Network Traffic Pattern Analysis Using Improved Information Theoretic Co-clustering Based Collective Anomaly Detection

Conference Paper · September 2014
DOI: 10.1007/978-3-319-23802-9_17

CITATIONS
9

1 author:

Mohiuddin Ahmed
Canberra Institute of Technology
32 PUBLICATIONS **286** CITATIONS

SEE PROFILE

Network Traffic Pattern Analysis using Improved Information Theoretic Co-clustering based Collective Anomaly Detection

Mohiuddin Ahmed, Abdun Naser Mahmood

School of Engineering and Information Technology,
UNSW Canberra, ACT 2600, Australia.
Mohiuddin.Ahmed@student.adfa.edu.au,Abdun.Mahmood@unsw.edu.au

Abstract. Collective anomaly is a pattern in the data when a group of similar data instances behave anomalously with respect to the entire dataset. Clustering is a useful unsupervised technique to identify the underlying pattern in the data as well as anomaly detection. However, existing clustering based techniques have high false alarm rates and consider individual data instance behaviour for anomaly detection. In this paper, we formulate the problem of detecting DoS (Denial of Service) attacks as collective anomaly detection and propose a mathematically logical criteria for selecting the important traffic attributes for detecting collective anomaly. Information theoretic co-clustering algorithm is advantageous over regular clustering for creating more fine-grained representation of the data, however lacks the ability to handle mixed attribute data. We extend the co-clustering algorithm by incorporating the ability to handle categorical attributes which augments the detection accuracy of DoS attacks in benchmark KDD cup 1999 network traffic dataset than the existing techniques.

Key words: Network Traffic Analysis, Information Theory, Co-clustering, Collective Anomaly Detection, Pattern Mining.

1 Introduction

Internet is a modern day communication platform which provides a diverse range of services. Applications like e-mail, real time video and voice communication, file transfers and storage, web based contents are the most common applications on Internet. Consequently, there is a growing demand for efficient algorithms to detect important trends and anomalies in network traffic data. For example, network managers need to understand user behavior in order to plan network capacity.

One important concern in today's networking environment is Internet security. The network administrators have to handle a large variety of intrusion attempts by individuals with malicious intent [1]. Although research in security domain is growing significantly, the threats are yet to be mitigated. According to Symantec Internet Security Threat Report, 2013 is considered as a year of mega breach

and the size of DoS attacks underwent a rapid increase [2]. Stuxnet stands out of all because of the destructive and malicious behavior, discovered in June 2010 [3]. The technology giant Google was also attacked along with numerous other companies in 2010 [5]. With the increasing number of cyber security experts, the number of individual with detrimental motif is also raising. According to Verizon's Data Breach Investigation Report 2014, 63437 security incidents were carried out by hackers [4]. In addition to that, the expertise required to commit such crimes have decreased due to easily available tools. So, the detection of network attacks has become the highest priority in today's Internet. In this paper, we present a co-clustering scheme for identifying significant traffic flow patterns such as DoS(denial of service) attacks. The contributions in this paper can be summarized as follows-

- We propose a novel framework for the detection of the DoS(denial of service) attacks.
- The characteristics of DoS attacks are analysed and considered as collective anomaly (a group of similar data instances behaving abnormal) unlike the traditional anomaly detection techniques [6–8] where an individual data instance is considered as anomalous.
- We propose a method for selecting the important traffic attributes for detecting collective anomaly.
- Additionally, we extend the co-clustering algorithm [9,10] by incorporating the ability to handle categorical attributes which augments the detection accuracy of DoS attacks in benchmark KDD cup 1999 network traffic dataset [11] than the existing techniques.

The roadmap of this paper is as follows. In Section 2, we describe the anomaly detection and different aspects of it. Section 3 contains the formulation of anomaly detection problem for network traffic analysis and a framework for important network traffic attribute selection. In Section 4, we discuss the information theoretic co-clustering and our proposed extension incorporating the ability to handle categorical data. Section 5 contains the analysis of the experimental results on benchmark KDD cup 1999 intrusion detection dataset [11] and followed by related works in Section 6. We conclude our paper stating the future works in Section 7.

