ISSN:0975-9646

Classifying Populated IP Addresses using SVM Clustering and Vertical partition ID3 Decision Tree

Gagan Sharma, Yogadhar Pandey
COMPUTER SCIENCE DEPT, SIRT, BHOPAL

Abstract: The **addresses that are estimated the number of the users of a specific application behind IP address (IPs). This problem is central to combating abusive traffic, such as DDoS attacks, ad click fraud and email spam, scams, phishing, and malware distribution. Here we proposed an efficient method to classify the IP addresses that are associated with a large number of user requests. The idea is to classify the network traffic based on the IP addresses by first clustering the data using SVM and then applying vertical partition based id3 decision tree.**

1. INTRODUCTION

Online services such as Web-based email, search, and online social networks are becoming increasingly popular. While these services have become everyday essentials for billions of users, they are also heavily abused by attackers for nefarious activities such as spamming, phishing, and identity theft [1].

Simple conventional mechanisms for abuse detection that rely on source IPs set a limit, i.e., filtering threshold, on the IP activity within a time period. Once the limit is reached by an IP, either the IP traffic gets filtered for the rest of that time period, or the IP gets blacklisted for several consecutive periods. These techniques typically set the same threshold for all IPs. Setting an aggressive threshold yields a high false positive rate since some IPs have numerous users behind them and are hence expected to send relatively large traffic volumes. Setting a conservative threshold yields a high false negative rate, since the threshold becomes ineffective for distributed attacks where IPs send relatively little traffic. This work tailors the thresholds to the sizes of the IPs. It proposes a new framework for timely estimation of the number of users behind IPs with high enough accuracy to reduce false positives and with high enough coverage in the IP space to reduce false negatives [2].

Populated IP addresses (PIP) - IP addresses that are associated with a large number of user requests are important for online service providers to efficiently allocate resources and to detect attacks. While some PIPs serve legitimate users, many others are heavily abused by attackers to conduct malicious activities such as scams, phishing, and malware distribution. Unfortunately, commercial proxy lists like Quova have a low coverage of PIP addresses and offer little support for distinguishing good PIPs from abused ones [1].

On the one hand, not all proxies, NATs, or gateways are PIP addresses. Some may be very infrequently used and thus are not of interest to online service providers. On the other hand, while some PIP addresses may belong to proxies or big NATs, many others are not real proxies. Some are dial-up or mobile IPs that have high churn rates. Others include IP addresses from large services, such as Facebook that connects to Hotmail to obtain user email contacts. Additionally, not all PIPs are associated with a large number of actual users. Although many good PIPs like enterprise-level proxies are associated with a large number of actual users, some abused PIPs may be associated with few real users but a large number of fake user accounts controlled by attackers. In an extreme case, bad PIPs may be entirely set up by attackers. For example, it is observed that >30% of the IP addresses that issue more than 20 sign-up requests to Windows Live per day are actually controlled by attackers, with all sign-ups for malicious uses.

Classifying PIPs is a challenging task for several reasons. First, ISPs and network operators consider the size and distribution of customer populations confidential and rarely publish their network usage information. Second, some PIP addresses are dynamic, e.g., those at small coffee shops with user population sizes changing frequently. Third, good PIPs and bad PIPs can locate next to each other in the IP address space. For example, attackers can buy or compromise Web hosting IPs that are right next to the IPs of legitimate services. In addition, a good PIP can temporarily be abused. Due to these challenges, not surprisingly, it is observed that commercial proxy lists offer a low precision in identifying PIPs and provide no support for distinguishing good PIPs from bad ones [1].

Clustering is a division of data into groups of similar objects. Each group called cluster, consists of objects that are similar amongst them and dissimilar compared to object of other groups. Representing data by fewer clusters necessarily loses certain fine details, but achieves simplification. It represents many data objects by few clusters, and hence it models data by its clusters [3].

Supervised Clustering Task: Clustering is sometimes applied to multiple sets of items, with each set being clustered separately. For example, in the noun-phrase co reference task, a single document's noun-phrases are clustered by which noun phrases refer to the same entity, and in news article clustering, a single day's worth of news articles are clustered by topic. In this method, users provide complete clustering of a few of these sets to express their preferences, e.g., provide a few complete clustering of several documents' noun-phrases, or several days' news articles [4].

SVM based clustering: The structural SVM algorithm provides a general framework for learning with complex structured output spaces [5]. This work shares many similarities with the semi supervised clustering, which attempts to form desirable clustering's by taking user information into account, typically of the form "these items do/do not belong together." Some supervised clustering methods modify a clustering algorithm so it satisfies constraints [6]. Clustering is sometimes useful to

numerous sets of items, with each set being clustered separately. In this, users provide a few complete clustering's of several documents' noun-phrases, or several days' news articles.

