), (x_2, y_2) ... (x_n, y_n)\}$ extracted from the original traffic flows to train a traffic classifier. Then the classifier identifies the instance of test dataset X^{test}. There is a common assumption that X^{train} and X^{test} are extracted from the same feature space and under the same distribution. In order to identify new traffic data generated from different domain, the original traffic classifier must be retrained with the data from the same domain.

Transfer learning does not need the above assumptions. The comparison between transfer learning and traditional machine learning methods are shown in Fig. 1.

Given one or more source domains D_s and tasks T_s, corresponding to one or more original labeling datasets of traffic flows and traffic classification tasks, one target domain D_t and task T_t, corresponding to the new traffic classification task, which comes from different network environment, the objective of transfer learning is to improve the learning ability of target prediction function h_t in D_t using the knowledge from D_s and T_s ($D_s \neq D_t$ or $T_s \neq T_t$). If the number of labeled training data T_b is not enough to train a satisfying learning model in D_t, the labeled training data T_a in D_s is introduced to train the learning model in D_t as the auxiliary data. Although T_a is outdated, it should not be given up, because it is expensive to label traffic data. In the present study, we consider T_a in one source D_s and a small quantity of T_b in D_t as the

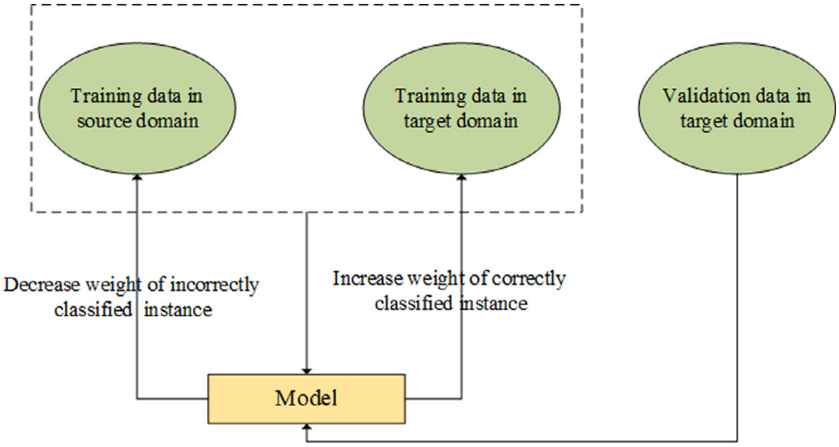

Fig. 2. The process of TrAdaBoost training.

training data T to train a learning model, where the knowledge in D_s is transferred. Then the training model classifies new flows as test data S in D_t.

4. Methods

Dai et al. [6] proposed TrAdaBoost in 2007, which is a modified version of the AdaBoost method based on the thought of transfer learning. Their experiments use T_a and one part of T_b ($T_b \ll T_a$) as the training data, and use the other part of T_b as the test data. TrAdaBoost reduces the weight of "bad" data from T_a by updating samples weight, which is useless to identify the test data in D_t.

In the iteration of AdaBoost training, if a sample is incorrectly classified, the model will pay more attention to the sample by increasing the weight of the sample in the next iteration. If a sample is correctly classified, its weight will be decreased. In contrast, if a sample in D_s is incorrectly classified in TrAdaBoost training, it will be different from the data in D_t and its weight will be decreased. In the next iteration, the influence of the sample will be reduced. If a sample in D_t is incorrectly classified, its weight will be increased. The model will pay more attention to the sample in the next iteration. The training process of TrAdaBoost is shown in Fig. 2.

In the present study, TrAdaBoost is applied to address multi-class network traffic classification as a learning framework. Many machine learning models are suitable for being the base classifier. In the present study, Maxent is utilized as the base classifier. The description of traffic classification based on TrAdaBoost is as shown below (Algorithm 1).

Algorithm 1 Traffic classification algorithm based on TrAdaBoost.