2 Anomaly Detection and Network Traffic Analysis

In this Section we provide a brief discussion on traditional anomaly detection and categories of anomaly. Next, we describe various assumptions of anomaly detection for network traffic analysis.

2.1 Different types of Anomaly

Anomaly detection is an important aspect of data mining, where the main objective is to identify anomalous or unusual data from a given dataset. Anomaly

detection is interesting because it involves automatically discovering interesting and rare patterns from datasets. Anomaly detection has been widely studied in statistics and machine learning, also known as outlier detection, deviation detection, novelty detection, and exception mining [12]. Anomalies are considered to be important because they indicate significant but rare events, and they can prompt critical actions to be taken in a wide range of application domains. For example, an anomaly in an MRI image may indicate the presence of a malignant tumour. Similarly, abnormal behaviour in a credit card transaction could indicate fraudulent activities, an unusual traffic pattern in a network could mean that a computer is hacked or under attack, e.g., using worms and Denial of Service attacks. An important aspect of anomaly detection is the nature of anomaly. Anomalies can be categorized in the following ways.

1. **Point Anomaly:** When a particular data instance deviates from the normal pattern of the dataset, it can be considered as a point anomaly. For a realistic example, we can consider expenditure on car fuel. If the usual car fuel usage of a person is five litres/day but if it becomes fifty litres in any random day then it is a point anomaly.
2. **Contextual Anomaly:** When a data instance is behaving anomalous in a particular context, but not in other context, then it is termed as a contextual anomaly, or conditional anomalies. For example, the expenditure on credit card during a festive period, e.g., Christmas or New Year, is usually higher than the rest of the year. Although, the expenditure during a festive month can be high, it may not be anomalous due to the high expenses being contextually normal in nature. On the other hand, an equally high expense during a non-festive month could be deemed as a contextual anomaly.
3. **Collective Anomaly:** When a collection of similar data instances are behaving anomalously with respect to the entire data set, then this collection is termed as collective anomaly. It might happen that the individual data instance is not an anomaly by itself, but due to its presence in a collection it is identified as an anomaly [13].

2.2 Anomaly in Network Traffic and Assumptions

Reliance on computer network and the increasing connectivity of these networks also raised the probability of damage caused by various types of network attacks. Network attacks, also named as intrusions are difficult to detect and prevent networks, with security policies due to the rapid change of system and applications. Simply, a threat/attack refers to anything which has the detrimental characteristics to compromise a host or network. Poor design of network, carelessness of users, misconfiguration of software or hardware cause the vulnerabilities. According to Kendall et al [14], the attacks can be classified into four major categories discussed below.

– **Denial of Service (DoS)**: It is a type of misuse of the rights to the resources of a network or a host. These are targeted to disrupt the normal computing environment and make the service unavailable. A simple example of DoS attack

is denial of access to a web service to legitimate users, when the server is flooded with numerous connection requests. Performing DoS attack need no prior access to the target and thus considered to be a dreaded attack (Fig. 1 illustrates the DoS attack execution).

Fig. 1: DoS attack execution. The attacker is labelled red and a huge amount of special requests are sent to the server to make the service unavailable to other legitimate users(labelled green), adapted from Internet [15]

- **Probe**: These attacks are used to gather information about a target network or host. More formally, these attacks are used for reconnaissance purpose. The reconnaissance attacks are quite common for gathering information about types and number of machines connected to a network, a host can be attacked to find out the types of softwares installed or application used. The probe attacks are considered as first step of an actual attack to compromise the host or network. There is no specific damage caused by these attacks but considered as serious threat to any corporation because it might give useful information to launch another dreadful attack.
- **User to Root**: When the attacker aims to gain illegal access to administrative account to manipulate or abuse important resources, user to root attacks are launched. To launch such attacks, using social engineering approaches or sniffing password, the attacker access a normal user account. Then exploits some vulnerability to gain the super user privilege.
- **Remote to Local**: When an attacker wants to gain local access as an user of a targeted machine, the R2L attacks are launched. The attacker have the privilege to send packets over network to the target machine. Most commonly the attacker tries hit and trial password guessing by automated scripts, brute

force method etc. There are also some sophisticated attacks where attacker installs a sniffing tool to capture password before penetrating the system.

