JCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.-6, Issue-6, Jun 2018 E-ISSN: 2347-2693

Outlier Detection Using Association Rule Mining and Cluster Analysis

C. Leela Krishna[1*], C. Kala Krishna[2]

[1*] Dept. of CSE, Sri Venkateswara University College of Engg., Tirupati, India
[2]Dept. of EIE, Sree Vidyanikethan Engg. College, Tirupati, India

[*]*Corresponding Author: chittem.leelakrishna03@gmail.com, Mobile: +91-86863-91266*

Available online at: www.ijcseonline.org

Accepted: 03/Jun/2018, Published: 30/Jun/2018

Abstract— An object whose behaviour is found to be different from others in a dataset is said to be an outlier. The existing outlier detection algorithms are able to detect outliers only in static datasets, but are found to be inappropriate, when it comes to dynamic datasets where data arrive continuously in a stream-lined fashion viz., sensor data. To deal with such steam data, Association rule mining serves as a best technique, where frequent item sets are internally evaluated from the data, in an iterative fashion. Outlier detection techniques for static datasets include cluster analysis, where clusters are being generated from the data using k-Means clustering to discover outliers. In this paper, we propose two different approaches for outlier detection. One uses association rule based technique on dynamic datasets and the other uses K-means clustering and distance based approach on static datasets to prune local outliers. Experiments are conducted on different variants of static and dynamic datasets to detect the deviant objects (outliers) effectively in fewer computations.

Keywords— outlier, static data, dynamic data, association rule mining, cluster analysis

I. Introduction

An outlier is referred to be a data point or an object which differs from other data points or objects within a data set. One of the important data mining techniques commonly used in detecting deviant data present in a data set is Outlier detection. Examples of outlier detection includes fraud detection in credit cards, detection of network intrusions, analysis of stock markets, etc., where efficient algorithms are used in resolving those problems.

The characteristic feature of streamed data is that data arrives continuously and in a time based order. Hence they are used in various applications like feeding of sensor data, measurement of network traffic. Generally, various outlier detection algorithms are applied on static databases. Because their work is to check whole data sets in an iterative manner to find top outliers or global outliers. But detecting outliers in streamed data is difficult for two reasons:

1. The data sets in streamed data are unbounded, which prevents scanning the entire data stream.
2. Streamed data require quick responses for their request and the process has to be completed in a limited time. Multiple scans for the data stream are not possible.

In static databases, distance-based techniques are used to model the data points and outliers are determined. Various distance functions used are Euclidean distance, Manhattan distance and Minkowski distance. In clustering based outlier detection method, K-means algorithm is the most popular one, where the data set is to be divided into clusters. The objects or data points which lie near to the centroid of the cluster are assigned to the cluster and those objects or data points which lie away from the cluster centroid are treated as outliers. While evaluating the deviation (degree of outlierness) of an object, first the objects within a cluster are obtained, and they are pruned from each cluster during outlier detection. This results in the reduction of computational time.

In this paper, we have proposed two approaches for outlier detection, one involving streaming data and the other involving static databases. In the first approach, an algorithm employing association rules to test the streamed data for finding the outliers is proposed. This algorithm defines a bounded set of transactions for a data stream, which in turn derives the hidden association rules among all the transactions. Those transactions containing outliers will be evaluated based on those rules. For instance, if {P,Q} → {R} is an association rule with a high confidence value of 85%, then a transaction <P,Q,S,T> is an outlier because the item R is not appearing in the transaction. A transaction is said to contain an outlier, if it does not contain an item which it is supposed to contain based on the high confidence association rule.

In the second approach, a static database is taken and among various data points, a distance function is applied on them and various clusters are formed. The objects which lie near to the cluster centroid are pruned to evaluate outliers. Then a distance-based measure i.e., a Local Distance-based Outlier Factor (LDOF) [1] is applied to the remaining objects, whose task is to find whether an object is an outlier or not. Once *n* outliers in a data set are found, then the top *n* objects among *n* are reported as outliers.

