Fuzzy Clustering as an Intrusion Detection Technique

Disha Sharma
Research Scholar
dishasharma210@gmail.com

Abstract

Intrusion detection and clustering have always been hot topics in the field of machine learning. Clustering as an intrusion detection technique has long before proved to be beneficial. But as the methods and types of attacks are changing, there is an ongoing need to develop more and more better techniques that can fight back. The main aim of this paper is to use Fuzzy c-medoids algorithm to intrusion detection. The beginning section of the paper deals with introduction to clustering in the field of intrusion detection while the later section defines how fuzzy k-medoids algorithm performs better than fuzzy c-means algorithm.

1. Intrusion Detection

All Any attempt to compromise the integrity, confidentiality or availability of a resource is called an intrusion. A wide range of activities fall under this definition. Added security measure can stop all such attacks. The goal of intrusion detection is to build a system which would automatically scan network activity and detect such attacks. Once an attack is detected, the system administrator could be informed and thus take corrective action.

Generally, there are four categories of attacks [1]. They are:

1. DoS (Denial of Service) – trying to prevent a legitimate user from accessing the service in the target machine.

2. Probe – scanning a target machine for information about potential vulnerabilities.

3. R2L (Remote to Local) – when attacker attempts to obtain non-authorized access into a machine or network.

4. U2R (User to Root) – when target machine is already invaded, but the attacker attempts to gain access with super-user privileges.

The rapid proliferation of computer networks has changed the prospects of network security. This generated a need of a system that can detect threats to the network instead of simply relying on intrusion prevention systems. Detecting such threats not only provides information on damage assessment, but also helps to prevent future attacks. These attacks are usually detected by tools referred to as Intrusion detection system. Researchers have developed intrusion detection system for various environments depending upon the security concerns of different networks. The function of Intrusion Detection System is to gather and analyze information from various areas within a computer or a network to determine all possible security breaches.

An Intrusion Detection System (IDS) is software and/or hardware designed to detect unwanted attempts at accessing, manipulating, and/or disabling of computer system, mainly through a network, such as internet. IDSs are proposed to improve computer security because it is not feasible to build completely secure systems. In particular, IDSs are used to identify, assess, and report unauthorized or unapproved network activities so that appropriate actions may be taken to prevent any future damage. Intrusion detection systems can be of two types: *signature based* and *anomaly based*. Signature detection systems are based on pattern matching i.e. they try to match the scenarios with already recorded signatures from the database while anomaly detection techniques compare the behaviour of data creating a baseline profile of the normal system, any deviation from the normal data is considered to be an anomaly.

Both the approaches have their own positive and negative sides. As in Signature detection the known attacks can be detected reliably with low false positive rates but the major drawback is that such systems

require a timely and continuously updated database of signatures for all possible attacks against a network. On the other hand, anomaly detection has two major advantages over signature detection. First, the ability to detect unknown attacks as well as "zero day" attacks. Second, every system/network has its own customized profiles of normal activity, which makes it difficult for the attacker to know with confidence what activities can be carried out without getting detected. A better performance of Intrusion Detection System can be achieved by combining the plus points of both misuse and anomaly detection approaches and trying to remove the drawbacks of both the approaches.

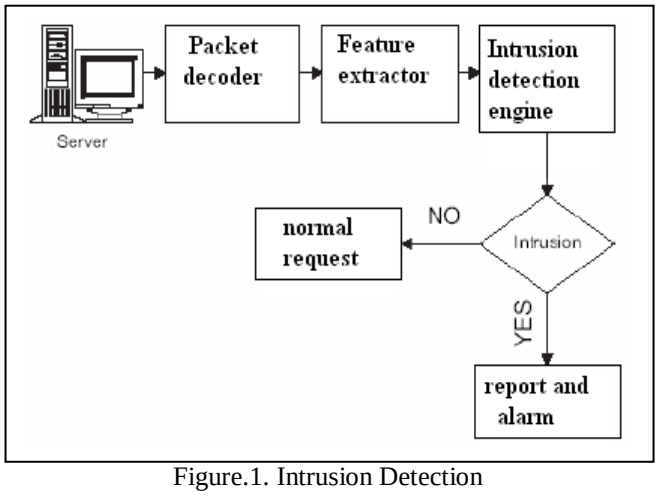

Figure.1. Intrusion Detection

.