It is a research challenge to efficiently identify such attacks/intrusions in network traffic to prevent the network from probable damages. Anomaly detection is one such technique to identify abnormal behaviour and analyse further. The basic assumptions for anomaly detection from network traffic are as follows-

- ***Assumption 1:*** '*The majority of the network connections are normal traffic, only a small percentage of traffic are malicious*' [16].
- ***Assumption 2:*** '*The attack traffic is statistically different from normal traffic*' [17].

In recent years, the traditional philosophy of using a knowledge base or external supervision is superseded by unsupervised anomaly detection. Unsupervised anomaly detection techniques are based on purely fundamental topics of data mining such as clustering. Without relying on expert supervision, unsupervised anomaly detection uses clustering techniques to divulge the underlying structure of unlabelled data as well as unknown behaviour. The clustering based anomaly detection follows similar assumptions as below-

- **Premise 1:** We can create clusters of normal data only, subsequently, any new data that do not fit well with existing clusters of normal data are considered as anomalies. For example, density based clustering algorithms do not include noise inside the clusters. Here noise is considered as anomalous.
- **Premise 2:** When a cluster contains both normal and anomaly data, it has been found that normal data lie close to the nearest cluster centroid but anomalies are far away from the centroids [6]. Under this assumption, anomalous events are detected using a distance score. For example, Mohiuddin et al [6] considered an outlier according to a points distance from its centroid. If the distance is a multiple of mean distances of all other data points from the centroid then it is considered as an outlier. Formally, '*an object o in set of n objects is an outlier if the distance between o and the centroid is greater than to p times the mean of the distances between centroid and other objects*'. They also showed that removing outliers from clusters can significantly improve clustering objective function.
- **Premise 3:** In a clustering where there are clusters of various sizes, smaller and sparser can be considered as anomalous and dense clusters can be considered normal. The instances belonging to clusters whose size and/or density is below a threshold are considered as anomalous. Amer et al [7] introduced Local Density Cluster-Based Outlier Factor (LDCOF) which can be considered as a variant of CBLOF [8]. The LDCOF score(4) is calculated as the distance to the nearest large cluster divided by the average distance to the cluster center of the elements in that large cluster. LDCOF score will be **A** when $p \in C_i \in$ SC where $C_j \in$ LC and **B** when $p \in C_i \in$LC.

$$distance_{avg}(C) = \frac{\sum_{i \in C} d(i, C)}{|C|} \tag{1}$$

$$A = \frac{min(d(p, C_j))}{distance_{avg}(C_j)} \tag{2}$$

$$B = \frac{d(p, C_i)}{distance_{avg}(C_i)} \tag{3}$$

$$LDCOF(p) = A \mid B; \tag{4}$$

3 DoS Attack as Collective Anomaly and Relevant Attribute Selection

In this Section, we discuss more about the how DoS attack can be considered as collective anomaly and propose a criteria for finding the relevant attributes for their detection. At first, we look at the data distribution of the benchmark KDD cup 1999 network intrusion dataset to understand the impact of these aforementioned attacks. The rationale behind using this dataset is quite logical in a sense that, the taxonomy used to classify attack types for intrusion detection evaluation is relevant today regardless of the advent of newer attacks. Additionally the scarcity of labelled network traffic datasets for intrusion detection evaluation is a major issue. Although being outdated, this dataset has been used widely for benchmarking purposes [18].

3.1 DoS Attack as Collective Anomaly

Table 1: Class distribution of KDD Cup 1999 dataset [11]

Class	**Full Training Dataset**	**10% of Training Dataset**	**Test Dataset**
Label	**(%age)**	**(%age)**	**(%age)**
DoS	79.28	79.24	73.9
Probe	0.84	0.83	1.34
U2R	0.001	0.01	0.02
R2L	0.023	0.23	5.26
Normal	19.86	19.69	19.48

From Table 1, it is evident that, the Probe, R2L and U2R attack types are quite insignificant in size and traditional machine learning approaches shown poor performance on these rare classes of attack [19, 20]. However, in this paper we are addressing the issue of detecting the DoS attacks as collective anomaly. Following the data distribution of KDD cup 1999 dataset and the anomaly detection assumptions discussed previously (Section 2.2), we observe a complete

mismatch. For network traffic analysis, DoS attacks do not follow these assumptions. Considering the characteristics of DoS attack and the size we can come to a conclusion that, DoS attack can be considered as a group of network traffic instances affecting the network as well as collective anomaly(Section 2.1). DoS attack has few variants and can be classified into two major groups based on their distribution. There are six variants of DoS attack as follows-