The remaining sections of this paper are organized as follows: Section 2 provides related works on outlier detection. Section 3 describes an association rule based outlier detection method. Section 4 describes a pruning based local distance-based outlier factor outlier detection method. Section 5 provides experimental results. Section 6 concludes the paper.

II. Related Work

A. Data Models in Streamed Data

Streamed data when compared with the traditional static databases contain unbounded data sets and it is difficult to choose finite number of transactions for mining. In order to resolve this problem, the below two models were proposed:

1. The accumulative data model which takes the transactions from the starting to the current time into consideration. If any new transaction arrives, then it is accumulated into the database.
2. The sliding window model which is used to form a bounded set of transactions. When mining is to be performed, then only the transactions that are in the sliding window are taken into consideration. The results of mining are updated as soon as the window passes by.

B. Association Rules and Outliers in Streamed Data

To obtain association rules, the prefix-tree is traversed for obtaining subsets of frequent item sets. A traversal stack is maintained to derive association rules, which stores a frequent itemset in the prefix-tree [1] and derives all possible association rules from the itemset.

A transaction is said to contain an outlier if some items were supposed to be present in a transaction but they are actually not present there. To evaluate whether a transaction has an outlier, an outlier degree is defined.

The estDec method [2] is considered which internally uses a sliding window model, and it finds frequent item sets within the current window. Initially, the estDec method performs a check on the transactions currently in the sliding window, so that it builds a prefix-tree *P* by considering the frequent item sets. Once the window slides to the next bucket, the estDec method finds new frequent item sets and it performs either insertion or deletion of item sets into *P* and the process continues iteratively.

C. Distance-Based Outlier Detection Techniques

Knorr and Ng [3] introduced a definition to distance-based outlier detection techniques [4]. *An object x in a data set D is a DB(y, dist) - outlier if at least fraction y of the objects in D lie at a greater distance than dist from x [4].*

Angiulli and Pizzuti [5]-[7] proposed an outlier detection method based on the concept of objects' neighbourhood, where outliers are determined after ranking each object based on calculating sum of the distances from its *k*-nearest neighbours.

Breunig, Kriegel and Ng [8] proposed a new definition named as Local Outlier Factor (LOF) to each object, which gives its degree of outlierness. To evaluate LOF for an object, a restricted neighbourhood of it is considered by comparing the density of the object with its neighbours.

Zhang, Hutter and Jin [9] proposed a local distance-based outlier detection method for finding outlier objects. For every object, the local distance-based outlier factor [9] (LDOF) evaluates the degree to which it deviates from its nearest objects. Calculation of LDOF for all the objects is complex i.e., $O(N^2)$, where *N* specifies number of objects in a data set.

Narita and Katigawa [10] proposed an approach for detecting outliers using association rules through a sliding window method. Based on [11]-[12] work, clustering techniques like K-means and Distance based methods were considered in detecting outliers in static data.

III. Association Rule Based Outlier Detection Method

A. Overview

Based on [10] work, a sliding window is taken for making a bounded data stream. After considering transactions within the window, a prefix-tree is built and then association rules are derived from the tree by traversing it iteratively in multiple scans. Once a transaction's items are not found in the association rules list, then such transaction may be an outlier.

The following figure shows a sample traversal stack containing items, their counts and also an arrow indicating its top position:

item	Count
i_n	$C(i_1,i_2,....i_n)$
i_{n-1}	$C(i_1,i_2,....i_{n-1})$
i_{n-2}	$C(i_1,i_2,....i_{n-2})$
...	...
i_2	$C(i_1,i_2)$
i_1	$C(i_1)$

Figure 1. A sample traversal stack

The following definitions are regarding the outlier degree given by Narita and Katigawa in [10]:

[1] Definition 1. Let *t* be a transaction and *R* be a set of association rules, and then *t's* associative closure t^+ is given as below:

$$t^0 = t$$
$$t^{i+1} = t^i \cup \{e | e \subseteq t^i \wedge X \rightarrow Y \in R\}$$

$$t^{+} = t^{\infty} \qquad (1)$$

[1] **Definition 2**. Let *t* be a transaction and *R* be a set of association rules, and $t^+$is the $t's$ associative closure. Then the outlier degree of *t* is given as below:

OutlierDegree(t)=$|t^{+} - t|/|t^{+}|$ (2)

The range of outlier degree lies between 0 and 1.