2. Data Mining

The term data mining is frequently used to designate the process of extracting useful information from large databases. Here we adopt a slightly different definition, which is similar to the one expressed by Fayyad et al. In this view, the term knowledge discovery in databases (KDD) is used to denote the process of extracting useful knowledge from large data sets. Data mining, by contrast, refers to one particular step in this process. Specifically, the data mining step applies data mining techniques to extract patterns from the data. Additionally, it is preceded and followed by other KDD steps, which ensure that the extracted patterns actually correspond to useful knowledge.

There has been some confusion about how data mining relates to the fields machine learning and statistics. Data mining extensively uses known techniques from machine learning, statistics, and other fields. Nevertheless, several differences between data mining and related fields have been identified. One of the most frequently cited characteristics of data mining is its focus on finding relatively simple, but interpretable models in an efficient and scalable manner. In other words, data mining emphasizes the efficient discovery of simple, but understandable models that can be interpreted as interesting or useful knowledge.

Data mining techniques basically correspond to pattern discovery algorithms, but most of them are drawn from related fields like machine learning or pattern recognition. In context to intrusion detection following data mining techniques [2] are widely used:-

- *Association rules* – defines the normal activity by determining attribute correlation or relationships among items in dataset which makes discovery of anomalies becomes easy.
- *Frequent Episode rules* – describes the audit data relationship using the occurrence of the data.
- *Classification* – classifies the data into one of the available categories of data as either normal data or one of the types of attacks.
- *Clustering* – clusters the data into groups with the property of inter-group similarity and intra-group dissimilarity.
- *Characterization* – differentiates the data, further used for deviation analysis.

An important problem of Intrusion Detection is how to effectively divide the normal behavior and the abnormal behavior from a large number of raw data's attributes, and how to effectively generate automatic intrusion rules after collected raw network data. If the network is small and signatures are kept up to date, then an analyst can observe all alarms and can determine the type of attack, if it is a new or old type of attack.

But as the network grows and becomes more complex, the human analyst will be overwhelmed with all alarms produced by the system daily and here comes the question: is it possible to automate the process so that it can detect new intrusions without the help of a human analyst? The answer is yes, using Data Mining.
The data mining technology was first applied in the intrusion detection research area by Lee and Salvatore J.Stolfo, Columbia University Wenke. Its idea is: Through analysis mining of the network data and the host call data discover misusing detection rule or exception detection model. Applying the data mining technology in the intrusion detection can widely audit

the data to obtain the model, thus it enables you to catch the actual invasion and the normal behavior pattern precisely. This automatic method does not need the manual analysis any more and the coding intrusion pattern, and when building normal skeleton, it no longer choose the statistical method by experience as before. The main benefit is that the same data mining tool may apply in many data streams, so it is advantageous to build strong intrusion detection system.

3. Clustering

By definition cluster analysis is art of finding groups in data. Clustering is the classification of similar objects into different groups or more precisely, partitioning of data into subset (clusters) so that data in each subset (ideally) share some common trait often proximity according to some defined distance measure. The process of grouping a set of physical or abstract objects into classes of similar objects is called clustering. A cluster is a collection of data objects that are similar to one another within the same cluster and are dissimilar to the objects in other clusters. By clustering, one can identify dense and sparse regions and therefore, discover overall distribution patterns and interesting correlations among data attributes.

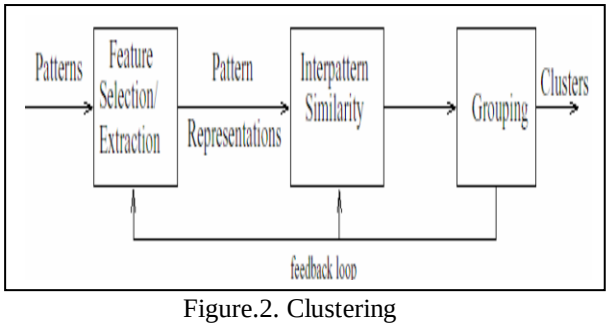

Figure.2. Clustering

A clustering algorithm attempts to find natural groups of components (or data) based on some similarity. Also, the clustering algorithm finds the centroid of a group of data sets. To determine cluster membership, most algorithms evaluate the distance between a point and the cluster centroid. The output from a clustering algorithm is basically a statistical description of the cluster centroid with the number of components in each cluster.