Input: T_a from D_s
T_b from D_t,
The merged training data $T = T_a \cup T_b$
Test data S from D_t
Iteration number N
1. Initialize the weight vector $W^1 = (w_1^1, \ldots\ldots, w_{n+m}^1)$, where
$$w_i^1 = \begin{cases} 1/n, & i = 1, \ldots\ldots, n \\ 1/m, & i = n+1, \ldots\ldots, n+m \end{cases}$$
2. Set $\beta = 1/(1 + \sqrt{2\ln(n/N)})$
3. For $t = 1, \ldots\ldots, N$
(1) Normalize $P^t = \frac{W^t}{\sum_{i=1}^{n+m} w_i{}^t}$.
(2) Call the base classifier based on the merged training dataset T and the weight distribution P^t of T, a classification hypothesis h_t is then obtained: $X \to Y$.
(3) Calculate the error rate ε_t of h_t on T_b:
$$\varepsilon_t = \sum_{i=n+1}^{n+m} \frac{w_i{}^t |h_t(x_i) - c(x_i)|}{\sum_{i=n+1}^{n+m} w^t{}_i}.$$
(4) Update the β_t and $\beta_t = \varepsilon_t/(1 - \varepsilon_t)$.
(5) Update the samples weight:
$$w_i{}^{t+1} = \begin{cases} w_i{}^t \beta^{|h_t(x_i) - c(x_i)|}, & 1 \le i \le n \\ w_i{}^t \beta_t{}^{-|h_t(x_i) - c(x_i)|}, & n+1 \le i \le n+m \end{cases}$$
Output: The final classifier.
$$h_f(x) = \arg\max_{y \in Y} \sum_{t=\lceil N/2 \rceil}^{N} [h_t(x) = y] \ln(1/\beta_t).$$

Table 1
The descriptions of Cambridge traffic dataset.

Cambridge dataset	Data duration	Samples number
entry 1	1821.8 s	24,863
entry 2	1696.7 s	23,801
entry 3	1724.1 s	22,932
entry 4	1784.1 s	22,285
entry 5	1794.9 s	21,648
entry 6	1658.5 s	19,384
entry 7	1739.2 s	55,835
entry 8	1665.9 s	55,494
entry 9	1664.5 s	66,248
entry 10	1613.4 s	65,036
entry 12	N/A	7839

Table 2
The descriptions of selected features.

Feature	Notation
server_port	server port
min_iat	min packet inter-arrival of a flow
mean_iat	mean packet inter-arrival of a flow
var_iat	variance packet inter-arrival of a flow
mean_data_wire	mean Ethernet packet bytes of a flow
mean_data_ctrl	mean control bytes of a flow
avg_win_agv_c	average window size of the client to server
mean_data_wire_s	mean Ethernet packet bytes of the server to client
mean_data_ip_s	mean IP packet bytes of the server to client

In step 1, T_a and T_b are merged as the training dataset T. The weight of samples W^1 are initialized (n is the sample number of T_a and m is the sample number of T_b). In step 2, β_t is the factor of model in each iteration, which is initialized based on n and N. In step 3, P^t is the weight distribution of $T_,$ which is decided by W^t. In each iteration, training samples are firstly selected with the probability of P^t, so P^t decides how the weight of samples affects the model training in the current iteration. The Maxent model is then built as the base classifier with the selected training samples. In the classification hypothesis h_t: $X->Y$, the error rate ε_t is calculated based on T_b and their weights $W_i{}^t$. β_t and $W_i{}^t$ will be updated based on the new error rate. After that, if a sample in T_a is incorrectly classified, its weight will be multiplied by $\beta^{|h_t(x_i)-c(x_i)|} \in [0, 1]$, to reduce the sample weight based on Hedge (β) theory [26,27]. If a sample in T_b is incorrectly classified, its weight will be multiplied by $\beta_t{}^{-|h_t(x_i)-c(x_i)|}$, to increase the sample weight. In the next iteration, the new base classifier will pay more attention to this sample. After each iteration, the hyperplane of TrAdaBoost becomes closer to target domain D_t. The samples in target domain will be identified better. The model weight $\ln(1/\beta_t)$ is stored with each base classifier. The final result is based on the sum of each classifier with its weight. Test data S is utilized to evaluate the performance of TrAdaBoost.