Definition 3. A transaction is said to be an outlier transaction if its outlier degree is greater than or equal to minimum outlier degree, which is considered to be a threshold.

The following steps provide a way to detect outliers in transaction data sets:

1. Building a prefix tree that keeps track of all the frequent items within a sliding window.
2. Deriving a set of association rules with higher confidence value from the prefix tree.
3. Evaluating outlier transactions from the transaction set within the sliding window.
4. Dividing outlier transactions into two subsets, one containing unobserved frequent item sets and the other containing infrequent items.

B. Example

Let us consider a transaction data set [1] consisting of 10 transactions in a sliding window, each consisting of the items [1] Book, Joke, milk, bacon, Chocos, egg. The following table gives the 10 transactions and the items in each transaction:

[1] Table 1. A Sample Transaction Data set

TID	Items
1	Book, Joke, Milk
2	Book, Chocos, Joke, Milk
3	Book, Joke, Milk
4	Bacon, Book, Chocos, Egg, Milk
5	Bacon, Book, Chocos, Egg, Joke, Milk
6	Book, Chocos, Joke, Milk
7	Bacon, Book, Egg, Milk
8	Bacon, Book, Egg, Joke, Milk
9	Book, Joke, Milk
10	Bacon, Egg, Milk

For deriving association rules from table 1, the values of minimum confidence and minimum support are set to 80% and 50% respectively. The association rules derived after setting the above values are given in table 2, where each association rule has a high confidence value.

[1] Table 2. Association Rules Derived from Table 1

RID	Rule
1	{Joke} → {Book}
2	{Joke, Milk} → {Book}
3	{Joke} → {Book, Milk}
4	{Bacon} → {Egg}
5	{Bacon, Milk} → {Egg}
6	{Bacon} → {Egg, Milk}
7	{Milk} → {Book}

From (1), the association closure t^+ is for transaction 2 will be <Bacon, Chocos, Book, Egg, Joke, Milk>. From (2), the outlier degree for transaction 2 will be 0.33.

According to definition 3, [1] if the minimum outlier degree is set to 0.3, then transaction 2 will be an outlier. Similarly, transaction 10 will also be an outlier since it does not follow any of the high confidence association rule given in table 2.

IV. Pruning Based L d of Outlier Detection Method

A. Overview

Based on the [9] work, we are using Local Distance-based Outlier Factor (LDOF) measure which specifies by how much an object deviates from its neighbors. If the value of LDOF is high, then such an object is said to deviate more from its neighbors and it can be considered to be an outlier. For an object *p*, the LDOF value can be given as:

LDOF (p) = $\overline{d_p}/\overline{D_p}$ (3)

where $\overline{d_p}$ specifies the *k*-nearest neighbor distance of *p* and $\overline{D_p}$ specifies the *k*-nearest neighbor inner distance of *p*.

For an object *p*, the $\overline{d_p}$ distance is equal to the average distance from *p* to all objects in its nearest neighbors' set.

$$\overline{d_p} = \frac{1}{k}\sum_{q\,\in\,N_p} distance\ (p, q) \qquad (4)$$

For an object *p*, the $\overline{D_p}$ distance is equal to the average distance among all the objects in its nearest neighbours' set.

$$\overline{D_p} = \frac{1}{k(k-1)}\sum_{q,q'\,\in\,N_{p,q\neq q'}} distance\ (q, q') \qquad (5)$$

B. Procedure

The major drawback of LDOF algorithm is that it is more expensive with respect to the computations, because for each object in the data set we have to calculate LDOF. To resolve this drawback, we have proposed a K-means algorithm to form clusters. After clusters are formed, radius for each cluster is calculated and the objects whose distance from the centroid is less than the radius of the cluster are pruned. Then LDOF for the remaining (unpruned) objects is calculated, which reduces the total computations. Among the objects, the top-*n* objects with higher LDOF values are considered as outliers.