The main advantage that clustering provides is the ability to learn from and detect intrusions in the audit data, while not requiring the system administrator to provide explicit description of various attack classes/types. As a result, the amount of training data that needs to be provided to the anomaly detection system is reduced.

3.1. Classification of Clustering

There are essentially two types of clustering methods: hierarchical algorithms and partitioning algorithms.

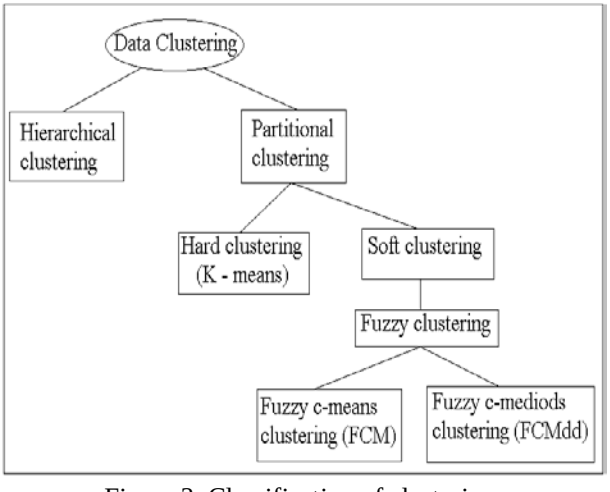

Figure.3. Classification of clustering

The hierarchical algorithms can be divided into agglomerative and splitting procedures. The first type of hierarchical clustering starts from the finest partition possible (each observation forms a cluster) and groups them. The second type starts with the coarsest partition possible: one cluster contains all of the observations. It proceeds by splitting the single cluster up into smaller sized clusters. The partitioning algorithms start from a given group definition and proceed by exchanging elements between groups until a certain score is optimized.

The main difference between the two clustering techniques is that in hierarchical clustering once groups are found and elements are assigned to the groups, this assignment cannot be changed.

In partitioning techniques, on the other hand, the assignment of objects into groups may change during the algorithm application.

- Hard Clustering – based on mathematical set theory i.e. either a data point belong to a particular Cluster or not.
- Soft Clustering – based on fuzzy set theory i.e. a data point may partially belong to a cluster.
- Fuzzy Clustering – Fuzzy Logic + Clustering.
  - Fuzzy c means (FCM) – Fuzzy Clustering + K-means.
  - Fuzzy c medoids (FCMdd) – based on distance between pairs of data points.

3.2. Hard Clustering Vs Soft Clustering

Hard partition means that the clustering algorithm divides the data set based on certain criteria that evaluate the goodness of a partition. In other words, hard partition divides the input space into the number of partitions defined by the user and as such each datum belongs to exactly one cluster of the partition.

Soft partitioning, not always fuzzy partitioning, partitions a given dataset and not an input space, since the input space can be larger than the dataset. In real time cases, most soft clustering algorithms form a data partition that also corresponds to fuzzy partition.

The biggest drawback of hard partitioning is the concept that each datum either belongs in a partition or is strictly excluded; there is no chance for the data elements to be a part of more than on partition at the same time. However, in natural clusters it is always the case that some data elements partially belong to one set and partially to one or more other sets. In order to overcome this limitation, the notion of fuzzy partitioning was introduced.

3.2. Fuzzy Clustering

Fuzzy logic is based on fuzzy set theory. Fuzzy set theory, unlike the well known mathematical set theory, allows an element to belong to more than one clustering the interval of [0, 1]. The degree of membership of each data element to the cluster is calculated which decides which cluster the data element is supposed to belong. The existence of a data element in more than one cluster depends on the value of Fuzzifier. The user defines the fuzzification value i.e. one data element can belong to how many clusters, also known as Fuzzifier.

