DOI: 10.21917/ijsc.2012.0043

PERFORMANCE EVALUATION OF C-FUZZY DECISION TREE BASED IDS WITH DIFFERENT DISTANCE MEASURES

Vinayak Mantoor[1], Krishnamoorthy Makkittaya[2] and C.B. Chandrakala[3]
[1,2]*Department of Master of Computer Applications, Manipal Institute of Technology, Karnataka, India*
E-mail: [1]vinayak.mantoor@manipal.edu and [2]k.moorthy@manipal.edu
[3]*Department of Information and Communication Technology, Manipal Institute of Technology, Karnataka, India*
E-mail: chandrakala.cb@manipal.edu

Abstract
With the ever-increasing growth of computer networks and emergence of electronic commerce in recent years, computer security has become a priority. Intrusion detection system (IDS) is often used as another wall of protection in addition to intrusion prevention techniques. This paper introduces a concept and design of decision trees based on Fuzzy clustering. Fuzzy clustering is the core functional part of the overall decision tree development and the developed tree will be referred to as C-fuzzy decision trees. Distance measure plays an important role in clustering data points. Choosing the right distance measure for a given dataset is a non-trivial problem. In this paper, we study the performance of C-fuzzy decision tree based IDS with different distance measures. We analyzed the results of our study using KDD Cup 1999 data and compared the accuracy of the classifier with different distance measures.

Keywords:
Decision Trees, Fuzzy Clustering, Experimental Study, Distance Measures, IDS

1. INTRODUCTION

The ubiquitous use of computers and computer networks in today's society has made computer network security an international priority. Since it is not technically feasible to build a system with no vulnerabilities, intrusion detection has become an important area of research. Intrusion Detection System (IDS) is a critical component of secured information systems, which detects hostile activities in a network. One key feature of intrusion detection system is its ability to provide a view of unusual activity and issue alerts notifying administrators and/or block a suspected connection. The current intrusion detection technology exists is signature based and has a lot of problems, such as failing in identifying new attacks. Therefore signature based detection has a low false positive rate and a high false negative rate. Thus a human reasoning and learning like-self-learning IDS is needed. Thus new IDS have to consider fuzzy logic concept [6]. Anomaly detection system learns the normal behavior and any communication pattern that deviates from this normal behavior is treated as intrusion. IDS is a complementary system to other security system such as Firewall, antivirus. When all of the security systems work together they can provide better security. Enormous amount of network data is to be analyzed in order to identify intrusion. Generally the intrusion detection problem is viewed as a two-class classification problem. The goal is to classify patterns of the system behavior in two categories (normal and intrusion), using patterns of known attacks, which belong to the intrusion class, and patterns of the normal behavior. The normal and the abnormal (intrusion) behaviors in networked computers are hard to predict as the boundaries cannot be well defined. This prediction process may generate false alarms in many intrusion detection systems. However, with fuzzy logic, the false alarm rate in determining intrusive activities can be reduced.

Decision trees are the commonly used architectures of machine learning and classification systems. They come with a comprehensive list of various training and pruning schemes, a diversity of discretization (quantization) algorithms, and a series of detailed learning refinements. The objective of this study is to study and modify a new class of decision tree a semi supervised technique to develop an effective IDS. The underlying conjecture is that data can be perceived as a collection of information granules [5]. Thus, the tree becomes spanned over these granules, treated now as fundamental building blocks. In turn, information granules and information granulation are almost a synonym of clusters and clustering. Fuzzy clustering is the core functional part of the overall generalized tree; they will be referred to as clustered-oriented decision trees or C-decision trees, for short. Clustering algorithms divide a data set into groups (clusters). Instances in the same cluster are similar to each other, they share certain properties. The criterion for measuring similarity is the distance between centroid of cluster and data points. Distance measure plays an important role in clustering data points. Choosing the right distance [7] measure for a given dataset is a non-trivial problem. For the datasets where the data is multidimensional, Euclidean or Minikowaski distance measures [3] are employed. In this paper, we present the experimental study of various distance (Manhattan, Euclidean, Minikowaski order 3 and 4) measures and their effects on the performance of C-fuzzy decision tree based IDS.

The rest of this paper is organized as follows: Section-2 gives an overview of C-fuzzy decision trees, and underlying processes in developing the tree. Section-3 gives analysis of results and Section-4 gives conclusion.