- **Back:** It is an attack against the Apache web server.
- **Land:** It is an effective attack where the attacker sends a packet with same source and destination address.
- **Neptune:** This attack makes the TCP/IP implementations vulnerable.
- **POD:** In *ping of death* attack, the size of ICMP packets are longer than 64000 bytes.
- **Smurf:** When there is a large number of 'echo replies' sent to a machine without any 'echo request' can be considered as Smurf attack.
- **Teardrop:** This attack has the ability to exploit the flaws in the implementations of IP fragmentation re-assembly code.

The Table 2 displays the distribution of different types of DoS attack in KDD cup dataset. Considering the volume of DoS attack and their characteristics, we can come to an understanding that, DoS attack cannot be treated as point anomaly and in this regard treating the same as collective anomaly is a better idea for more accurate results.

Table 2: DoS attack distribution of KDD Cup 1999 dataset [11]

Attack	**No. of Instance**
Smurf	280790
Neptune	107201
Back	2203
Teardrop	979
POD	264
Land	21

From the Table 2, it is evident that there exists predominant two classes of DoS attack based on the size. We can also summarize in the way that, clustering these different attacks will result in two classes. Next, we discuss the technique to select the attributes responsible for differentiating DoS from normal traffic.

3.2 Traffic Attribute Selection

In this Section, we investigate the behaviour of DoS and normal traffic attributes. Since, there is a huge amount of network traffic corresponding to DoS attack, we

can consider the similarity in instances and the difference in standard deviation between the normal and DoS instances as important factor for attribute selection. It is mathematically logical that, if there are more similar instance in a group then the difference of standard deviation will be less with the other group with smaller number of similar instances. Lets give an example to show that our hypothesis. Consider the dataset D has two attributes and contains only DoS and normal instances. Now, as we know, DoS attack is a collective anomaly and outnumber the normal instances, we need to identify the attributes which play important role for differentiating DoS and normal instances. Additionally, these huge number of traffic instances are linearly scaled between [0,1] for avoiding the impact of distance function of clustering algorithms. Let A and B be two attributes where A = (1,....,100000) and B = (1,....,100). The range is defined as range(i) = max(i)- min(i), consequently the range(A) = 100000 and range(B) = 100. When Euclidean distance (5) will be used for clustering the dataset, the attribute A will have a greater impact than the attribute B. Table 3 displays the sample dataset where instances corresponding to DoS and normal are labelled.

$$D(C_1, C_2)) = \sqrt{\sum_{i=1}^{d} D_i(C_1, C_2)^2} \tag{5}$$

Table 3: Sample dataset

Label	**A1**	**A2**
DoS	0.11	0.88
DoS	0.33	0.11
DoS	0.44	0.33
DoS	0.55	0.44
DoS	0.88	0.11
Normal	0.11	0.94
Normal	0.94	0.33
Normal	0.22	0.11

In the Table 3, the similarity of instances between DoS and normal for attribute A1 is 1 and for A2 is 2. The standard deviation difference between DoS and normal for attribute A1 is 0.1661 and for A2 is 0.1453. In equation(6) *Sim* indicates the number of similar instances in DoS and normal. In equation(7) *Stdev* indicates the standard deviation (8) and d is the difference.

$$Sim_{A1}(DoS, Normal) < Sim_{A2}(DoS, Normal) \tag{6}$$

$$d(Stdev_{A1}(DoS, Normal)) > d(Stdev_{A2}(DoS, Normal)) \tag{7}$$

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \tag{8}$$

Once we have the similarity and difference in standard deviation for the original dataset, clustering algorithm is applied to find out the underlying pattern from the data with $D(Sim_i, d_i)$, where i=1 to n, the number of attributes in the dataset. Here we propose to apply *x-means* clustering algorithm which is a variant of basic *k-means* and use bayesian information criterion (9) to identify the number of clusters in the data [13]. Where l_j (D) is the log-likelihood of the data according to the j_{th} model and taken at the maximum likelihood point, p_j is the number of parameters in M_j, which refers to a family of alternative models. R refers to the size of dataset D.