FUZZY LOGIC + CLUSTERING =
FUZZY CLUSTERING

There are two main types of fuzzy clustering algorithms focused in this work:-

1. *Fuzzy c-means:*

FUZZY LOGIC + K-MEANS PARTITION =
FUZZY C-MEANS

Based on mean of data points

2. *Fuzzy k-medoids:*

FUZZY LOGIC + K-MEDOIDS PARTITION =
FUZZY K-MEDOIDS

Based on median of data points

The reasons for introducing fuzzy logic is two fold, the first being the involvement of many quantitative features where there is no separation between normal operations and anomalies. Thus fuzzy association rules can be mined to find the abstract correlation among different security features.

The objective of a *fuzzy clustering* algorithm is to partition the data into clusters so that the similarity of data objects within each cluster is maximized and the similarity of data objects among clusters is minimized. In the objective function based methods, the objective function is a function of data matrix, membership matrix and prototypes of clusters. It measures the overall dissimilarity of data objects within each cluster.

Hence, by minimizing the objective function, we can obtain the best partition of the data set.

4. Fuzzy C – Means Algorithm

Fuzzy c-means clustering involves two processes: the calculation of cluster centres and the assignment of points to these centres using a form of Euclidian distance. This process is repeated until the cluster centres stabilize. The algorithm is similar to k-means clustering in many ways but it assigns a membership value to the data items for the clusters within a range of 0 to 1. So it incorporates fuzzy set's concepts of partial membership and forms overlapping clusters to support it. The algorithm needs a fuzzification parameter m in the range [1, n] which determines the degree of fuzziness in the clusters. When *m* reaches the value of 1 the algorithm works like a crisp partitioning algorithm and for larger values of m the overlapping of clusters is tend to be more.

In fuzzy clustering, the data points can belong to more than one cluster, and associated with each of the points are membership grades which indicate the degree to which the data points belong to the different clusters. Thus, points on the edge of a cluster may be *in the cluster* to a lesser degree than points in the center of cluster. For each point x we have a coefficient giving the degree of being in the kth cluster $u_k(x)$. Usually, the sum of those coefficients for any given x is defined to be 1:

$$\forall x \left(\sum_{k=1}^{\text{num. clusters}} u_k(x) = 1 \right).$$

With fuzzy *c*-means, the centroid of a cluster is the mean of all points, weighted by their degree of belonging to the cluster:

$$\text{center}_k = \frac{\sum_x u_k(x)^m x}{\sum_x u_k(x)^m}.$$

The degree of belonging is related to the inverse of the distance to the cluster center:

$$u_k(x) = \frac{1}{d(\text{center}_k, x)},$$

then the coefficients are normalized and fuzzyfied with a real parameter $m > 1$ so that their sum is 1. So

$$u_k(x) = \frac{1}{\sum_j \left(\frac{d(\text{center}_k, x)}{d(\text{center}_j, x)} \right)^{2/(m-1)}}.$$

For *m* equal to 2, this is equivalent to normalizing the coefficient linearly to make their sum 1. When *m* is close to 1, then cluster center closest to the point is given much more weight than the others.

5. Fuzzy C – Medoids Algorithm

Kaufman *et al*. in 1987 developed a k-medoids-based clustering called PAM. A medoid is defined as the object of a cluster, whose average dissimilarity to all the objects in the cluster is minimal i.e. it is a most centrally located point in the given data set.

The k-medoids [3] approach also produces a data set partition with k clusters in order to minimize the total intra-cluster dissimilarity, just like k-means algorithm. But the main difference between k-means and k-medoids lies in the selection of centre of cluster that represents the cluster. In k-medoids, the centre is a real object from the dataset while in k-means the centre may be a non-real object calculated as mean of all the data elements.

K – Medoids algorithm avoids calculating means of clusters in which extremely large values may affect the membership computations substantially. K-medoids can handle outliers well by selecting the most centrally located object in a cluster as a reference point, namely, medoid. The basic idea of k-medoids is that it first arbitrarily finds k objects amongst n objects in the dataset as the initial medoids. Then the remaining objects are partitioned into k clusters by computing the minimum Euclidian distances that can be maintained for the members in each of the clusters. An iterative process then starts to consider objects P_i, $i = 1,\ldots,n$ if a medoid o_j, $j = 1,\ldots,k$, can be replaced by a candidate object o_c, $c = 1,\ldots,n$, c not equal to i.

