Nuka Raju Kolli* et al.
(IJITR) INTERNATIONAL JOURNAL OF INNOVATIVE TECHNOLOGY AND RESEARCH
Volume No.2, Issue No. 4, June – July 2014, 1044 – 1048.

Research Article

A Novel Cost Metric Evaluation Method for Anomaly Detection

NUKA RAJU KOLLI
Associate Professor
Dadi Institute of Engineering & Technology
Anakapalle,

BALAJI KANDREGULA
Associate Professor
Dadi Institute of Engineering & Technology
Anakapalle

Abstract:- **In this paper we present "A Novel clustering algorithm" which is a partition based clustering algorithm that works well for data with mixed numeric and categorical features for classifying anomalous and normal activities in a computer network. The proposed method first partitions the training instances into k-clusters using dissimilarity measurement. On each cluster representing a density region of normal or anomaly instances we apply either of the two rules 1.Threshold rule 2. Bayes decision rule to obtain a final decision. We report our results of applying k-prototype clustering algorithm to the extensively gathered network audit data for the 1998 DARPA intrusion detection evaluation program.**

Index Terms: **Anomaly detection, clustering, k-prototype clustering, k-means clustering.**

I. INTRODUCTION

Intrusion detection systems aim at detecting attacks against computer systems and networks, or against information systems in general, as it is difficult to provide provably secure information systems and maintain them in such a secure state for their entire lifetime and for every utilization. Therefore, the task of intrusion detection systems is to monitor the usage of such systems and to detect the apparition of insecure states. Intrusion detection technology [1] is an important component of information security technology and an important supplement to traditional computer security mechanisms. Intrusion detection can be categorized into two types: one is anomaly detection. It firstly stores users normal behavior into feature database, then compares characters of current behavior with characters of feature database. If the deviation is large enough, We can say that the current behavior is anomaly or intrusion. Although having a low false negative rate and high false alarm rate, it can detect unknown types of attacks. The other is misuse detection. It establishes a feature library according to the known attacks, and then matches the happened behaviors to detect attacks. It can only detect known types of attacks, but is unable to detect new types of attacks. Therefore misuse detection has a low false alarm rate and a high false negative rate.

There are many methods applied into intrusion detection [7] , such as methods based on statistics, methods based on data mining, methods based on machine learning and so on. In recent years, data mining technology is developing rapidly and increasingly mature and now it is gradually applied to Intrusion Detection field. Clustering is a data mining technique where data points are clustered together based on their feature values and a similarity metric. Clustering algorithms are generally categorized under two different categories- partitional and hierarchical. Partitional clustering algorithms divide the data set into non-overlapping groups [9, 10]. Algorithms k-mean, k-modes, etc. fall under this category.

Hierarchical algorithms use the distance matrix as input and create a hierarchical set of clusters. Hierarchical clusters are may be agglomerative or divisive, each of which has different ways of determining cluster membership and representation. Bloedorn [2] use k-means approach for network intrusion detection.

II. RELATED WORK

K-means is the most important flat clustering algorithm. Its objective is to minimize the average squared Euclidean distance of documents from their cluster centers where a cluster center is defined as the mean or centroid $\vec{\mu}$ of the documents in a cluster ω :

$$\vec{\mu}(\omega) = \frac{1}{|\omega|} \sum_{\vec{x} \in \omega} \vec{x}$$

The definition assumes that documents are represented as length-normalized vectors in a real-valued space in the familiar way. They play a similar role here. The ideal cluster in K-means is a sphere with the centroid as its center of gravity. Ideally, the clusters should not overlap. Our desiderata for classes

Research Article

in Rocchio classification were the same. The difference is that we have no labeled training set in clustering for which we know which documents should be in the same cluster.

A measure of how well the centroids represent the members of their clusters is the residual sum of squares or RSS , the squared distance of each vector from its centroid summed over all vectors:

Where $$\mathrm{RSS}_k = \sum_{\vec{x} \in \omega_k} |\vec{x} - \vec{\mu}(\omega_k)|^2$$

$$\mathrm{RSS} = \sum_{k=1}^{K} \mathrm{RSS}_j$$

RSS is the objective function in K-means and our goal is to minimize it. Since N is fixed, minimizing RSS is equivalent to minimizing the average squared distance, a measure of how well centroids represent their documents.