$$BIC(M_j) = l_j(D) - \frac{p_j}{2} \times \log R \tag{9}$$

Now, the cluster with lowest similarity and highest difference in standard deviation contains the expected attribute set of the data which can be used for differentiating DoS attack as well as collective anomaly from the normal instances with better accuracy than using the attributes which can exacerbate the detection accuracy. (Section 5.1 includes more details on the experimental analysis)

4 Improved Information Theoretic Co-clustering

In this Section, we briefly discuss about the co-clustering technique at first, then we describe the information theoretic co-clustering framework. Finally, we highlight the issue of mixed attribute dissimilarity measure for co-clustering and integrate such measure for information theoretic co-clustering to improve it's ability to handle network traffic.

4.1 Co-clustering

Co-clustering can be simply considered as a simultaneous clustering of both rows and columns. Co-clustering can produce a set of **c** column clusters of the original columns **C** and a set of **r** row clusters of original row instances **R**. Unlike other clustering algorithms, co-clustering also defines a clustering criterion and then optimizes it. In a nutshell, co-clustering finds out the subsets of rows and columns simultaneously of a data matrix using a specified criterion. Co-clustering has been widely applied in various application domain such as text clustering, gene-microarray analysis, natural language processing and many more [10]. The benefits of co-clustering over the regular clustering are the following-

1. Simultaneous grouping of both rows and columns can provide a more compressed representation and preserve information contained in the original data.
2. Co-clustering can be considered as a dimensionality reduction technique and suitable for creating new features.
3. Significant reduction in computational complexity. For example, traditional *k-means* algorithm has the $\mathcal{O}(mnk)$ as computational complexity where $m=$ number of rows, n = number of columns and k is the number of clusters. But in co-clustering the computational complexity is $\mathcal{O}(mkl + nkl)$, here l is the number of column clusters. Obviously $\mathcal{O}(mnk) \gg \mathcal{O}(mkl + nkl)$.

4.2 Information Theoretic Co-clustering

Information theoretic co-clustering is first proposed by Dhillon [9] where it is modeled as the joint probability distribution. According to their approach an optimal co-clustering confirms the loss minimization of 'Mutual Information' as follows in equation(10).

$$min(I(X;Y) - I(\hat{X};\hat{Y})) \tag{10}$$

Banerjee et al [10] pointed out that the information theoretic co-clustering uses the joint probability distribution which may not be known and calculated from contingency table or co-occurrence matrix. Additionally the data matrix may contain negative entries and distortion measure other than KL-divergence may be more appropriate. Banerjee et al [10] extended the information theoretic co-clustering [9] in three directions as follows-

– Nearness is measured by Bregman divergence.
– Allows multiple co-clustering schemes.
– Generalization of maximum entropy approach.

Bregman co-clustering tries to minimize the information loss in the approximation of a data matrix X, in terms of a predefined bregman divergence function. For a given co-clustering *(R,C)* and a matrix approximation scheme M, a class of random variables which store the characteristics of data matrix X is defined. The objective function tries to minimize the information loss on the approximation of $\widetilde{X}$ for a co-clustering *R,C*. The Bregman information of X can be defined as follows

$$I_\phi(X) = E\left[log\left(\frac{X}{E[X]}\right)\right] \tag{11}$$

Here, the matrix approximation scheme is defined by the expected value and the bregman divergence d_ϕ for a optimal co-clustering as follows

$$(R^*, C^*) = arg\ min\ E[d_\phi(X, \widetilde{X})] \tag{12}$$

Here, d_ϕ, can be considered in two ways as follows.

