ISSN XXXX XXXX © 2019 IJESC

Research Article Volume 9 Issue No.2

Network Intrusion Detection with Unlabeled Data using Unsupervised Clustering Approach

Uttam Kumar Dey[1], Mohammad Alauddin[2], Tanzillah Wahid[3]
Senior Lecturer[1, 3], Lecturer[2]
Department of Computer Science & Engineering
Uttara University, Dhaka, Bangladesh

Abstract:
Intrusions cause a serious security threat in a network environment and therefore need to be quickly detected and dealt with. Network Intrusion Detection Systems (NIDS) train to identify attacks or malicious activity in a network with a high detection rate while maintaining a low false alarm rate. New intrusion types, of which detection systems may not even be aware are difficult to detect. Signature based methods and misuse detection methods, which rely on labeled data to train, can detect previously known attacks with good accuracy but are not capable of detecting new types of attacks. Anomaly detection techniques can make use of unsupervised learning methods to identify new emerging threats with unlabeled data with a prospective false alarm rate. We reviewed different network intrusion detection methods and present here a comparative study with more emphasis on the unsupervised learning methods for anomaly detection. The K- Means and Expectation Maximization (EM) clustering algorithm were chosen to evaluate the performance of an unsupervised learning method for anomaly detection using the NSL-KDD network data set. The results of the evaluation assert that Expectation Maximization (EM) algorithm outperformed K-Means clustering algorithm with a high detection rate while maintaining a low false alarm rate.

Keywords: Intrusion Detection, Expectation Maximization (EM) Clustering, K-Means Clustering, Machine Learning.

I. INTRODUCTION

An intrusion is defined to be a violation of the security policy of the system; intrusion detection thus refers to the mechanisms that are developed to detect violations of system security policy [1]. A network intrusion is any type of attack or malicious activity that can compromise the stability or security of a network environment. It also compromises the security goals, namely confidentiality, integrity, and availability of a computing and networking resources. Network intrusions keep increasing over the years with new emerging and complex threats. This new emerging threats are the most difficult to identify. A network intrusion detection system (NIDS) is one which automatically scans the network activities attempt to detect the intrusions or attacks. Once attacks are detected, the system administrator may be alerted and thus take appropriate actions. Conventionally, signature-based automatic detection methods have been used for network intrusion detection .This method extracts features from the network data, and detects intrusion using a preset hard coded algorithm provided by human experts. Indubitably, such methods cannot adopt to new types of intrusion quickly, and the present algorithm has to be manually updated for each new type of attack that is detected. Other approaches use machine learning algorithm to train on labeled data. This approach is called misuse detection. Misuse detection is a model-based supervised method which train a classifier with labeled data to classify new unlabeled data. However, more often we don't have labeled data available. Usually, we must deal with vary large volumes of network data, and thus it is very difficult and tedious to classify it manually. We can obtain labeled data by simulating intrusions, but then we would be limited to the set of known attacks that we were able to simulate and new types of attacks occurring in the future will not be reflected in the training data. Therefore, in the end our intrusion detection system will not be able to detect new attacks. To solve these difficulties, we need a technique for detecting intrusions when our training data is unlabeled, as well as for detecting new and unknown types of intrusions. A method that offers potentiality in this task is anomaly detection. Anomaly detection detects anomalies in the data (i.e. data instances in the data that deviate from normal or regular ones) [2]. It also allows us to detect new types of intrusions, because these new types, by assumption are deviations from the normal network usage, just like the other intrusions. Anomaly detection approaches can make use of supervised [3] or unsupervised methods to detect abnormal behaviors in patterns. The main objective of this study is to ensure the advantage of anomaly detection for intrusion detection using unsupervised clustering algorithms over the NSL-KDD network dataset.

II. RELATED WORK

Intrusion detection is presumably the most well-known application of anomaly detection. Various techniques for modeling normal and anomalous traffic have been developed for intrusion detection. A survey of these techniques is given in [4]. They have done a study using k-NN and one-class SVM to detect intrusion on the same dataset. Lazarevic et al. [5] used k-NN, LOF, PCA and unsupervised SVM for intrusion detection. They compared the performance of these algorithms using the KDD-Cup99 dataset. Ding etal. [6] Used a k-NN classifier, SVDD, k-means and a GMM for detecting anomalies in ten different datasets. Yousef et al [7] used algorithms namely Random Forest, Naive Bayes, K-means and Support Vector Machine to identify four types of attacks. They also proposed best feature selection method. They concluded that the Random Forest Classifier (RFC) outperforms the other methods. They have mentioned that hierarchical clustering method can be used to improve the performance.