The following steps are involved in performing the pruning based LDOF outlier detection algorithm:

1. **Formation of clusters:** Here we consider the entire data set and apply K-means algorithm to generate clusters and the radius for each cluster is calculated.
2. **Clusters with minimal objects:** If a cluster is found to contain less number of objects than the number of outliers required, then pruning is avoided in the cluster.

3. **Pruning objects within a cluster:** Distance for each object in a cluster from its centroid is calculated. If the distance is less than the cluster radius, then the object is pruned.
4. **Evaluation of outliers:** For the unpruned objects in each cluster, LDOF is calculated and the ton-*n* objects with high LDOF value are declared as outliers.

C. Algorithm

The following algorithm gives the steps involved in pruning based LDOF outlier detection algorithm:

```
begin OutlierDetection($DS,c,it,n$)
set $X \leftarrow Kmeans(c,it,DS)$
for each cluster $C_j$ in X do
$Radius_j \leftarrow radius(C_j)$
endfor
if$|C_j| > n$then
for each object $p_i$ in $C_j$do
        if$distance(p_i,o_j) < Radius_j$then
        $prune(p_i)$
        else
          Add $p_i$ to $U$
        endif
endfor
else
for each object $p_i$ in $C_j$do
        Add $p_i$ to $U$
endfor
endif
for each object $p_i$ in $C_j$do
  calculate LDOF ($p_i$)
endfor
sort the objects based on their LDOF ($p_i$) values
choose $n$ objects among the highest LDOF ($p_i$) values and display them as outliers.
end OutlierDetection()
```

The computational complexity of our algorithm is $c*it*N + c*n_p + (w*N)^2$, where *c* is the number of clusters that are to be generated, *it* is the number of iterations needed and *N* is the number of objects in the DS data set, n_p is the average number of objects in each cluster, *w* represents the fraction of objects that are unpruned.

V. Experimental Results

In this section, we perform two experiments one regarding the outlier detection by deriving association rules from a streamed data and the other regarding the outlier detection by calculating LDOF through pruning. While experimenting, we considered a supermarket data set consisting of 4627 transactions and 217 attributes representing items in the supermarket. For deriving [1] association rules, both the minimum support and minimum confidence are set to 30% and 80% respectively. Among 217 attributes, we have considered only 18 attributes. The number of association rules is set to 8. The association rules derived from the data set are given in table 3, which consists of rule id and the rules which have high confidence value.

Table 3. Association Rules Derived From Supermarket Data Set

Rule ID	Rule
1	{biscuits, vegetables} → {Book, cake}
2	{total} → {Book, cake}
3	{biscuits, milk-cream} → {Book, cake}
4	{biscuits, fruit} → {Book, cake}
5	{biscuits, frozen-foods} → {Book, cake}
6	{frozen-foods, fruit} → {Book, cake}
7	{frozen-foods, milk-cream} → {Book, cake}
8	{baking-needs, milk-cream} → {Book, cake}

The confidence value for the rules 1-4 is 0.84 and the confidence value for the rules 5-8 is 0.83. As per the association rules, if a transaction contains biscuits and vegetables, then it should contain both Book and cake. If any one of Book and cake is found to be missing, then such a transaction is said to be an outlier. Similarly, transactions which do not follow above high confidence association rules are said to be outliers.

VI. Conclusions and Future Work

In this paper, we have considered outlier detection problem in two varieties of databases, one involving data streams, where data arrives continuously and also in a time based order, and the other involving static databases. We have provided two algorithms for the problem.

For streamed data, a sliding window is considered to make the data items in a database bounded. Then for all the transactions in the bounded data set, a prefix-tree is taken and items are added to it, and it serves as a stack. Association rules are derived from the transactions data set and the items which do not obey the association rules are declared as outliers.