$$\textbf{I} - \textbf{Divergence} : d_\phi(x_1, x_2) = x_1 log(\frac{x_1}{x_2}) - (x_1 - x_2) \tag{13}$$

$$\textbf{Euclidean Distance} : d_\phi(x_1, x_2) = (x_1 - x_2)^2 \tag{14}$$

4.3 Co-clustering Mixed Attribute Data

Since, we are inspired to use the co-clustering for network traffic analysis, we find that, these co-clustering techniques are not using any nearness measures for mixed attribute data instances such as the data matrix with both categorical and numerical data. However, network traffic instances contain both categorical and numerical data. For example, the protocols of traffic instances are categorical and port numbers are numerical in nature. In this scenario, we incorporate the mixed attribute distance measure for co-clustering network traffic as well as collective anomaly detection. There are various measures for similarity calculation of categorical data [21] but for simplicity we just consider that, the dissimilarity between two data instances is 1 when they mismatch and zero otherwise (15). The following Table 4 displays the concept for the network traffic protocols which are categorical data. As a whole, for numerical data, we simply use the Euclidean distance and for categorical data we consider the similar data instance has distance zero and dissimilar data has distance one(16).

Table 4: Nearness calculation for categorical data [21]

Label	**TCP**	**UDP**	**ICMP**
TCP	0	1	1
UDP	1	0	1
ICMP	1	1	0

$$D(X_k, Y_k) = \begin{cases} 0 \; if \; X_k = Y_k \\ 1 \;\; otherwise \end{cases} \quad [For \; Categorical \; data] \tag{15}$$

$$[Mixed \; Attribute \; Distance \; Measure] \;\; D(X, Y) = \sqrt{\sum_{k=1}^{d} D_k(X_k, Y_k)^2} \tag{16}$$

Consequently the distance between two traffic instances $d_1 = (TCP, 0.11, 0.78)$, $d_2 = (ICMP, 0.33, 0.74)$ will be calculated as (16)
$\sqrt{(TCP - ICMP)^2 + (0.11 - 0.33)^2 + (0.78 - 0.74)^2} = \sqrt{1 + 0.0484 + 0.0016}$
$= 1.02$. As a result, bregman co-clustering is extended for handling dataset with both categorical and numerical data and is suitable for applying on network traffic datasets.

5 Experimental Analysis

As discussed earlier, we use the KDD cup 1999 dataset for the experimental evaluation. The first part of our experiment contains the attribute selection for collective anomaly detection and then we show the effectiveness of improved information theoretic co-clustering for network traffic analysis.

5.1 Attribute Selection from KDD cup 1999 for Collective Anomaly Detection

The KDD cup 1999 dataset has 41 attributes which can be classified into four main groups as the basic, time, host, content features. Since, we are using the normalized data which is linearly scaled between 0 and 1, it is important to use the attributes which has the ability to distinguish DoS attack from normal instances. We calculate the standard deviation of the DoS and normal data instances from labelled data and measure the difference as $d_{stdev}(DoS, Normal)$. Also, the number of similar number of data instances in both category of the data as *Sim(DoS,Normal)*. Then we apply *x-means* algorithm on the dataset as *x-means*($d_{stdev}(DoS, Normal)$, *Sim(DoS,Normal)*). The following Table 5 depicts the results after the clustering and it is clear that, the cluster which has less similar data instances and higher standard deviation will be suitable group of attributes for collective anomaly detection. For the space scarcity, we represent the attributes with the numbers serially.

Table 5: Attribute Selection Results

Attributes	***Sim(DoS,Normal))***	$d_{stdev}(DoS, Normal)$
1-20,23,24,25,26,28,29,35,37	257	255.50
21,22,30,31	1056	111.05
27,32,33,34,36,38,39	536	115.10

5.2 Collective Anomaly Detection using Improved Co-clustering

Once we have the desired attribute set, next we apply the improved information theoretic co-clustering which can handle both the numeric and categorical data. Since, we are focusing on detecting DoS attacks and in Section 3.1 it was discussed that in KDD cup has predominant 2 groups of attack according to size. So, the input row as three for the co-clustering will be appropriate and based on the attributes the number of column clusters will be four. We consider the smaller cluster will be the cluster containing normal instances and the larger clusters as attack clusters or collective anomaly. We measure the accuracy of our approach using the standard confusion metrics. The metrics are listed as True

Positive (TP = Attack correctly identified as attack.), False Positive (FP = Normal traffic incorrectly identified as attack.), True Negative (TN = Normal traffic correctly identified as normal), False Negative (FN = Attack incorrectly identified as normal.). Table 6 displays the possible test outcomes and the confusion metrics.