III. CLUSTERING ALGORITHMS

A. *K- Means Clustering Algorithm*

K-Means clustering is an unsupervised learning algorithm that finds a fixed number (k) of clusters in a set of unlabeled data. A cluster is a group of data points that are grouped together based on their features similarities. In K-Means algorithm, a cluster is defined by a centroid, which is a point (either real or imaginary) at the center of a cluster. Each and every point in a data set is part of the cluster whose centroid is most closely located. In summary, K-Means finds k number of centroids, and then assigns all data points to the closest cluster, with the aim of keeping the centroids small. The common steps for the K means algorithms were the following:

1. Choose number of clusers(K)
2. Initialize centroids (Choose random K points from data set)
3. Calculate the distance from each instance to all centroids using Euclidean distance method
4. Assign each instance to the closest centroid
5. Calculate means of each clusters to be its new centroid.
6. Repeat step 3-5 until the stopping criteria is met(no instances move to another cluster).

B. *Expectation Maximization (EM) Clustering Algorithm*

Expectation Maximization (EM) clustering is a variant of k-means clustering and is widely used to estimate the density of data points in an unsupervised clustering [8]. In the EM clustering, we use an EM algorithm to determine the parameters that maximize the probability of the data, assuming that the data is generated from k normal distributions. The algorithm learns both the means and the covariance of the normal distributions. This method requires several inputs which are the data set, the maximum error tolerance, the total number of clusters, and the maximum number of iteration. The EM can be divided into two vital steps which are Expectation (E-step) and Maximization (M-step). The goal of E-step is to calculate the expectation of the likelihood (the cluster probabilities) for every instance within the dataset and then re-label the instances supported their probability estimations. The M-step is employed to re-estimate the parameters values from the E-step results. The outputs of M-step (the parameters values) are then used as inputs for the subsequent E-step. These two processes are performed iteratively until the results convergence. The mathematical formulas of EM clustering are described in [8][9] and the pseudo codes can be found in [9].

IV. EXPERIMENTAL SETUP

This section describes the intrusion data sets used in the experiment, the software used and the performance metric used to evaluate the performance of two clustering algorithms and the experimental settings and its results.

A. *NSL- KDD dataset for anomaly dectection evaluation*

NSL-KDD dataset was chosen for this work [10]. The NSL-KDD data set is an improvement of the old KDD CUP99 data set. It has been improved by removing some of the redundant data points that could cause errors and give better results than what should be [10]. The data set has atotalof42 different features whether it is the anomaly or the multiclass data set. These are:

duration, protocol_type, service, flag, src_bytes, dst_bytes, land, wrong_fragment, urgent, hot, num_failed_logins, logged_in, num_compromised, root_shell, su_attempted, num_root, num_file_creations, num_shells, num_ access_files, num_outbound_cmds, is_host_login, is_guest_login, count, srv_count, serror_rate, srv_serror_rate, rerror_rate, srv_rerror_rate, same_srv_rate, diff_srv_rate, srv_diff_host_rate, dst_host_count, dst_host_srv_count, dst_host_same_srv_rate, dst_host_diff_srv_rate, dst_host_same_src_port_rate, dst_host_srv_diff_host_rate, dst_host_serror_rate, dst_host_srv_serror_rate, dst_host_rerror_rate, dst_host_srv_rerror_rate, attacks.

B. *Software*

Weka machine learning tool was used with some external Java based libraries for the implementation purposes. The Weka workbench is a open source software providing a collection of machine learning algorithms and data pre-processing tools.[11].

C. *Performance Matric*

To evaluate our system, the following criteria was used: detection rate and false alarm rate. The detection rate is defined as the number of attacks detected divided by the total number of attacks. The false alarm rate is defined as the number of 'normal' patterns classify as attacks divided by the total number of 'normal' patterns. The labels of the patterns were used for this evaluation, but never used for the clustering procedure. We used detection rate and false alarm rate as the performance criteria based on the following metric shown in Table 1 below.

Table .1. Confusion Matrix

		Predicted Class	
		Anomaly	***Normal***
Actual Class	Anomaly	TP	FN
	Normal	FP	TN

Here,
TP – Classified as Normal while they actually were Normal.
TN – Classified as Attack while they actually were Attack
FP – Classified as Attack while they actually were Normal
FN – Classified as Normal while they actually were Attack

The detection rate and false positive rate are calculated using following formula:

$$TPR = \frac{TP}{TP + FN} \quad (1)$$

$$FPR = \frac{FP}{TN + FP} \quad (2)$$

V. EXPERIMENTAL RESULTS

We use NSL-KDD test dataset [10] to evaluate the performance of K-Means and EM algorithms. One of our assumptions is that the training set represents only normal activities where attacks are rare and most of the data represents normal operations. Compare to training dataset, test data set contains more attacks data. Therefore, we believe using test dataset will show better detection accuracy by these two algorithms.

