International Journal of Electrical and Computer Engineering (IJECE)
Vol. 6, No. 2, April 2016, pp. 778~784
ISSN: 2088-8708, DOI: 10.11591/ijece.v6i2.8909

A Preliminary Performance Evaluation of K-means, KNN and EM Unsupervised Machine Learning Methods for Network Flow Classification

Alhamza Munther[1], Rozmie Razif [1], Mosleh AbuAlhaj[2], Mohammed Anbar[3], Shahrul Nizam[1]
[1]School of Computer and Communication Engineering, Universiti Malaysia Perlis, Perlis, Malaysia
[2]Dept. of Network and Information Security, Faculty of Information Technology, Al-Ahliyya Amman University, Amman, Jordan
[3] National Advanced IPv6 Centre of Excellence, UniversitiSains Malaysia, Penang, Malaysia

Article Info

Article history:

Received Aug 26, 2015
Revised Nov 19, 2015
Accepted Dec 7, 2015

Keyword:

Machine learning
Network traffic classification
Network traffic engineering
Unsupervised learning

ABSTRACT

Unsupervised leaning is a popular method for classify unlabeled dataset i.e. without prior knowledge about data class. Many of unsupervised learning are used to inspect and classify network flow. This paper presents in-deep study for three unsupervised classifiers, namely: K-means, K-nearest neighbor and Expectation maximization. The methodologies and how it's employed to classify network flow are elaborated in details. The three classifiers are evaluated using three significant metrics, which are classification accuracy, classification speed and memory consuming. The K-nearest neighbor introduces better results for accuracy and memory; while K-means announce lowest processing time.

Corresponding Author:

Alhamza Munther,
School of Computer and Communication Engineering,
Universiti Malaysia Perlis, Perlis, Malaysia.
Email: alhamza.munther@gmail.com

1. INTRODUCTION

Network traffic classification occupied a significant role in several fields, such as network security, network management and surveillance… etc. It is the process of classifying network traffic into the original application that generated this traffic. The challenges that face this process is increased because of emerging new applications that caused redoubled size of data [1]. Port-based is one of the first techniques that used in data classification. However, this technique is no longer used since it's easy to masquerade, by using the well-known ports of some applications by other applications. For example, some VoIP applications use port 23 that allocated by IANA to the Telnet protocol [2], [3]. Payload-based and signature-based [4] are two alternative methods that used in data classification. Unfortunately, the two approaches suffer from consuming space of memory and long processing time. In addition, they fail to classify encrypted packets accurately. Behavior-based [5] is another method that used for data classification. However, it fail in real time and online classification evaluation.

As you can see, the aforementioned methods suffer from many problems; which compelled the researches to suggest new approach for data classification; that is, machine learning. Machine learning [6] populates to be a suitable solution since it's powerful of automation, identification, and predication. Basically, machine learning can be classified into supervised and unsupervised learning [7], [8], [9]. Supervised is classify dataset with a prior knowledge about class result. In contrast unsupervised had the potential to classify dataset without knowledge about the resulting class. In this paper, we will evaluate and compare three popular unsupervised classification methods; namely, k-means, k-nearest neighbor, and expectation

maximization. The three methods are evaluated in term of classification accuracy, classification speed and memory consuming.

This paper is organized as follows. Section 2 introduces the three classification methods and shows how to employ each method in data classification. Section 3 evaluates and compares the result of the three methods and discusses the results. Finally, Section 4 presents the conclusion.

2. NETWORK TRAFFIC CLASSIFICATION METHODS

This section reveals the approaches of three unsupervised classification methods and how they employed to classify network flow. The procedure of each one is explained in details to fully understand these methods.

2.1. K-means Clustering

Bernaille et al. [10] proposed using K-means cluster unsupervised learning method that classify network flow by categorizing a dataset into a definite number of clusters (assume *k* clusters) fixed a priori. The key idea is to select *k* centroids randomly, one for each cluster. Each input represented as coordinator by considering the features values which is consisted a group of points, each point is allocated to the closest centroid, and each group of points allocated to a centroid is a cluster the distance is measure. The centroid of each cluster is updated later based on the points allocated to the cluster. Network flows are represented by points in a P-dimensional space (dimension refer to the feature such as packet size), where each packet is linked with a dimension; the coordinate on dimension p is the size of packet p in the flow. The procedure is repeated with updating the steps until no changes clusters, or equally, until the centroids remain the same. Figure 1 shows simply the steps of K-means idea [11].