ID3 Algorithm: The ID3 algorithm (Inducing Decision Trees) was originally introduced by Quinlan in [7] and is described below in Algorithm. Here they briefly recall the steps involved in the algorithm. For a thorough discussion of the algorithm we refer the interested reader to [8].

Require: R, a set of attributes.
Require: C, the class attribute.
Require: S, data set of tuples.
1: if R is empty then
2: Return the leaf having the most frequent value in data set S.
3: else if all tuples in S have the same class value then
4: Return a leaf with that specific class value.
5: else
6: Determine attribute A with the highest information gain in S.
7: Partition S in m parts S(a1), ..., S(am) such that a1, ..., am are the different values of A.
8: Return a tree with root A and m branches labeled a1...am, such that branch i contains ID3(R − {A}, C, S (ai)).
9: end if

2. RELATED WORK

Chi-Yao Hong et. Al. proposes PIPMiner, a fully automated method to extract and classify PIPs through analyzing service logs. Our methods combine machine learning and time series analysis to distinguish good PIPs from abused ones with over 99:6% accuracy. When applying the derived PIP list to several applications, we can identify millions of malicious Windows Live accounts right on the day of their sign- ups, and detect millions of malicious Hotmail accounts well before the current detection system captures them [1].

Alka Gangrade et. al. proposed how to build privacy preserving two-layer decision tree classifier, where database is vertically partitioned and communicate their intermediate results to the UTP not their private data. In our protocol, an UTP allows well-designed solutions that meet privacy constraints and achieve acceptable performance [9].

This paper presents a novel host-based combinatorial method based on k-Means clustering and ID3 decision tree learning algorithms for unsupervised classification of anomalous and normal activities in computer network ARP traffic. The k-Means clustering method is first applied to the normal training instances to partition it into k clusters using Euclidean distance similarity. An ID3 decision tree is constructed on each cluster. Anomaly scores from the k-Means clustering algorithm and decisions of the ID3 decision trees are extracted. A special algorithm is used to combine results of the two algorithms and obtain final anomaly score values. The threshold rule is applied for making decision on the test instance normality or abnormality [10].

Bart Kuijpers et. Al. considers privacy preserving decision tree induction via ID3 in the case where the training data is vertically or vertically distributed. Furthermore, we consider the same problem in the case where the data is both vertically and vertically distributed, a situation we refer to as grid partitioned data. We give an algorithm for privacy preserving ID3 over vertically partitioned data involving more than two parties. For grid partitioned data, we discuss two different evaluation methods for preserving privacy ID3, namely, first merging vertically and developing vertically or first merging vertically and next developing vertically. Next to introducing privacy preserving data mining over grid-partitioned data, the main contribution of this paper is that we show, by means of a complexity analysis that the former evaluation method is the more efficient [11].

This paper is intended to study and compare different data clustering algorithms. The algorithms under investigation are: k-means algorithm, hierarchical clustering algorithm, self-organizing maps algorithm, and expectation maximization clustering algorithm. All these algorithms are compared according to the following factors: size of dataset, number of clusters, type of dataset and type of software used [12].

Cemal Cagatay Bilgin et al. review the significant contributions in the literature on complex evolving networks; metrics used from degree distribution to spectral graph analysis, real world applications from biology to social sciences, problem domains from anomaly detection, dynamic graph clustering to community detection [13].

Chi-Yao Hong et. Al. [1] uses various studies for their work some for result generation techniques [14] [15] [16] [17]. Here also we refer another related data from various resources such as [18] [19].

3. PROBLEM DEFINITION

The algorithm accepts a series of "training clusters," a series of sets of items and clustering's over that set. The method learns a similarity measure between item pairs to cluster future sets of items in the same fashion as the training clusters. But the SVM based clustering is not very efficient for the detection of IP addresses containing huge dataset. The ability to distinguish bad or abused populated IP ad- dresses from good ones is critical to online service using classification algorithm such PIPMiner where the classified accuracy is low.

4. PROPOSED METHODOLOGY

Here we proposed solution algorithm for support Vector Machine (SVM) to classify the data set in to number of clusters.

A. Algorithm for SVM:

1: Input: (x1,y1) ,.....(xn,yn),C,⊑
2. Si←∅ for all i=1,.....n
3. repeat
4. for i=1,.....n do
5. H(y) =$\Delta(yt, y) + w^T\psi(xt, y) - w^T\psi(xt, yt)$
6. compute Ÿ=argmaxy∈ $yH(y)$
7. compute £i=max{0,maxy∈ $SiH(y)$
8. if H(Ÿ)>£i + ∈ $then$
9. Si←Si ∪ $[\hat{Y}]$
10. w←optimize primal over S=∪ iSi

11. end if
12. end for
13. until no Si has changed during iteration.