5. Experiments and analysis

5.1. Dataset

The traffic dataset of Cambridge's Nprobe project is introduced to conduct the performance evaluation of the method in the present study. The dataset has been applied widely in traffic classification, such as experiments based on Bayesian methods by Moore and Zuev [8]. They proposed various features of traffic flows, including packet length, inter-packet timing, and information derived from traffic flows. The total dataset contains 11 subsets, e.g., entry 1–10 and entry 12, as shown in Table 1. Entry 1–10 datasets were collected at different time in one day. Entry 12 and the other 10 datasets are not identical, as entry 12 was collected after 12 months and network environment changes. The dataset includes 12 application classes, including WWW, MAIL, FTP-CONTROL, FTP-PASV, FTP-DATA, P2P, DATABASE, SERVICES, ATTACK, MULTIMEDIA, INTERACTIVE and GAMES. Because not all categories have enough traffic samples, six classes are chosen in the experiments, including WWW, MAIL, DATABASE, FTP-DATA, P2P, SERVICES.

Each sample in the dataset consists of 248 features and one class label. Some features play a minimal and even negative role in improving the classification accuracy [8]. In order to reduce the feature dimension, the consistency algorithm was applied to select features combining with greedy searching strategy. Nine features were selected and shown in Table 2. Then, entry 1–4 datasets are combined as T_a, in which the number of WWW and MAIL samples is reduced to avoid the influence of the imbalanced data. Entry 12 is divided into T_b and S. The ratio between T_a and T_b is more than 35 times. The datasets are applied in the experiments, as shown in Table 3.

Please cite this article as: G. Sun et al., Network traffic classification based on transfer learning, Computers and Electrical Engineering (2018), https://doi.org/10.1016/j.compeleceng.2018.03.005

Table 3
The descriptions of experimental datasets.

Dataset	Set name	samples counts
Training dataset as T_a	entry 1, entry 2, entry 3 and entry 4	21,520
Train dataset as T_b	part of entry 12	600
Test dataset as S	the other part of entry 12	7050

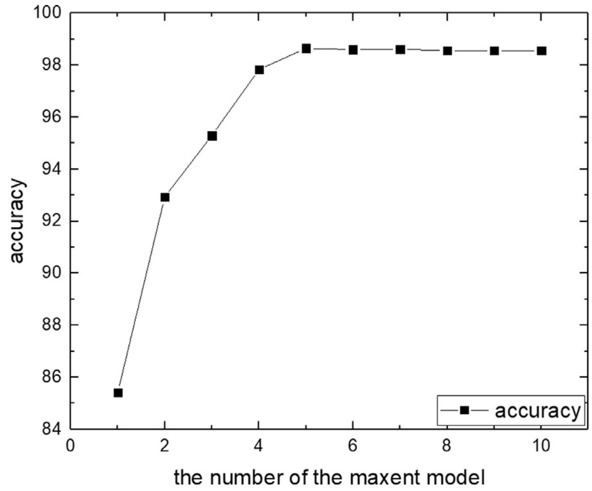

Fig. 3. The tendency of performance with the number of base classifier increasing.

Table 4
The accuracy of three methods on test dataset.

Method	NoTL(T_a)	NoTL($T_a + T_b$)	TrAdaBoost
Accuracy	81.2%	85.4%	98.7%

Table 5
The accuracy of three methods on test dataset (per class).

Method	WWW	MAIL	DATABASE	FTP-DATA	P2P	SERVICES
NoTL (T_a)	86.8%	86.2%	43.3%	54%	63.7%	44.3%
NoTL($T_a + T_b$)	93.7%	94.6%	61.8%	58.9%	71.4%	57.6%
TrAdaBoost	99.8%	99.7%	92.6%	95.1%	91.8%	91.4%

5.2. Experimental setting

Two traditional machine learning methods based on Maxent are utilized as comparison methods in traffic classification. One is called NoTL (T_a). Maxent is implemented to train the classifier with T_a in D_s. The other is called NoTL($T_a + T_b$). Maxent is also applied to train the classifier with T_a and T_b.