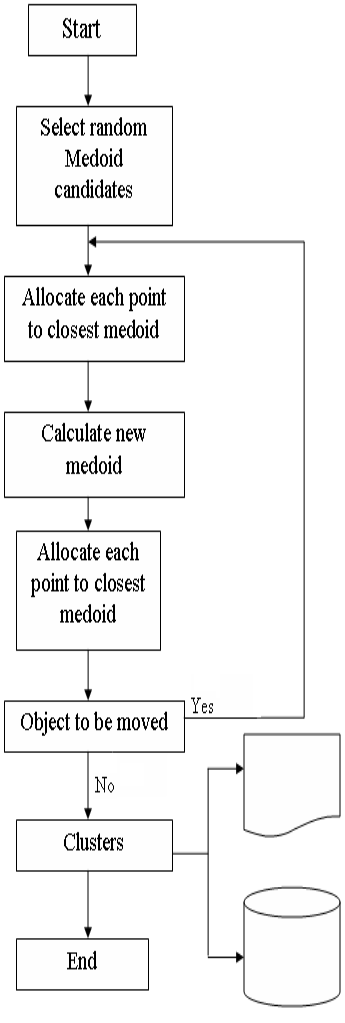

Figure.4. Flowchart of Fuzzy c-medoids algorithm

There are four situations to be considered in this process:

1. *Shift-out membership:* an object P_i may need to be shifted from currently considered cluster o_j to another cluster.

2. *Update the current medoid:* a new medoid o_c is found to replace the current medoid o_j.

3. *No change:* objects in the current cluster result have the same or even smaller SEC (square error criterion) for all possible redistributions considered.

4. *Shift in membership:* an outside object p_i is assigned to the current cluster with the new (replaced) medoid o_c.

6. Results and conclusions

The main problems in unsupervised learning are guessing the right number of output clusters and stopping criteria like number of iteration to stop specified. These two affect the accuracy and performance of the algorithm. The Robustness in handling noisy data, outliers, Order sensitivity, Dataset size and Shape that affects understanding of the clustering results.

Fuzzy k-medoids algorithm is observed to give better results than fuzzy c-means algorithm as k-medoids algorithm is based on calculating the median of the data points and gives more appropriate results. Fuzzy c-means algorithm being based on calculating the mean of data points gives centroid as the centre point which may also be close to anomaly point. In addition to this fuzzy k-medoids algorithm takes only real data points as centroid while fuzzy c-means may also take non – real data points as centroid resulting in anomaly.

6. Future Work

Although we have tried to work out the best possible and a loads of work has already been done till date but there are still many open challenges still left behind. Some of them include:

- Better methods to determine the number of clusters for evaluation of the algorithms. The optimal number of clusters may vary with every dataset, so it is very important to determine some better ways to calculate the appropriate number of clusters.
- Another issue is to classify the type of attacks. Instead of only defining the type of data as normal or anomaly it would rather be more beneficial to identify which type of attack the data element represents.

- The cost and workload involved in creating a proper environment for the simulation of IDS is high enough, which also contributes in the difficulties to do good research works in the field of intrusion detection.

10. References

[1] Markos Markou and Sameer Singh, "Novelty detection: a review – part 2", Signal processing 83, Department of Computer Science UK, pp. 2499 – 2521, 2003.

[2] A Murali M Rao, "a survey on Intrusion Detection approaches", IEEE, pp.233-240, 2005.

[3] Jian-Ping Mei, Lihui Chen, "Fuzzy clustering with weighted medoids for relational data", pp. no. 1964-1974, Pattern Recognition 43 (2010).

[4] YANG Jian, NING Yufu, "Research on initial clustering centres of Fuzzy c-means algorithm and its application to intrusion detection", pp. no. 161-163, 2nd conference on environmental science and information application technology, 2010.