B. Vertical Partition based id3 decision tree

Input Layer:

- Define P_1, P_2… P_n Parties. (Vertically partitioned).
- Each Party contains R set of attributes A_1, A_2, …., A_R.
- C the class attributes contains c class values C_1, C_2, …., C_c.
- For party P_i where i = 1 to n do
- If R is Empty Then
- Return a leaf node with class value
- Else If all transaction in $T(P_i)$ have the same class Then
- Return a leaf node with the class value
- Else
- Calculate Expected Information classify the given sample for each party P_i individually.
- Calculate Entropy for each attribute (A_1, A_2, …., A_R) of each party P_i.
- Calculate Information Gain for each attribute (A_1, A_2,…., A_R) of each party P_i
- Calculate Total Information Gain for each attribute of all parties (TotalInformationGain( )).
- $A_{BestAttribute} \leftarrow$ MaxInformationGain( )
- Let V_1, V_2, …., V_m be the value of attributes. $A_{BestAttribute}$ partitioned P_1, P_2,…., P_n parties into m parties
- $P_1(V_1)$, $P_1(V_2)$, …., $P_1(V_m)$
- $P_2(V_1)$, $P_2(V_2)$, …., $P_2(V_m)$
- . .
- . .
- $P_n(V_1)$, $P_n(V_2)$, …., $P_n(V_m)$
- Return the Tree whose Root is labelled $A_{BestAttribute}$ and has m edges labelled V_1, V_2, …., V_m. Such that for every i the edge Vi goes to the Tree
- NPPID3(R – $A_{BestAttribute}$, C, ($P_1(V_i)$, $P_2(V_i)$, …., $P_n(V_i)$))
- End.

5. Analysis Parameter

Here we enlist parameter for result analysis on behalf of that we analyze our result.

1.Time complexity
2.Mean Absolute Error
3.Kappa Statistics
4.Classified instances
5.Unclassified instances
6.Mean Relative Error

6. Conclusion

The security plays a vital role during the transmission of data from the sender to the receiver. Although there are various techniques to reduce the number of attacks specially heavy traffic from a particular ip address. Hence the classification of these populated IP address can be detected using a combinatorial method of SVM based clustering and vertical partition based decision tree which provides less computational time.

References

[1] Chi-Yao Hong, Fang Yu, Yinglian Xie "Populated IP Addresses - Classification and Applications", Proceedings of the 2012 ACM conference on Computer and communications security, pp. 329-340, 2012.

[2] Ahmed Metwally and Matt Paduano "Estimating the Number of Users behind IP Addresses for Combating Abusive Traffic", Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 249-257. 2011.

[3] Han J. and Kamber M. "Data Mining: Concepts and Techniques", Morgan Kaufmann Publishers, 2001.

[4] Thomas Finley, Thorsten Joachims "Supervised Clustering with Support Vector Machines", Proceedings of the 22 nd International Conference on Machine Learning, pp. 217 – 224, 2005.

[5] Tsochantaridis, I., Hofmann, T., Joachims, T., & Altun, Y. Support vector machine learning for interdependent and structured output spaces. ICML 2004.

[6] Wagstaff, K., Cardie, C., Rogers, S., & Schroedl, S., Constrained k-means clustering with background knowledge. ICML 2001.

[7] Stefano Zanero and Sergio M. Savaresi. Unsupervised learning techniques for an intrusion detection system, ACM March 2004.

[8] Wenke Lee and S. J. Stolfo. Data Mining Approaches for Intrusion Detection, 1998.

[9] Alka Gangrade, Ravindra Patel," Privacy Preserving Two-Layer Decision Tree Classifier for Multiparty Databases",2012.

[10] Yasser Yasami, Saadat Pour Mozaffari "A Novel Unsupervised Classification Approach for Network Anomaly Detection by K Means Clustering and IDs3 Decision Tree Learning Methods", The Journal of Supercomputing, Volume 53 Issue 1, pp. 231 – 245, 2010.

[11] Bart Kuijpers, Vanessa Lemmens, Bart Moelans "Privacy Preserving ID3 over Horizontally, Vertically and Grid Partitioned Data", Theoretical Computer Science, Hasselt University & Transnational University Limburg, Belgium.

[12] Osama Abu Abbas "Comparisons between Data Clustering Algorithms", The International Arab Journal of Information Technology, Vol. 5, No. 3, July 2008.

[13] Cemal Cagatay Bilgin and Bulent Yener "Dynamic Network Evolution: Models, Clustering, Anomaly Detection", Rensselaer Polytechnic Institute, Tech. Rep., 2008

[14] GML AdaBoost Matlab Toolbox. http://goo.gl/vh0R9.

[15] Networks enterprise data acquisition and IP rotation services. http://x5.net.

[16] Quova. http://www.quova.com/.

[17] ToR network status. http://torstatus.blutmagie.de/.

[18] J. D. Brutlag. Aberrant behavior detection in time series for network monitoring. In USENIX Conference on System Administration, 2000.

[19] X. Cai and J. Heidemann. Understanding block-level address usage in the visible Internet. In SIGCOMM, 2010.