A. *Experiment One: For Binary class(Anomaly and Normal)*

The detection accuracy of EM and K-Means clustering algorithms are shown in Table 2. These clustering algorithms are able to detect intrusions without using prior experience. In

this experiment, the EM algorithm accomplished best accuracy rate with 65.85% compare to K-Means algorithm with 54.69%.

Table.2.Intrusion Detection Accuracy Using Clustering Algorithms.

Algorithm	Detection Rate	False Positive Rate
K-Means	54.69%	22.41%
EM	65.85%	13.20%

In figure 1, the bar chart shows the performance comparison of EM and K-Means clustering algorithms in terms of their detection accuracy and false positive rate prediction.

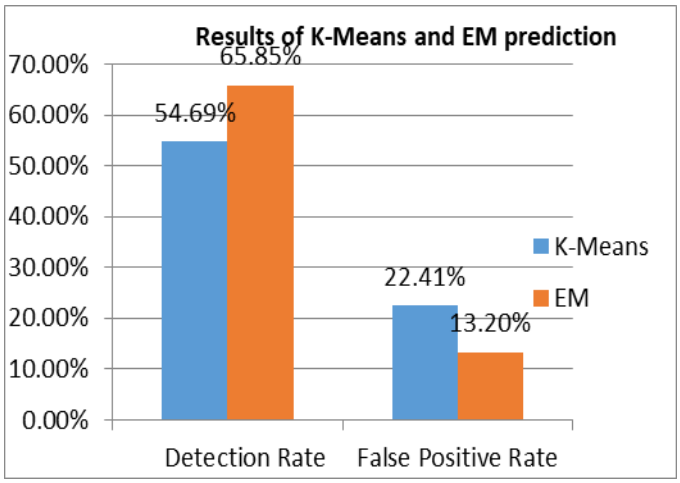

Figure.1. Performance comparison of EM and K-Means clustering algorithms.

B. Experiment Two: For Multi class (DoS, Probing, R2L, U2R)

The NSL-KDD intrusion dataset is classified into four types of intrusion which are DoS, Probing attacks, R2L attacks and U2R attacks. This means that, instead of detecting anomaly or normal, algorithm has to recognize the attack pattern and put them as one of the four types of attack. Table 3 shows the detection accuracy of K-Means clustering algorithm for multi class attacks.

This experiment shows that the K-Means algorithm is able to detect U2R attack with 99.00% accuracy and DoS attacks with 77.37% accuracy. Unfortunately, this algorithm failed to accurately detect R2L attack (25.00%).

One reason is that the R2L attacks have very similar behavior with normal traffics which makes them very difficult to distinguish. Moreover, the number of R2L attacks in intrusion dataset is very small compare to the whole data set. In Figure 2, the line chart shows the prediction made by the K-Means clustering algorithm.

Table.3. Intrusion detection accuracy for multiclass attacks using K-Means clustering algorithm

Algorithm	Attacks			
	DoS	Probe	U2R	R2L
K-Means	77.37%	64.74%	99.00%	25.00%

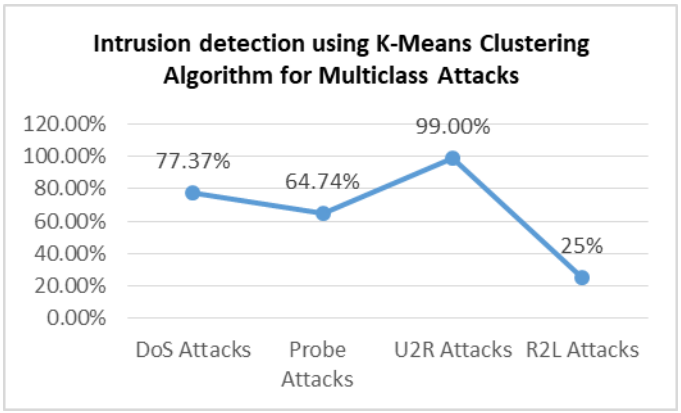

Figure .2. Performance of K-Means clustering algorithm for multiclass attacks detection.

Table 4 shows the detection accuracy of EM clustering algorithm for multi class attacks. This experiment shows that the EM algorithm is able to detect DoS attacks with 43.71% accuracy. Unfortunately, this algorithm failed to accurately detect U2R attack (4.46%). In Figure 3, the line chart shows the prediction made by the clustering algorithm.