2. OVERALL ARCHITECTURE OF THE CLUSTER BASED DECISION TREE

The architecture of the cluster–based decision tree develops around fuzzy clusters that are treated as generic building blocks of the tree. The way in which these trees are constructed deals with successive refinements of the clusters (granules) forming the nodes of the tree. The tree grows gradually by using fuzzy C-means (FCM) clustering algorithm to split the patterns in a selected node with the maximum heterogeneity into C corresponding children nodes. The fuzzy nature of algorithm makes it as soft algorithm (unlike k-means is hard algorithm).In fuzzy clustering, the data points can belong to more than one cluster, and associated with each of the points are membership grades (membership matrix)[1] which indicate the degree to which the data points belong to the different clusters. The training data set X is clustered into ‘C’ clusters so that the data points (patterns) that are similar are put together. These clusters are completely characterized by their prototypes (centroids). Tree building start with them positioned at ‘C’ top nodes of the tree structure. The way of building the clusters implies a specific way in which elements of X allocated to each of them. In other words, each cluster comes with a subset of X namely X_1, X_2,..., X_c The process of growing the tree is guided by a certain heterogeneity criterion that quantifies a diversity of the data (with respect to the output variable y) falling under the given cluster (node). The values of the heterogeneity criterion at each node denoted by V_1, V_2, …, V_c respectively. Then choose the node with the highest value of the heterogeneity criterion and treat it as a candidate for further refinement. Let i_o be the one for which V_i assumes a maximal values. The i_o^{th} node is refined by splitting it into C clusters as visualized in Fig.1. The process is repeated by selecting the most heterogeneous node out of all final nodes. The growth of the tree is carried out by expanding the nodes and building their consecutive levels that capture more details of the structure. It is noticeable that the node expansion leads to the increase in either the depth or width (breadth) of the tree. The pattern of the growth is very much implied by the characteristics of the data as well as influenced by the number of the clusters.

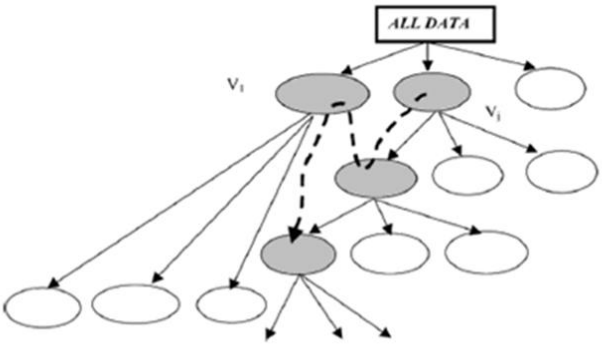

Fig.1. Growing a decision tree by expanding nodes (which are viewed as clusters located at its nodes)

Shadowed nodes are those with maximal values of the diversity criterion and thus being subject to the split operations.

Some typical patterns of growth are illustrated in Fig.1. Considering a way in which the tree expands; it is easy to notice that each node of the tree has exactly zero or children. Each node is associated the following components: the heterogeneity criterion, the number of patterns associated with it, and a list of these patterns. Moreover, each pattern on this list comes with a degree of belongingness (membership) to that node. Further splitting is stopped based on following two conditions, namely- leaf nodes may not have enough number of data points or all data points encounter full membership to the respective clusters.

2.1 DEVELOPMENT OF C-FUZZY DECISION TREE

The FCM algorithm attempts to partition a finite collection of data elements X={x_1,x_2 , , ... ,x_n } into a collection of c fuzzy clusters with respect to some given criterion. Given a finite set of data, the algorithm returns a list of C -cluster centers V such that- V =V_i where i =1, 2, , C and a partition matrix (membership matrix) U.

2.1.1 Membership Matrix:

Membership matrix describes the belongingness of each data element to different clusters centroids. The membership matrix is of following form and is calculated as below:

$$U = \begin{bmatrix} u_{11} & \cdots & u_{1n} \\ \vdots & u_{ik} & \vdots \\ u_{c1} & \cdots & u_{cn} \end{bmatrix} \tag{1}$$

Each u_{ik} is calculated as below:

$$u_{ik} = \frac{1}{\sum_{j=1}^{c}\left(\frac{d_{ik}}{d_{jk}}\right)^{2/(m-1)}} \tag{2}$$

where d_{ik} and d_{ij} are the distance between k^{th} –data point to i^{th} centroid and distance between k^{th} data point to all j=1 to c centroids.

We used four kind of distance measures namely- Manhattan, Euclidean, Minikowaski order 3 and 4.

The Manhattan distance between the point P1 with coordinates (attributes) (x_1, y_1) and the point P2 at (x_2, y_2) is:

$$D_{Manhattan} = |x_1 - x_2| + |y_1 - y_2| \tag{3}$$

It can be extended to data points with n-dimensions also.