2.1 Contribution of the Paper

The contribution of the paper is enumerated as follows:

• The paper presents a k-prototype clustering algorithm for classifying the data as normal or anomaly using Threshold rule or Bayes decision rule.

• The paper evaluates the performance of k-prototype clustering algorithm for anomaly detection and compares with the k-means clustering algorithm.

The rest of the paper is organized as follows: In section2, review of k-prototype clustering algorithm. In section3, describes related work. In section4, we discuss experimental datasets. In section5, we discuss results. In section6, we conclude our work.

III. REVIEW OF K-PROTOTYPE CLUSTERING ALGORITHM

The k-prototype algorithm [3] which work well for mixed as well as pure numeric and categorical data sets. This uses joint probability distributions based on probability of co-occurrence with other attributes. K-prototype Clustering Algorithm

```
K-MEANS({x_1, ..., x_N}, K)
 1  (s_1, s_2, ..., s_K) ← SELECTRANDOMSEEDS({x_1, ..., x_N}, K)
 2  for k ← 1 to K
 3  do μ_k ← s_k
 4  while stopping criterion has not been met
 5  do for k ← 1 to K
 6     do ω_k ← {}
 7     for n ← 1 to N
 8     do j ← arg min_j' |μ_j' − x_n|
 9        ω_j ← ω_j ∪ {x_n}  (reassignment of vectors)
10     for k ← 1 to K
11     do μ_k ← (1/|ω_k|) Σ_{x∈ω_k} x  (recomputation of centroids)
12  return {μ_1, ..., μ_K}
```

Fig1:the k-means algorithm for most IR applications, the vectors x_n belongs to R power M should be length-normalized.

Begin

Initialization – Allocate data objects to a pre-determined k number of clusters randomly.

• For every categorical attribute

• Compute distance δ(r, s) between two categorical values r and s.

• For every numeric attribute

• Compute significance of attribute

• Assign data objects to different clusters randomly.

Repeat steps 1–2

1. Compute cluster centers for C1, C2, C3, , , „Ck.

2. Each data object di (i = 1, 2, . . . , n) {n is number of data objects in data set} is assigned to its closest cluster center using (i, j) =d ,C

Until no elements change clusters or a pre-defined number of iterations are reached. End.

The cost function of k-prototype is specified in Eq.1, which is to be minimized for clustering mixed data sets.

$$\zeta = \sum_{i=1}^{n} \vartheta(d_i, C_j) \qquad (1)$$

Where

$$\vartheta(d_i, C_j) = \sum_{t=1}^{m_r} \left(w_t \left(d_{it}^{r} - C_{jt}^{r}\right)\right)^2 + \sum_{t=1}^{m_c} \Omega\left(d_{it}^{c}, C_{jt}^{c}\right)^2 \qquad (1.1)$$

$$\sum_{t=1}^{m_r} \left(w_t \left(d_{it}^{r} - C_{jt}^{r}\right)\right)^2$$

Where denotes the distance of object di from its closest cluster center Cj , for numeric attributes only, wt denotes the significance of tth numeric attribute

Research Article

which is to be computed from the data set, $\sum_{t=1}^{m} \Omega(d_{it}^{c}, C_{jt}^{c})^2$ denotes the distance between the data object di and its closest cluster center Cj in categorical attributes only. Let Ai, k denote the kth value for categorical attribute Ai. Let the total number of distinct values for Ai be pi. Then this distance is defined as

$$\Omega(X,C)=(N_{i,1c}/N_c)*\delta(X,A_{i1})+(N_{i,2c}/N_c)*\delta(X,A_{i2})+...+(N_{i,pic}/N_c)*\delta(X,A_{ipi}) \quad (1.2)$$

Algorithm ALGO_DISTANCE [3] computes the distance δ(x, y) .

The following properties hold for of δ(x, y):
(1) 0 <= δ(x, y) <= 1.
(2) δ(x, y) = δ(y, x).
(3) δ(x, x) = 0.

IV. ANOMALY DETECTION WITH K-PROTOTYPE CLUSTERING ALGORITHM

I am provided with a training data set (Xi , Yi) i=1, 2, ...N, where Xi represents an n-dimensional continuous valued vector and Yi represents the corresponding class label with "0" for normal and "1" for anomaly.