For static databases, pruning based outlier detection is performed, which internally uses K-means clustering algorithm followed by local distance-based outlier factor for each object within a cluster. The objects with high LDOF values are declared as outliers.

Finally, we hope this work will attain further interest in various problem areas of data mining such as text mining, multimedia data mining etc.

Acknowledgments

The authors would like to acknowledge Li-Jen Kao, Yo-Ping Huang, P. Rajendra, D. Jatindra Kumar and N. Sukumar for their valuable discussions.

References

[1] Li-Jen Kao, Yo-Ping Huang, “Association rules based algorithm for identifying outlier transactions in data stream,” *IEEE International Conference on Systems, Man, and Cybernetics*, Oct. 14-17, 2012.

[2] J.H. Chang and W.S. Lee, “Finding recent frequent item sets adaptively over online data streams,” *in Proceedings of the 9th ACM SIGKDD, Washington, DC, USA*, pp.487-492, August 2003.

[3] E.M. Knorr and R.T. Ng, “Algorithms for mining distance-based outliers in large databases,” *In Proceedings 24th International Conference on Very Large Data Bases, VLDB*, pp. 392-403, 1998.

[4] P. Rajendra, D. Jatindra Kumar, N. Sukumar, “An outlier detection method based on clustering*,” International Conference on Emerging Applications of Information Technology*, 2011.

[5] F. Angiulli, S. Basta, and C. Pizzuti, “Distance-based detection and prediction of outliers,” *IEEE Transactions on Knowledge and Data Engineering*, 18:145-160, 2006.

[6] F. Angiulli and C. Pizzuti, “Fast outlier detection in high dimensional spaces,” *In PKDD ’02: Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery*, pp. 15-26, 2002.

[7] F. Angiulli and C. Pizzuti, “Outlier mining in large high-dimensional data sets,” *IEEE Transactions on Knowledge and Data Engineering*, 17:203-215, 2005.

[8] M.M. Breunig, H.-P. Kriegel, R.T. Ng, and J. Sander, “LOF: identifying density-based local outliers,” *SIGMOD Rec*., 29(2):93-104, 2000.

[9] K. Zhang, M. Hutter, and H. Jin, “A new local distance-based outlier detection approach for scattered real-world data,” *In PAKDD ’09: Proceedings of the 13th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, pp.813-822, 2009.

[10] K. Narita and H. Katigawa, “Outlier detection for transactional Databases using association rules,” in *Proceedings of the 9th International Conference on Web-Age Information Management,* Zhangjiajie, Hunan, pp. 373-380, July 2008.

[11] R.S. Walse, G.D. Kurundkar, P.U. Bhalchandra, “A Review: Design and Development of Novel Techniques for Clustering and Classification of Data”, *International Journal of Scientific Research in Computer Sciences and Engineering*, Vol. 06, pp. 19-22, Jan-2018.

[12] Namrata Ghuse, Pranali Pawar, Amol Potgantwar, “An Improved Approach For Fraud Detection in Health Insurance Using Data Mining Techniques”, International Journal of Scientific Research in Network Security and Communication, vol. 5, issue 5, June-2017.

Authors Profile

Mr. C. Leela Krishna received his B.Tech degree from SVU Tirupati in the department of Computer Science and Engineering in 2012 and M.Tech degree from JNTU Anantapur in the department of Computer Science and Engineering in 2014. He is currently pursuing his Ph.D as a Full-time Research Scholar in the department of Computer Science and Engineering at S.V. University, Tirupati. His areas of Research include Data Mining, Web Intelligence and Machine Learning.

Mr. C. Kala Krishna received his B.Tech degree from JNTU Anantapur in the department of Electronics and Instrumentation Engineering in 2014 and M.Tech degree from SVU Tirupati in the department of Electrical and Electronics Engineering in 2017. He is currently working as Assistant Professor in the department of Electronics and Instrumentation Engineering at Sree Vidyanikethan Engineering College, Tirupati. His areas of interest include Cluster Analysis, Network Theory and Control Systems.