Table 6: Standard confusion metrics

Actual traffic label	**Normal**	**Attack**
Normal	TN	FP
Attack	FN	TP

The accuracy is computed using equation(17).

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{17}$$

We also consider the *Precision, Recall* and *F-measure* for evaluation. In pattern mining, precision is referred as the fraction of retrieved instances that are relevant, while recall is the fraction of relevant instances that are retrieved. F-measure combines precision and recall as the harmonic mean of precision and recall(18). We also consider cluster purity as another evaluation criterion(19).

$$F - measure = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{18}$$

$$Cluster\ Purity = \frac{Number\ of\ Attack/Normal\ Instances}{Size\ of\ the\ cluster} \tag{19}$$

Table 7 contains the evaluation results of all the metrics discussed above. The experimental results are quite satisfactory, however, it is not worthy to compare our results with other clustering based techniques because the concept of collective anomaly detection is proposed by ourselves for the first time and not considered by others. The closest approach [1] which used co-clustering for network anomaly detection considered only cluster purity for evaluation and only seven numerical attributes for co-clustering. However, our proposed technique outperforms their approach with the cluster purity as well considering both normal and attack cluster purity(Table 8).

Table 7: Evaluation Results

Accuracy	**Precision**	**Recall**	**F-measure**	**Attack Cluster Purity**	**Normal Cluster Purity**
92.82%	**0.9236**	**0.9923**	**0.96**	**92.36%**	**95.6%**

Table 8: Cluster purity comparison

Purity	Our Proposed Technique	Network Anomaly Detection using Co-clustering [1]
Normal	95.6%	75.84%
Attack	92.36%	92.44%

Table 9: Network anomaly detection accuracy using clustering algorithm [23]

Algorithm	Accuracy	False Positive
k-means	57.81%	22.95%
Improves k-means	65.40%	21.52%
k-medoids	76.71%	21.83%
EM clustering	78.06%	20.74%
Distance based Outlier Detection	80.15%	21.14%

6 Related Works

In this Section, we provide a brief description on the existing techniques for network anomaly detection. There are various approaches to deal with the network anomaly detection, however we focus on the clustering based network anomaly detection since our approach is of this category.

To the best of our knowledge, we are the first to propose collective anomaly detection using co-clustering. Although in [1], co-clustering is used for anomaly detection however considered the numerical attributes for co-clustering. So, the comparison is not logical due to different set of attributes. Also, the cluster purity, accuracy of our approach is significantly better than their approach. Portnoy et al [16] proposed clustering based on width to classify data instances. The width is a constant and remains same for all the clusters. Once the clustering is done, based on the assumption that normal instances constitute overwhelmingly large portion of the entire dataset, N percent of clusters are normal and other are anomalous. Using the assumption of Portnoy et al [16], Kingsly et al [22] proposed a density-based and grid-based clustering algorithm which is suitable for unsupervised anomaly detection. Iwan et al [23] described the advantages of using the anomaly detection approach over the misuse detection technique in detecting unknown network intrusions or attacks. It also investigates the performance of various clustering algorithms when applied to anomaly detection (Shown in Table 9). Four different clustering algorithms: *k-means, improved k-means, k-medoids, Expectation Maximization(EM)* clustering and distance-based outlier detection algorithms are used. The anomaly detection module produced high false positive rate (more than 20%) for all four clustering algorithms. None of these techniques considered collective anomaly detection and avoids the vol-

ume issue of DoS attacks, consequently performing poor than our proposed approach.

7 Conclusion

In this paper, we have proposed to solve the network intrusion detection problem using a set of emerging data mining and machine learning techniques. Our contribution includes detection of the DoS attacks due to its volume and detrimental impact on the network. The characteristics of these type of attacks are analysed and considered as collective anomaly unlike the traditional anomaly detection techniques. We propose a method for selecting the traffic attributes responsible for detecting collective anomaly. We also explore the effectiveness of information theoretic co-clustering algorithm which is advantageous over regular clustering for creating more fine-grained representation. Additionally, we extend the co-clustering algorithm by incorporating the ability to handle categorical attributes. Experimental results show that our proposed approach have better results on various evaluation metrics using benchmark KDD cup 1999 network traffic dataset than the existing techniques. In future, we will focus on creating concise and informative network traffic summaries using co-clustering techniques.