The k-prototype algorithm has the following steps:

Apply the steps listed at 2nd section to get k clusters For each test instance Z:

- Compute the distance D(Ci, Z), i=1,2,….k. Find cluster Cr that is closest to Z.

- Classify Z as an anomaly or a normal instance using either the Threshold rule or the Bayes Decision rule. The Threshold rule for classifying a test instance Z that belongs to cluster Cr is:

Assign Z->1 if [[P(ω)]↓(1γ)|Z€C)] ↓(γ))> τ Otherwise Z->0 Where "0" and "1" represent normal and anomaly classes in cluster Cr ,

[[P(ω)]↓(0γ)|Z€C)] ↓(γ)) represents the probability of anomaly instances in cluster Cr , and τ is predefined threshold. A test instance is classified as an anomaly only if it belongs to a cluster that has anomaly instances in majority.

Assign Z->1 if [[P(ω)]↓(1γ)|Z€C)] ↓(γ))> [[P(ω)]↓(oγ)|Z€C)] ↓(γ)) Otherwise Z->0,where ω0 Represents the normal class in cluster cr, [[P(ω)]↓(oγ)|Z€C)] ↓(γ)) is the probability of normal instances and in cluster cr

In our experiments I use Bayes Decision rule for classifying the given test instance as normal or anomaly activity.

V. DATA SETS

In this section, I discuss the experimental data set Network Anomaly Data (NAD), obtained by feature extracting the 1998 MIT-DARPA network traffic corpora [5,8]. The 1998 MIT-DARPA data sets [4] were collected on an evaluation test bed simulating network traffic similar to that seen between an Air Force base (INSIDE network) and the internet (OUTSIDE network). Approximately seven weeks of training data and two weeks of test data were collected by a sniffer deployed between the INSIDE and OUTSIDE network. Thirty eight different attacks were launched from the outside network. The training and testing data subsets were randomly drawn from the original NAD, the no of dimensions are 11, no of instances in training data is restricted to utmost 5,000 instances, with 70 percent of them being normal and the rest being anomaly instances. The testing data set contain utmost 2,500 unseen instances, with 80 percent of them being normal and the remaining 20 percent being anomaly instances.

VI. RESULTS AND DISCUSSION

The first step of K-means is to select as initial cluster centers K randomly selected documents, the seeds . The algorithm then moves the cluster centers around in space in order to minimize RSS. This is done iteratively by repeating two steps until a stopping criterion is met: reassigning documents to the cluster with the closest centroid; and recomputing each centroid based on the current members of its cluster. From nine iterations of the K-means algorithm for a set of points.

I can apply one of the following termination conditions. A fixed number of iterations I has been completed. This condition limits the runtime of the clustering algorithm, but in some cases the quality of the clustering will be poor because of an insufficient number of iterations.

Assignment of documents to clusters (the partitioning function γ) does not change between iterations. Except for cases with a bad local minimum, this produces a good clustering, but runtimes may be unacceptably long.

Centroids $\vec{\mu}_k$ do not change between iterations. This is equivalent to γ not changing

Research Article

Terminate when RSS falls below a threshold. This criterion ensures that the clustering is of a desired quality after termination. In practice, I need to combine it with a bound on the number of iterations to guarantee termination.

Terminate when the decrease in RSS falls below a threshold θ. For small θ, this indicates that I are close to convergence. Again, I need to combine it with a bound on the number of iterations to prevent very long runtimes.

I now show that K-means converges by proving that RSS monotonically decreases in each iteration. I will use decrease in the meaning decrease or does not change in this section. First, RSS decreases in the reassignment

step since each vector is assigned to the closest centroid, so the distance it contributes to RSS decreases. Second, it decreases in the recomputation step because the new centroid is the vector $\vec{v}$ for which RSS_k reaches its minimum.