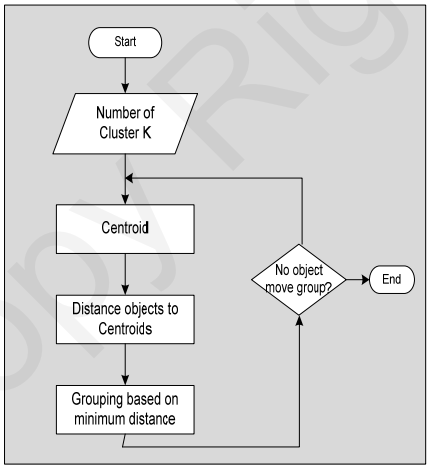


Figure 1. Key steps of K-means method [11]

The similarity between flows is represented by measuring distance between each point in cluster and centroid which is calculate using Euclidean distance as formulated in equation 1. The K-means method attempts to find an optimal solution by reducing the square error, which is defined as in equation 2. The square error is calculated with the distance squared between each point (object) x and the center of its cluster c.

$$dist(x,y)=\left(\sum_{i=1}^{n}(x_i-y_i)^2\right)^{1/2} \quad (1)$$

$$E=\sum_{i=1}^{k}\sum_{j=1}^{n}\left|dist(x_j,y_{i)}\right|^2 \quad (2)$$

Where,
E= objective function
K= number of cluster
n= number of cases (points)
x_i= case i
c_j= centroid for cluster j

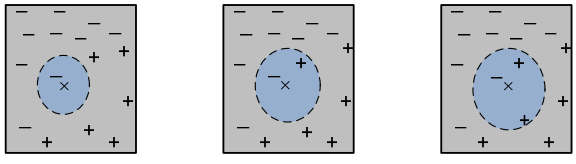
The results illustration that more than 80% of entire flows are correctly classified for a number of applications. One exceptional case is the POP3 application. The classifier labels 86% of POP3 flows as NNTP and 12.6% as SMTP, because POP3 flows always belong to clusters [12]. However, this method is failed to classify some of application with low accuracy; furthermore, the main weakness is that the initial partitions (clusters) are very important. If the initial clusters are not well selected then the K-Means can converge to a local minimum instead of the global minimum solution. To avoid that, a solution is to run the algorithm several times and preserve the best solution. This was led for emerge two issues computationally expensive and extra time of processing [13].

2.2. K-Nearest Neighbor

Roughan et al [14] suggested k-nearest neighbor to classify network traffic. K-nearest neighbor is type of common method called instance-based learning (IBL), which uses specific training instances to make classifications without having to build model from training data. IBL algorithms require a proximity measure to determine the similarity or distance between data inputs (instances) and a classification function that returns the resulted class of a test instance based on its proximity to other instances. A nearest neighbor's classifier represents each instance as a data point in a d-dimensional space, where d is the number of attributes. For a given test instance, we compute it proximity to the rest of the data points in the training set by measuring distance between the instance and class. The k-nearest neighbors for instance r denote to the k points that are closest to r. For an example figure 2 demonstrates the 1-, 2-, 3- nearest neighbors of a data point located at the center of each circle. The data point is predicated based on the class labels of its neighbors. In the case where the neighbors have more than one class, the data point is assigned based on the majority class of its nearest neighbors. In figure 2a, the 1-nearest neighbor of the data point is a negative instance. Therefore the data point is assigned to the negative class. If the number of nearest neighbors is three, as shown in Figure 2c, then the neighborhood contains two positive samples and one negative sample. Based on the majority voting scheme, the data point is allocated to the positive class. K-nearest neighbor computes the similarity by measuring the distance between each test instance point $r = (x, y)$ and all the training instances $(x, y) \in D$ where (D represent whole dataset) to compute its nearest neighbor list. Commonly, there are different ways to compute the distance between point and neighbor class for continuous features such as Euclidean, Manhattan, Minkowski and formulated in equations 3, 4 and 5 respectively, for discrete features using hamming distance which is implement XOR between points.

$$Manhattan\,Dist = \sum_{i=1}^{k} |x_i - y_i| \quad (3)$$

$$Minkowski\,Dist = \left[\sum_{i=1}^{k} (|x_i - y_{i|})^q \right]^{1/q} \quad (4)$$



(a) (b) (c)

Figure 2. 1-2-3-Nearest Neighbor [15]

Generally KNN possess some limitations. At first, it needs to determine the neighbors list for each instances such computation can be costly if the training dataset is large. In addition, k value is sensitive for

choosing. In other word, if dataset k is too small the nearest neighbor classifier may be susceptible to overfitting because of noise in the training data. On the other hand, if k is too large, the nearest neighbor classifier may misclassify the test instance because listing of nearest neighbors may include points that are located far away from its neighborhood as shown in figure 3.