The Euclidean distance is between two points p(x_1, y1) and q(x_2, y2) is computed as:

$$D_{Euclidean} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{4}$$

General Euclidean distance for n-space points is defined as:

$$D_{Euclidean\ (n-space)} = \sqrt{\sum_{i=0}^{n}(x_i - y_i)^2} \tag{5}$$

The Minikowaski distance of order 3 and 4 is given as below-

$$D_{\text{Minikowaski (n−space)}} = \sqrt[\lambda]{\sum_{i=0}^{n}(x_{i-}y_i)^\lambda} \tag{6}$$

where λ=3 for order 3 and λ=4 for order 4

2.1.2 Cluster Formation:

Each cluster formed is a node in FCM-tree, and each node N_i is of the following form.

$$N_i = < X_i, Y_i, U_i> \tag{7}$$

$$X_i=\{x(k) \mid u_i(x(k)) > u_j(x(k))\} \text{ for all } j \neq i$$

where - X_i denotes all the elements of data set that belongs to this node by virtue of membership grade.

$$Y_i=\{y(k) \mid x(k) \in X_i \}$$

where Y_i collects the output coordinates (class labels) of elements that have been already assigned to X_i.

2.1.3 Prototype Updating:

After very new sub clusters creation, the centroid for the new clusters is recomputed according to new data elements belonging to the sub clusters.

$$f_i = \frac{\sum_{k=1}^{N} u^{m}{}_{ik} . z_k}{\sum_{k=1}^{N} u^{m}{}_{ik}} \tag{8}$$

where Z_k– Data element of form $<x_1,x_2,\ldots\ldots x_N>$

Thus whenever, a new cluster is formed, correspondingly its centroid also gets updated.

2.1.4 Node Representative:

Node representative specifies the class label (calculated) for the node formed. This value lies between 1(normal pattern) to 2(intrusion pattern).

$$m_i = \frac{\sum_{(x_k,y_k)\in X_i \times Y_i} u_i\left(X\left(k\right)\right).y(k)}{\sum_{(x_k,y_k)\in X_i \times Y_i} u_i(X\left(k\right))} \tag{9}$$

2.1.5 Variability Index:

The variability of the data in the output space existing at this node V_i is taken as a spread around the node representative (m_i).Variability index[1] represents to what degree calculated class label (node representative) of node formed matches with the actual class labels (1-normal or 2-intrusion).

$$V_i = \sum_{(x(k),y(k))\in X_i \times Y_i} u_i\left(x(k)\right).\left(y(k)-m_i\right)^2 \tag{10}$$

In the next step, select the node of the tree (leaf) that has the highest value of V, say V_{jmax} and expand the node by forming its children by applying the clustering of the associated data set into C clusters. The process is then repeated: every time examine the leaves of the tree and expand the one with the highest value of the diversity criterion (variability index).

The growth of tree stopped based on following two conditions: The first one is - a given node can be expanded if it contains enough data points. In order to create clusters, number of records must be greater than the number of the clusters; otherwise, the clusters cannot be formed. The second stopping condition pertains to the structure of data that is considered to discover through clustering. It becomes obvious that once splitting encounters the smaller subsets of data. This becomes reflected in the entries of the partition matrix that tend to be equal to each other and equal to 1/C. If all entries of the partition matrix are equal to 1/C, then the result is equal to zero. If all the data point in a cluster encounters a full membership to a certain cluster, then the resulting value is equal to 1.

2.1.6 Decision Tree Result Validation:

The decision tree formed is validated by using following measures- confusion matrix, sensitivity, specificity and accuracy [5].

Confusion matrix tells about, out of total data points, how many data points in training set are correctly identified as intrusion (true positives) and correctly identified as normal(true negatives). It also tells how many normal connections are incorrectly identified as intrusion (false positives) and how many intrusions as normal (false negatives).

2 - Indicates intrusion (positives); 1 - Indicates normal (negatives)

(2-2)– gives true positives; (1-1) – gives true negatives

(2-1)–gives false negatives; (1-2) – gives false positives

Table.1. Confusion Matrix

Confusion Matrix		**Predicated**	
		1	2
Actual	1	(1-1)	(1-2)
	2	(2-1)	(2-2)

Sensitivity: It is the ratio of true positives to total Positives, which describes to what extent intrusions are detected as intrusions only.

$$Sentivity = \frac{True\,Positives}{Total\,Positives\,in\,sample} \tag{11}$$

Specificity: It is the ratio of true negatives to total negatives which describes to what extent normal points are detected as normal only.

$$Specificity = \frac{True\,Negatives}{Total\,Negatives\,in\,sample} \tag{12}$$

Accuracy: Accuracy is defined as –

$$\begin{aligned} Accuracy = \left(Sensitivity\right)&\frac{Positives}{Positives+Negatives} \\ +\left(Specificty\right)&\frac{Negatives}{Positives+Negatives} \end{aligned} \tag{13}$$

Positives mean total number of positives in the sample and Negatives means total negatives in the sample.