References

1. E. E. Papalexakis, A. Beutel, and P. Steenkiste, "Network anomaly detection using co-clustering," in *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*, ser. ASONAM '12. Washington, DC, USA: IEEE Computer Society, 2012, pp. 403–410.
2. Symantec Internet Security Threat Report, http://www.symantec.com/content/en/us/enterprise/
3. Stuxnet, http://www.stuxnet.net/
4. Verizon's Data Breach Investigation Report 2014, http://www.verizonenterprise.com/DBIR/2014/
5. Google hack attack was ultra sophisticated, new details show, http://www.wired.com/2010/01/operation-aurora/
6. M. Ahmed and A. Naser, "A novel approach for outlier detection and clustering improvement," in *Industrial Electronics and Applications (ICIEA), 2013 8th IEEE Conference on*, 2013, pp. 577–582.
7. M. G. Mennatallah Amer, "Nearest-neighbor and clustering based anomaly detection algorithms for rapidminer." Shaker Verlag GmbH, Aachen, 8 2012, pp. 1–12.
8. Z. He, X. Xu, and S. Deng, "Discovering cluster based local outliers," *Pattern Recognition Letters*, vol. 2003, pp. 9–10, 2003.
9. I. S. Dhillon, S. Mallela, and D. S. Modha, "Information-theoretic co-clustering," in *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '03. New York, NY, USA: ACM, 2003, pp. 89–98.

10. A. Banerjee, I. Dhillon, J. Ghosh, S. Merugu, and D. S. Modha, "A generalized maximum entropy approach to bregman co-clustering and matrix approximation," *J. Mach. Learn. Res.*, vol. 8, pp. 1919–1986, Dec. 2007.
11. "1999 kdd cup dataset." [Online]. Available: `www.kdd.ics.uci.edu`
12. M. Ahmed, A. Mahmood, J. Hu, Outlier detection, in: The State of the Art in Intrusion Prevention and Detection, CRC Press, USA, pp. 3–23.
13. M. Ahmed and A. Mahmood, "Network traffic analysis based on collective anomaly detection," in *Industrial Electronics and Applications (ICIEA), 2014 IEEE 9th Conference on*, June 2014, pp. 1141–1146.
14. K. Kendall, "A database of computer attacks for the evaluation of intrusion detection systems," in *DARPA Off-line Intrusion Detection Evaluation, Proceedings of DARPA Information Survivality Conference and Eexposition (DISCEX)*, 1999, pp. 12–26.
15. How to survive botnet attacks - Understanding Botnets and DDOS attacks, https://www.youtube.com
16. L. Portnoy, E. Eskin, and S. Stolfo, "Intrusion detection with unlabeled data using clustering," in *In Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001*, 2001, pp. 5–8.
17. V. A. Javitz H.S, "The nides statistical component: Description and justification," in *Technical Report*, 1993.
18. M. Ahmed, A. N. Mahmood, and M. R. Islam, "A survey of anomaly detection techniques in financial domain," *Future Generation Computer Systems*, 2015.
19. I. Levin, "Kdd-99 classifier learning contest llsoft's results overview," *SIGKDD Explor. Newsl.*, vol. 1, no. 2, pp. 67–75, Jan. 2000.
20. R. Agarwal and M. V. Joshi, "Pnrule: A new framework for learning classifier models in data mining (a case-study in network intrusion detection," IBM Research Report, Computer Science/Mathematics, Tech. Rep., 2000.
21. S. Boriah, V. Chandola, and V. Kumar, "Similarity measures for categorical data: A comparative evaluation," in *In Proceedings of the eighth SIAM International Conference on Data Mining*, pp. 243–254.
22. K. Leung and C. Leckie, "Unsupervised anomaly detection in network intrusion detection using clusters," in *Proceedings of the Twenty-eighth Australasian Conference on Computer Science - Volume 38*, ser. ACSC '05. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2005, pp. 333–342.
23. I. Syarif, A. Prugel-Bennett, and G. Wills, "Unsupervised clustering approach for network anomaly detection," in *Networked Digital Technologies*, ser. Communications in Computer and Information Science, R. Benlamri, Ed. Springer Berlin Heidelberg, 2012, vol. 293, pp. 135–145.