2.3. Expectation Maximization

Jeffrey Erman*et al.* [16] is employed expectation maximization (EM) unsupervised machine learning method to classify network traffic according to the application. EM is an iterative procedure that converges to a maximum likelihood using posteriori probability function. EM works based on two steps. In first step, EM expects the calculation of the cluster probabilities (i.e. expected class values) therefore, this step described as "expectation". In second step, EM calculates of the distribution parameters, is "maximization" of the likelihood of the distribution given the data. Figure 4 shows EM iteration alternatives between performing an expectation (E) step, which produces a function for the expectation of the likelihood calculated using the two estimate parameters means μ and variance σ^2 of points, and a maximization (M) step, which computes the maximum parameters for expected likelihood that found it in the step (E).

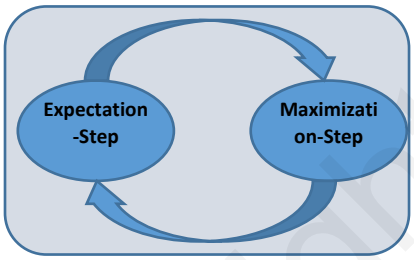


Figure 4. Life cycle of expectation-maximization

To estimate the probability for each class (application type) C for a given certain features-vector x using posterior probability function as used in equation 6 for Naïve Bayse method. The maximum likelihood is calculated by re-estimate the value of mean μ and variance σ^2 continuity then substituted again in the conditional probability function $P(X|C)$ is calculate using the below formula, where i number of instance in each feature x. The authors used 200 iteration as conditional to stop EM loop.

$$P(X|C) = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{-(x-\mu)^2}{2}} \quad (5)$$

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i \quad (6)$$

$$\sigma 2 = \frac{1}{N}\sum_{i=1}^{N} (x_i - \mu) \quad (7)$$

Authors used nine different classes in the experiment namely (SMTP, HTTP, DNS, FTP-CONTROL, Socks, IRC, POP3, FTP-DATA and P2P LimeWire). The overall classification accuracy was 91% for the collected dataset. However, the iteration process consume resources (Memory, CPU) and adding extra processing time where repeated the parameters (means and covariance) calculation up to 200 times [16].

3. PERFORMANCE EVALUATION

In this section, the performance of the K-means clustering, K-Nearest neighbor and expectation maximization methods is evaluated and compared. We have evaluated and compared the three methods using three factors; namely, classification accuracy, classification speed, and memory consumption. We used these

three factors in the comparison because they are play a significant role in real time and online classification environment.

3.1. Testing Environment

The Weka software version 3.7.10 [17] and the Moore dataset [18] are used to evaluate and compare the three classification methods; namely, K-means clustering, K-Nearest neighbor and expectation maximization. The Moore dataset consists of 24863 instances, 248 attributes, and 11 classes, which are WWW, FTP-CONTROL, MAIL, FTP-PASV, P2P, ATTACK, FTP-DATA, DATABASE, SERVICES, MULTIMEDIA, and INTERACTIVE. 14918 out of 24863 records are used as a training dataset while the remaining dataset, 9945 records, are adopted as testing data.

3.2. Results and Discussion

The classification accuracy is evaluated by testing the overall accuracy through determine correctly and incorrectly classified instances. Figure 5 shows the overall accuracy of the three classification methods. The result showed that the K-Nearest neighbor (KNN) (using three neighbors) achieved the highest accuracy by up to 98%, expectation maximization (EM) achieved the second highest accuracy by up to 91%, and K-means achieved the lowest accuracy by up to 80 %. Figure 6 shows the total processing time of the three classification methods including the buildup time. The result showed that the total processing time of EM, KNN, and K-means is 900, 350, and 60 seconds, respectively. As you can see, K-means achieved the best processing time between the three methods, which make it a suitable solution for online classification. Figure 7 shows the memory consumption of the three classification methods. The result showed that the memory consumption of EM, KNN, and K-means is 223MB, 60MB, and 130MB, respectively.

The results showed KNN with 3-Nearst neighbors is the best in term of accuracy and memory consumption due to the powerful and low complexity of method but the memory and processing time is threaten to increase in case number of neighbors increase. K-means was the best in term of time consumption this aspect make it anappropriate solution for real time classification but still needs enhancement with regard to accuracy and memory consumption where data traffic is pumped in high rate in real time and online environment [19]. Expectation-Maximization ranked in the end due to the cost computation was led to high memory and time consuming.