[5] R.A.Maxion and R.R.Roberts, "Proper use of ROC curves in Intrusion/Anomaly detection", pp no. 01-32, Technical report series CS-TR-871, School of computing science, University of Newcastle upon Tyne, 2004.

[6] Liu Hui, CAO Yonghui, "Research intrusion detection techniques from perspective of machine learning", pp no. 166-168, 2nd international conference on multimedia and information technology, 2010.

[7] Mohd Fadzli Marhusim, David Cornforth, Henry Larkin, "An overview of recent advances in Intrusion Detection", pp no. 432-437, CIT 2008.

[8] P.Garcia-Teodoro, J.Diaz-Verdejo, G.Macia-Fernandez, E.Vazquez, "Anomaly based network intrusion detection: Techniques, systems and challenges", pp no. 18-28 computer and security 28 (2009).

[9] Kusum Kumari bharti, Sanyam shukla, Sweta Jain, "Intrusion detection using clustering", pp no.158-165 IJCCT VOl1 Issue 2,3,4, 2010.

[10] Matthias Templ , Peter Filzmoser , Clemens Reimann, "Cluster analysis applied to regional geochemical data: Problems and possibilities", pp. no. 2198-2213, Applied Geochemistry 23 (2008).

[11] T.Velmurugan, T.Santhanam, "Performance evaluation of k-means and fuzzy c-means clustering algorithms for statistical distributions of input data points", pp. no. 320-330, European journal of scientific research Vol.46 No.3, 2010.

[12] Chen zhuo, Liu qiao, Lin shan, "Research on evaluation method of Intrusion detection system", IEEE, 2010.

[13] Shu-Chuan chu, John F.Roddick, Jeng-Shyang pen, "Improved search strategies and extensions to k-medoids based algorithms –extended report", Knowledge discovery and management Laboratory, Technical Report KDM-02-005, 2002.

[14] Nelcileno Araújo, Ruy de Oliveira, Ailton Akira Shinoda, Bharat Bhargava, Ed'Wilson Ferreira, " Identifying Important Characteristics in the KDD99 Intrusion Detection Dataset by Feature Selection using a Hybrid Approach", pp. no. 552-558, 17th International Conference on Telecommunications, 2010.

[15] Animesh Patcha, Jung-Min Park, "An overview of anomaly detection techniques: existing solution and latest technological trends", Computer Networks 51(2007), pp.3448-3470, February 2007.

[16] QingQing Zhang, Hongbian Yang, Kai Li, Qian Zhang, "Research on the Intrusion Detection Technology with Hybrid Model",pp no. 646-649, 2nd conference on environment science and information application technology, 2010.

[17] Hui Zhe Zhang, Hong Chen, Li Xia Bao, "An improved Fuzzy c means clustering algorithm and its application in traffic condition recognition", pp. no. 1608-1612, 7th international conference on fuzzy systems and knowledge discovery, 2010.

[18] QingPeng Zeng, ShuiXiu Wu, "A fuzzy clustering approach for intrusion detection", 728-732, International conference on web information systems and mining, 2009.

[19] Wanli Ma, Dat Tran, Dharmendra Sharma, "A study on feature selection of network traffic for intrusion detection purpose", pp. no. 245-247, ISI, 2008.

[20] Wuling Ren, Jinzhu Cao, Xianjie Wu, "Application of Network Intrusion Detection Based on Fuzzy C-Means Clustering Algorithm", pp. no. 19-22, Third International Symposium on Intelligent Information Technology Application, 2009.

[21] Wei Jiang, Min YAO, Jun YAN, "Intrusion Detection Based on Improved Fuzzy C-means Algorithm", pp. no. 326-329, International Symposium on Information Science and Engineering, 2008.

[22] Eduardo Raul Hruschka, J. G. B. Campello, Alex A. Freitas, Andr´e C. Ponce Leon F. de Carvalho, "A Survey of Evolutionary Algorithms for Clustering", pp. no. 133-155, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS, VOL. 39, NO. 2 MARCH 2009.

[23] Devi Prasad bhukya, S Ramachandram, Reeta Sony A L, "Performance evaluation of partition based clustering algorithms in grid environment using design of experiments", pp. no. 46-53, International Journal of Reviews in Computing, 2009-10.