3. RESULTS

The data set used in this paper is the KDD Cup 1999[4] for intrusion detection which is generally used for benchmarking intrusion detection problems. This data was prepared by the 1998 DARPA Intrusion Detection Evaluation program by MIT Lincoln Laboratory. Lincoln labs acquired nine weeks of raw TCP dump data. The raw data was processed into connection records, which consist of about 5 million connection records. The data set contains 24 attack types. These attacks fall into four main categories: DOS, Probe, u2r, and r2l. The data set has 41 attributes for each connection record plus one class label. The data set for our experiments contained 37016 records which were randomly generated from the MIT data set. Random generation of data include the number of data from each class proportional to its size. The data is partitioned into the two classes of ‘‘Normal’’ and ‘‘Attack’’ patterns where Attack is the collection of four classes (Probe, DOS, U2R, and R2L) of attacks. The objective is to separate normal and attack pattern to make classifier into two class classifier Out of 34 numeric attributes, only 10 numeric attributes are selected [5] and the 11th attribute will be the class label. The experimental results of C-fuzzy tree based IDS with different distance measures are summarized in the Table.2.

The total variability index shows that with Euclidean distance it is less compared to other distance measures for the fixed size C-fuzzy tree. Also Sensitivity, Accuracy for C-fuzzy decision tree with Euclidean distance measure is better than with other distance measures. Hence experimentally it is concluded that the performance of C-fuzzy tree will be better with Euclidean distance measure for a fixed size. However Specificity is slightly lower than others, which indicate extent to which normal pattern are recognized as normal is slightly less than others, but overall accuracy in identifying intrusion and normal patterns is more as compared to other distance measures.

Table.2. Summarized Results with Distance Measures

Distance Measures	Total Variability index	Sensitivity	Specificity	Accuracy
Manhattan	2.1024	0.8999	0.9528	0.9119
Euclidean	**2.003**	**0.9209**	**0.9105**	**0.9201**
Minikowaski-3	2.0483	0.8820	0.9107	0.9089
Minikowaski-4	2.091	0.8908	0.9123	0.9059

4. CONCLUSION

This paper presents the performance evaluation of C-fuzzy decision tree based IDS with different distance measures. The experimental results show that the Euclidean distance measure is the best distance measure for C-fuzzy decision tree based IDS. As a future work an analytical reasoning can be developed in support of the experimental result that why Euclidean distance is better for the development of C-fuzzy decision tree based IDS.

REFERENCES

[1] Witold Pedrycz, and Zenon A. Sosnowski, “C Fuzzy Decision Tree”, *IEEE Transactions on systems, Man, and Cybernetics—part c: applications and reviews*, Vol. 35, No.4, pp. 498-511, 2005.

[2] Sapna S Kaushik, P.R.Deshmukh, “Detection of Attacks in an Intrusion Detection System”, *International Journal of Computer Science and Information Technologies (IJCSIT)*, Vol. 2, No.3, pp. 982-986, 2011.

[3] Zhan Jiuhua, “Intrusion Detection System Based on Data Mining”, *First International Workshop on Knowledge Discovery and Data Mining (WKDD), IEEE Computer Society*, pp. 402-405, 2008.

[4] KDDCup’99 Intrusion Detection datasets. Available at: http://kdd.ics.uci.edu/databases/kddcup1999/kddcup99.html, 1999. Accessed 13December 2010

[5] Krishnamoorthi Makkithaya, N.V. Subba Reddy and U. Dinesh Acharya, “Improved fuzzy decision for Intrusion detection System”, *World Academy of Science, Engineering and Technology*, Vol. 42, pp. 273-277, 2008.

[6] Lotfi A. Zadeh, “Towards a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic”, *Fuzzy Sets and Systems- ELSEVIER*, Vol. 90, pp. 111-127, 1997.

[7] Eric P. Xing, Andrew Y. Ng, Michael I. Jordan and Stuart Russell,”Distance metric learning, with application to clustering with side-information”, *Advances in Neural Information, University of California, Berkeley*, 2002.