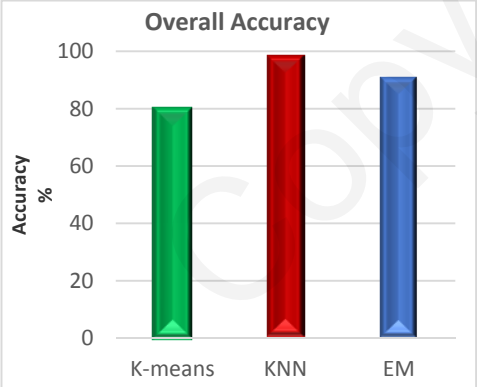


Figure 5. Overall Accuracy of k-means, KNN, and EM

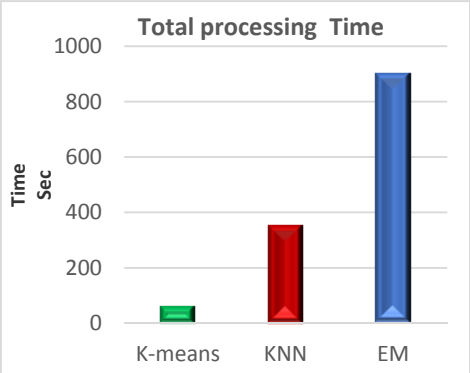


Figure 6. Total processing time of k-means, KNN, and EM

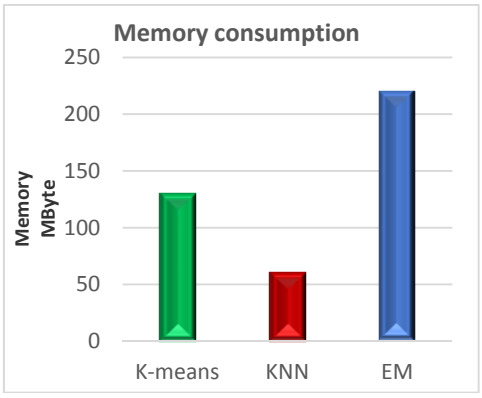


Figure 7. Memory consumption of k-means, KNN, and EM

4. CONCLUSION

Unsupervised learning is known method used widely for identify and classify network traffic. There are several unsupervised classifiers were proposed by researchers to classify network flow. These researchers are competed in term of QoS to test which classifiers are more suitable for real time and online classification. This paper presents a comparative study for three popular unsupervised classifiers namely K-means, K-nearest neighbor (KNN) and Expectation Maximization (EM). These classifiers were studied deeply through explain the methodologies for each and how were employed to classify network flow. The classifiers are evaluated with regard to three significant metrics spatially for real time and online environment. These metrics are classification accuracy, classification speed and memory consumption. As a result KNN was the best in term of accuracy and memory consuming but k-means introduced better performance with regard to total time of processing while expectation maximization was the worst for the three metrics. Based on the generated results we recommend to study the avenues to optimize KNN to reduce time processing to be fit with real time and online environment. Furthermore, we recommend enhancing classification accuracy and decreasing memory consumption for K-means. Thereafter, implement both on huge dataset.

REFERENCES

[1] A. Callado, C. Kamienski, S.N. Fernandes, and D. Sadok, "A Survey on Internet Traffic Identification and Classification", 2009.
[2] IANA. Internet Assigned Numbers Authority. Available: http://www.iana.org/assignments/port-numbers.
[3] L. Bernaille, R. Teixeira, and K. Salamatian, "Early application identification", in *Proceedings of the 2006 ACM CoNEXT conference*, 2006, p. 6.
[4] P. Haffner, S. Sen, O. Spatscheck, and D. Wang, "ACAS: automated construction of application signatures", in *Proceedings of the 2005 ACM SIGCOMM workshop on Mining network data*, 2005, pp. 197-202.
[5] H.T. Marques Nt, L.C. Rocha, P.H. Guerra, J.M. Almeida, W. Meira Jr, and V.A. Almeida, "Characterizing broadband user behavior", in *Proceedings of the 2004 ACM workshop on Next-generation residential broadband challenges*, 2004, pp. 11-18.
[6] Z. Shi, *Principles of machine learning*: International Academic Publishers, 1992.
[7] T.M. Mitchell, "Machine learning. 1997", *Burr Ridge, IL: McGraw Hill*, vol. 45, 1997.
[8] A. Munther, R. Razif, S. Nizam, N. Sabri, and M. Anbar, "Active Build-Model Random Forest Method for Network Traffic Classification", *International Journal of Engineering & Technology (0975-4024)*, vol. 6, 2014.
[9] Y. Wang and Q. Li, "Review on the Studies and Advances of Machine Learning Approaches", *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 12, pp. 1487-1494, 2014.
[10] L. Bernaille, R. Teixeira, I. Akodkenou, A. Soule, and K. Salamatian, "Traffic classification on the fly", *ACM SIGCOMM Computer Communication Review*, vol. 36, pp. 23-26, 2006.
[11] T. Pang-Ning, M. Steinbach, and V. Kumar, "Introduction to data mining", in *Library of Congress*, 2006, p. 74.
[12] T.T. Nguyen and G. Armitage, "A survey of techniques for internet traffic classification using machine learning", *Communications Surveys & Tutorials, IEEE*, vol. 10, pp. 56-76, 2008.
[13] L. Bernaille, A. Soule, M.I. Jeannin, and K. Salamatian, "Blind application recognition through behavioral classification", ed, 2005.
[14] M. Roughan, S. Sen, O. Spatscheck, and N. Duffield, "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*, 2004, pp. 135-148.
[15] J. Han and M. Kamber, "Data mining: concepts and techniques (the Morgan Kaufmann Series in data management systems)", 2000.