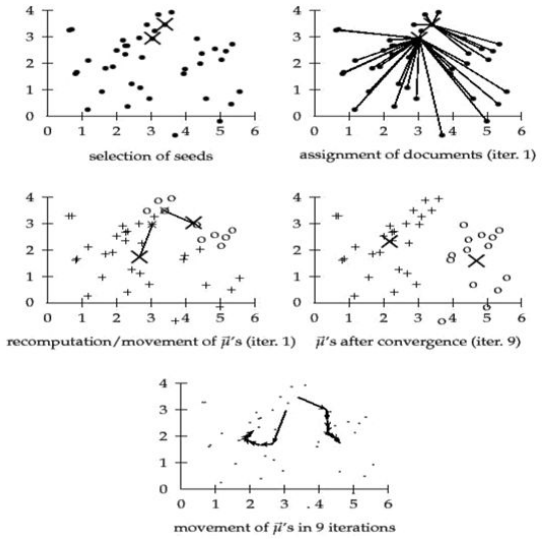

Fig1: a k-means example for k=2 in R2. The position of the two centroids converges after nine iterations.

In this section, I present and compare our results of the k-prototype clustering algorithm with Bayes Decision rule for classifying anomaly detection with k-means method [6] over NAD-98 data set. I use four measures for comparing performance as shown in Fig 1:

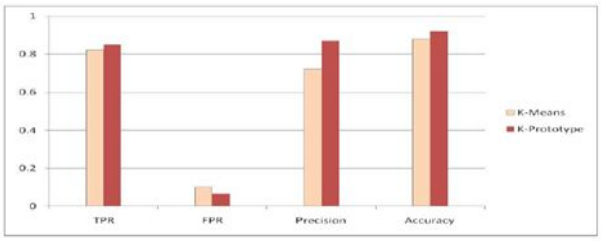

Fig.2 Performance of the k-means, k-prototype Clustering methods for Anomaly Detection over the NAD-1998 data set.

The same efficiency problem is addressed by K-medoids , a variant of K-means that computes medoids instead of centroids as cluster centers. I define the medoid of a cluster as the document vector that is closest to the centroid. Since medoids are sparse document vectors, distance computations are fast.

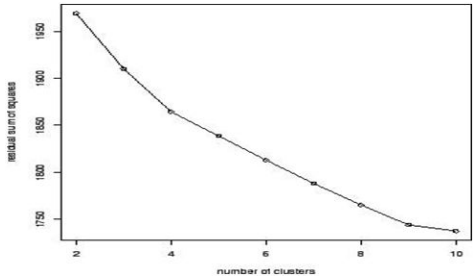

Fig3:Estimated minimal residual sum of squares as a function of the number of clusters in K-means.

In this clustering of 1203 Reuters-RCV1 documents, there are two points where the RSS_{min} curve flattens: at 4 clusters and at 9 clusters.

The performance measures precision, TPR determine how the k-means and k-prototype clustering methods perform in identifying anomaly instances. The performance measures accuracy determines the number of normal and anomaly instances correctly classified. The measures FPR determine the number of normal instances that incorrectly classified as anomaly.

VII. CONCLUSION

In this paper, I developed the k-prototype clustering method for anomaly detection. In this k-prototype method, cost function and distance measure is based on co-occurrence of values. The k-prototype clustering method is first applied to partition the training instances into k disjoint clusters, and then I apply the Bayes Decision rule for classification of

given instance as normal or anomaly, thereby improving the overall classification performance. I compare our results with k-means for classification over the NAD-98 data set.

REFERENCES

[1] Qun Yang, "A survey of Intrusion Detection Technology [J]," Network Security Technology & Application, 2008.

[2] Bloedorn, E., A. D. Christiansen, W.Hill, C. Skorupka, L. M. Talbot, and J.Tivel (2001, August). Data mining for network intrusion detection: How to get started . http://citeseer.nj.nec.com/523955.html.

[3] Amir Ahmad and Lipika Dey "A k-mean clustering algorithm for mixed numeric and categorical data." Data & Knowledge Engineering 63 (2007) 503–527].

[4] R.P. Lippman, D.J. Fried, I. Graf, J. Haines, K. Kendall, D.McClung, D. Weber, S. Webster, D. Wyschogrod, R.K. Cunningham, and M.A. Zissman, "Evaluating Intrusion Detection Systems: The 1998 DARPA Off-Line Intrusion Detection Evaluation," Proc.DARPA Information Survivability Conf. and Exposition (DISCEX '00),pp. 12-26, Jan. 2000.

[5] J. Haines, L. Rossey, R.P. Lippman, and R.K. Cunningham,"Extending the DARPA Offline Intrusion Detection Evaluation," Proc. DARPA Information Survivability Conf. and Exposition (DISCEX'01),June 2001.