The main metric applied to assess the effectiveness of traffic classification is accuracy that is the percentage that traffic flows correctly classified accounts for of the total traffic flows [28]. The accuracy is obtained for the whole test dataset and each class.

5.3. Results

In TrAdaBoost, the number of base classifiers is critical to the performance of classification. Different numbers of base classifiers are tried, as shown in Fig. 3.

When the number of Maxent is 5, the performance of TrAdaBoost is the best. After that, the model sometimes reduces the performance, because it does not always lower the generalization error as some research noted.

The results of three methods are shown in Table 4. TrAdaBoost performs better than the comparison methods in accuracy. The overall performance is greatly improved based on transfer learning model. In Table 5, the accuracy of each class is shown respectively. In each class, transfer learning method also performs better than the comparison methods. As WWW and MAIL classes have enough data to train the classifier, NoTL (T_a) method achieves the accuracy of 86.8% and 86.2%, although the training dataset and test dataset are not identical. When T_b is added to the training dataset, NoTL($T_a + T_b$) method achieves the better accuracy of 93.7% and 94.6%. The reason is that T_b and test dataset are identical. It describes the characteristics of test dataset better. However, TrAdaBoost achieves the performance much better. The improvements of accuracy in the other 4 classes are more obvious, as the class with fewer samples is easily affected by the distribution of training samples. TrAdaBoost retains the training samples which help the new classification task to a great extent, whether the samples belong to T_a or T_b. Further, it transferred useful knowledge from source domain D_s into target domain D_t, when T_b is too few to train a satisfying classifier.

6. Conclusion and future work

Changes of network environment lead to performance degradation of traffic classification based on traditional machine learning models. TrAdaBoost for multi-class task is utilized to transfer the knowledge from source domain into target domain in traffic classification. Maxent model is applied as the base classifier. Based on the application of the proposed method, high classification accuracy is achieved in the new dataset. The new dataset and most of the training dataset are not identical.

In the future study, more efforts will be taken to apply transfer learning models into solving problems of classification, e.g., how to carry out classification when the source domain and target domain are with different class labels.

Acknowledgments

This work was partly financially supported through grants from the National Natural Science Foundation of China (No. 61702140 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.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.compeleceng.2018.03.005.