Table.4. Intrusion detection accuracy for multiclass attacks using EM clustering algorithm.

Algorithm	Attacks			
	DoS	Probe	U2R	R2L
EM	43.71%	19.86%	4.46%	32.00%

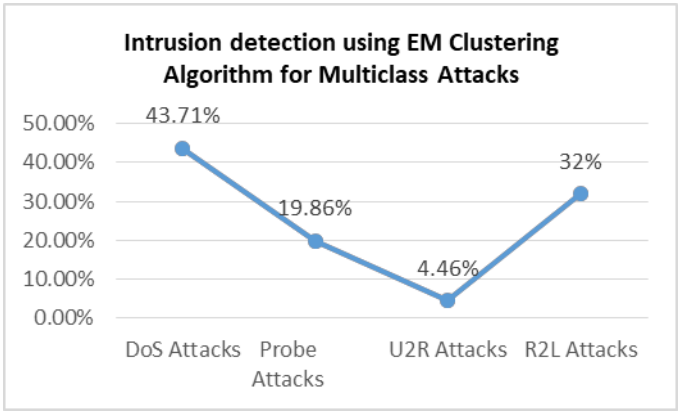

Figure.3. Performance of EM clustering algorithm for multiclass attacks detection.

VI. CONCLUSION

In this work, two unsupervised machine learning algorithms were used for network intrusion detection on NSL-KDD dataset. Our experiment shows that Expectation Maximization (EM) algorithm provided better detection accuracy of 65.85%. Further experiment shows that the K-Means algorithm performs very well in detecting U2R attacks (99.00%) and DoS attacks (77.37%) but it fails to detect R2L attacks (25.00%). But when it comes to detect multiclass attacks, our experiment shows that Expectation Maximization (EM) algorithm performs very poorly. Unfortunately, two algorithms we used in our module for intrusion detection e produces high false positive rate. Therefore, our future work will be focused in reducing the false positive rate and improving the accuracy using deep learning algorithm.

VII. REFERENCES

[1]. S. Chebrolu, A. Abraham, and J. P. Thomas, “Feature deduction and ensemble design of intrusion detection systems”, Computers and Security, 2004.

[2]. L.Portnoy, E. Eskin, S. Stolfo,"Intrusion Detection with Unlabeled Data Using Clustering", In Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001).

[3]. NicoGörnitz, MariusKlof, "Toward Supervised Anomaly Detection", Journal of Artificial Intelligence Research 46 (2013) 235-262.

[4]. Eskin,E., Arnold, A., Prerau,M., Portnoy, L., Stolfo, S.,“A Geometric Framework for Unsupervised Anomaly Detection: Detecting Intrusions in Unlabeled Data”, In Proceedings of the Seventeenth International Conference on Machine Learning. Morgan Kaufmann PublishersInc., pp. 255-262. (2000).

[5]. Lazarevic A, ErtozL,KumarV,OzgurA,Srivastava J. “A Comparative Study of Anomaly Detection Schemes in Network Intrusion Detection”, InProceedings of the Third SIAM International Conference on Data Mining. vol.3. Siam; 2003.p.25–36.

[6]. Ding X, LiY, BelatrecheA,MaguireL P., “An Experimental Evaluation of Novelty Detection Methods”, Neurocomputing. 2014;135:313-327.doi:10.1016/j.neucom.2013.12.002.

[7] A. T. Yousef El, Mourabit, Anouar Bouirden and N. E. Moussaidr, “Intrusion Detection Techniques in Wireless Sensor Network Using Data Mining Algorithms: Comparative Evaluation Based on Attacks Detection”, International Journal of Advanced Computer Science and Applications, vol. 6, no. 9, pp. 164–172, (2015).

[8] Seetha, J., Varadharajan, R., Vaithiyananthan,V.,” Un super vised Learning Algorithm for Color Texture Segmentation Based MultiscaleImage Fusion.”, European Journal of Scientific Research, ISSN 1450-216X, Vol 67, pp. 506-511(2012).

[9]. Lu, Wei., Tong, Hengjian.,” Detecting Network Anomalies Using CUSUM and EM Clustering.“, Advance in Computation and Intelligence, Volume 5821, pp.297-308 (2009).

[10]. A Habibi .L University of New Brunswick. ISCX NSL-KDD dataset, UNB. URL: http://www.unb.ca/ research/ iscx/ dataset/iscx-nsl-kdddataset.html.

[11]. Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, Ian H. Witten, “The WEKA Data Mining Software: An Update,” SIGKDD Explorations, vol. 11, no. 1, 2009.