[16] J. Erman, A. Mahanti, and M. Arlitt, "Qrp05-4: Internet traffic identification using machine learning", in *Global Telecommunications Conference, 2006. GLOBECOM'06. IEEE*, 2006, pp. 1-6.
[17] I.H. Witten, E. Frank, L.E. Trigg, M.A. Hall, G. Holmes, and S.J. Cunningham, "Weka: Practical machine learning tools and techniques with Java implementations", 1999.
[18] A.W. Moore and K. Papagiannaki, "Toward the accurate identification of network applications", in *Passive and Active Network Measurement*, ed: Springer, 2005, pp. 41-54.
[19] Y. Zhao, X. Xie, and M. Jiang, "Hierarchical Real-time Network Traffic Classification Based on ECOC", *TELKOMNIKA Indonesian Journal of Electrical Engineering*, vol. 12, pp. 1551-1560, 2014.

BIOGRAPHIES OF AUTHORS

Alhamza Munther is a Ph.D. student in School of Computer and Communication Engineering at Universiti Malaysia Perlis, Malaysia. He received his Master degree in advanced computer networks from Universiti Sains Malaysia (USM) in 2012 and B.Sc of Computer and Software Engineering at University of Technology in 2003. His research focuses on overlay networks, multimedia distribution, traffic engineering and machine learning.

Rozmie R. Othman obtained his BEng degree in Electronics (Computer) from Multimedia University, Malaysia in 2006, MEng in Telecommunications Engineering from Universiti Malaya, Malaysia in 2009 and PhD in Software Engineering from University Sains Malaysia in 2012. He is currently a senior lecturer attached to the Embedded, Networks and Advanced Computing Research Group (ENAC), School of Computer and Communication Engineering, Universiti Malaysia Perlis. His main research interest includes Software Engineering, Combinatorial Software Testing, and Algorithm Design. He is a member of IEEE, British Computer Society (BCS), and Board of Engineer Malaysia (BEM).

Dr. Mosleh M. Abu-Alhaj is a senior lecturer in Al-Ahliyya Amman University. He received his first degree in Computer Science from Philadelphia University, Jordan, in July 2004, master degree in Computer Information System from the ArabAcademy for Banking and Financial Sciences, Jordan in July 2007, and doctorate degree in Multimedia Networks Protocols from UniversitiSainsMalaysia in 2011. His research area of interest includes VoIP, Multimedia Networking, and Congestion Control. Apart from research, Dr. Mosleh M. Abu-Alhaj also does consultancy services in the above research areas and directs the Cisco academy team at Al-Ahliyya Amman University.

Dr. Mohammed Anbar holds a PhD in the area of Advanced Computer Networks, M.Sc. in Information Technology, and B.Sc. in Computer System Engineering. His research interest includes Malware Detection, Web Security, Intrusion Detection System (IDS), Intrusion Prevention System (IPS) and network monitoring.

Dr. Shahrul Nizam is a member of the teaching staff at the School of Computer and Communication Engineering, University Malaysia Perlis. He obtained his BSc. degree from University Technology Malaysia in 2004, his M.Sc. in Computer Engineering from University Malaysia Perlis in 2006 and Ph.D. Degree in Computer system Eng. at University of South Australia in 2010. His research areas of interest include the artificial intelligence and machine learning systems. At the moment his is focusing on the agent learning algorithm architecture design for both single and multi-agent systems.