References

[1] Kim H, Claffy KC, Fomenkov M, et al. Internet traffic classification demystified: myths, caveats, and the best practices. In: Proceedings of the 2008 ACM CoNEXT conference. ACM; 2008. p. 11.
[2] Nguyen TTT, Armitage G. A survey of techniques for internet traffic classification using machine learning. IEEE Commun Surv Tutorials 2008;10(4):56–76.
[3] Zhang J, Xiang Y, Wang Y, et al. Network traffic classification using correlation information. IEEE Trans Parallel Distrib Syst 2013;24(1):104–17.
[4] Pan SJ, Yang Q. A survey on transfer learning. IEEE Trans Knowl Data Eng 2010;22(10):1345–59.
[5] Raina R, Ng AY, Koller D. Constructing informative priors using transfer learning. In: Proceedings of the 23rd international conference on machine learning. ACM; 2006. p. 713–20.
[6] Dai W, Yang Q, Xue GR, et al. Boosting for transfer learning [C]. In: Proceedings of the 24th international conference on machine learning. ACM; 2007. p. 193–200.
[7] Dainotti A, Pescape A, Claffy KC. Issues and future directions in traffic classification. IEEE Netw 2012;26(1):35–40.
[8] Moore A, Zuev D. Internet traffic classification using Bayesian analysis techniques; 2005. p. 50–60.
[9] Williams N, Zander S, Armitage G. A preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification. ACM SIGCOMM Comput Commun Rev 2006;36(5):5–16.
[10] Este A, Gringoli F, Salgarelli L. Support vector machines for TCP traffic classification. Comput. Netw 2009;53(14):2476–90.
[11] Finamore A, Mellia M, Meo M, et al. KISS: stochastic packet inspection classifier for UDP traffic. IEEE/ACM Trans Netw 2010;18(5):1505–15.
[12] Nguyen TTT, Armitage G, Branch P, et al. Timely and continuous machine-learning-based classification for interactive IP traffic. IEEE/ACM Trans Netw 2012;20(6):1880–94.
[13] Ye W, Cho K. Hybrid P2P traffic classification with heuristic rules and machine learning. Soft Comput 2014;18(9):1815–27.
[14] Li D, Hu G, Wang Y, et al. Network traffic classification via non-convex multi-task feature learning. Neurocomputing 2015;152:322–32.
[15] Peng L, Yang B, Chen Y. Effective packet number for early stage internet traffic identification. Neurocomputing 2015;156:252–67.
[16] McGregor A, Hall M, Lorier P, et al. Flow clustering using machine learning techniques. In: International workshop on passive and active network measurement. Springer; 2004. p. 205–14.
[17] Erman J, Arlitt M, Mahanti A. Traffic classification using clustering algorithms. In: Proceedings of the 2006 SIGCOMM workshop on mining network data. ACM; 2006. p. 281–6.
[18] Keralapura R, Nucci A, Chuah CN. A novel self-learning architecture for p2p traffic classification in high speed networks. Comput Networks 2010;54(7):1055–68.
[19] Zhang J, Xiang Y, Zhou W, et al. Unsupervised traffic classification using flow statistical properties and IP packet payload. J Comput Syst Sci 2013;79(5):573–85.
[20] Wang Y, Xiang Y, Zhang J, et al. Internet traffic classification using constrained clustering. IEEE Trans Parallel Distrib Syst 2014;25(11):2932–43.
[21] Erman J, Mahanti A, Arlitt M, et al. Semi-supervised network traffic classification, 35. ACM; 2007. p. 369–70.
[22] Casas P, Mazel J, Owezarski P. MINETRAC: mining flows for unsupervised analysis & semi-supervised classification. In: Proceedings of the 23rd international teletraffic congress; 2011. p. 87–94.
[23] Zhang J, Chen C, Xiang Y, et al. An effective network traffic classification method with unknown flow detection. IEEE Trans Netw Serv Manage 2013;10(2):133–47.
[24] Lu J, Behbood V, Hao P, et al. Transfer learning using computational intelligence: a survey. Knowledge-Based Syst 2015;80:14–23.
[25] Yang L, Jing L, Ng MK. Robust and non-negative collective matrix factorization for text-to-image transfer learning. IEEE Trans Image Process 2015;24(12):4701–14.
[26] Freund Y, Schapire R. A decision-theoretic generalization of on-line learning and an application to boosting. J Comput Syst Sci 1997;55(1):119–39.
[27] Liu S, Cheng X, Fu W, et al. Numeric characteristics of generalized M-set with its asymptote. Appl Math Comput 2014;243:767–74.
[28] Liu S, Lu M, Liu G. A novel distance metric: generalized relative entropy. Entropy 2017;19(6):269.

Guanglu Sun is currently a professor in the School of Computer Science and Technology, and a director in the Center of Information Security and Intelligent Technology at Harbin University of Science and Technology, China. He received his B.S., M.S. and Ph.D. from Harbin Institute of Technology. His research interests include computer networks and security, machine learning, and intelligent information processing.

Lili Liang is a master candidate in the School of Computer Science and Technology at Harbin University of Science and Technology. Her research interests include computer networks and security, machine learning.

Teng Chen is a master candidate in the School of Computer Science and Technology at Harbin University of Science and Technology. His research interests lie in machine learning, intelligent information processing.

Feng Xiao is a master candidate in the School of Computer Science and Technology at Harbin University of Science and Technology. His research interests lie in computer networks and security, machine learning.

Fei Lang is an associate professor in the School of Foreign Languages, and a researcher in the Center of Information Security and Intelligent Technology at Harbin University of Science and Technology. She received her M.S. from Nottingham University and her Ph.D. from Shanghai International Studies University. Her research interests include corpus linguistics, cognitive psychology and machine-aided translation.