An Efficient Strategy to Detect Outlier Transactions

Anjali Barmade, Madhu M.Nashipudinath

Abstract—Instant identification of outlier patterns is very important in modern-day engineering problems such as credit card fraud detection and network intrusion detection.Most previous studies focused on finding outliers that are hidden in numerical datasets. Unfortunately, those outlier detection methods were not directly applicable to real life transaction databases. .Outlier detection methods are divided into transaction specific and non transaction specific outlier detection methods,In these paper we are going to focus mainly on transaction specific methods and detect outlier transactions from transactional databases e.g. purchase of the data at the store, customer dataset at a company.Here we are going to compare two transaction specific methods and find efficient method from them.

Keywords—outlierdetection, transactional databases, association rule, frequent pattern

I. INTRODUCTION

Outlier detection has been a very important concept in the realm of data analysis. Recently, several application domains have realized the direct mapping between outliers in data and real world anomalies, that are of great interest to an analyst.Outlier detection refers to the problem of finding patterns in data that do not conform to expected normal behavior. These anomalous patterns are often referred to as outliers, anomalies, discordant observations, exceptions, faults, defects, aberrations, noise, errors, damage, surprise, novelty, peculiarities or contaminants in different application domains.

Outlier detection has been a widely researched problem and immense use in a wide variety of application domains such as credit card, insurance, tax fraud detection, intrusion detection for cyber security, fault detection in safety critical systems, military surveillance for enemy activities and many other areas.Outliers are non-conforming patterns in data; that is, they are patterns that do not exhibit normal behaviour.

Data mining is the principle of sorting through large amounts of data and picking out relevant information. Outlier detection is the one of data mining techniques that detects rare events, deviant objects,and exceptions[7][8] .Most previous studies focused on finding outliers that are hidden in numerical datasets. Unfortunately, those outlier detection methods were not directly applicable to real life transaction databases. Although a limited literature presented methods to find outliers in the transaction datasets, they did not address what really caused the transactions to become abnormal[2][5]. Since they need efforts to transform categorical to numerical data, most existing methods are not directly applicable to categorical datasets such as transaction records in databases.

Manuscript received January 15, 2014.

Anjali Barmade, Department of Computer Department, PIIT, Mumbai, India

Madhu M. Nashipudinath, Department of Computer Department, PIIT, Mumbai, India.

Since there are many transaction databases in the real world, detecting outliers from them is also important and desirable. Now, we target transaction data and present a framework for detecting outlier transactions that significantly deviate from regularities or features in the input data.

There are two outlier detection methods transaction specific and non transaction specific outlier detection methods.We are going to focus on transaction specific outlier detection methods.In these paper we are going to compare two methods association rule based outlier detection method and frequent pattern based outlier detection method.

The remaining part of the paper is organized as follows.Section II introduces to related work on outlier detection methods.Section III describes the two outlier detection methods and compare them.Section IV provides experimental evidences to find efficient method and Section V concludes the paper.

II. RELATED WORK

This section describes the work related to outlier detection methods.

A. Outliers

Outliers are non-conforming patterns in data; that is, they are patterns that do not exhibit normal behaviour.Points that are sufficiently far away from the normal region (e.g., points O1,O2, O3 and points in O4 regions) are outliers.[7]

Outliers are often considered as an error or noise, they may carry important information. Detected outliers are candidates for aberrant data that may otherwise adversely lead to model misspecification, biased parameter estimation and incorrect results. It is therefore important to identify them prior to modeling and analysis.

Outlier, is one that appears to deviate markedly from other members of the sample in which it occurs. Anomaly objects are often known as outliers. Anomaly Detection find objects that are different from most other objects.

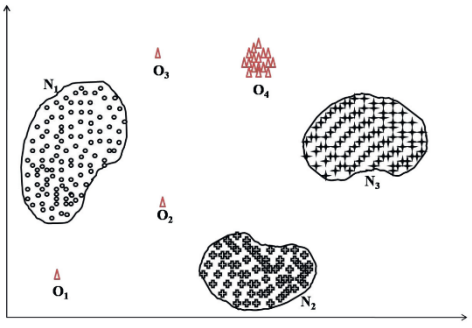

Figure 1. Outliers

B. Difference between noise and Outlier

Noise and outlier are different.Any data that has been received, stored, or changed in such a manner that it cannot be read or used by the program that originally created it can be described as noisy.It is meaningless data or corrupted data.It unnecessarily increases the amount of storage space. It is caused by hardware failure, programming error etc.It is the

error scattered in data.It may have values close to your true signal.Outliers are extreme from normal . Outlier can be analyzed and used ,but noise should be removed and has no applications or use to analyst. Outlier violates the mechanism that generates the normal data.

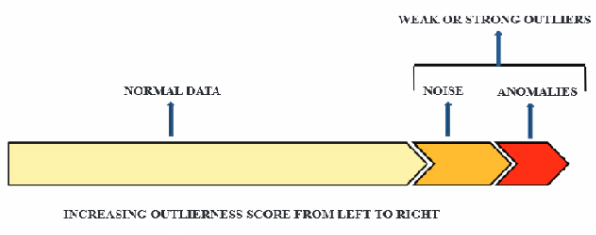

Figure 2. Difference between Outliers and Noise

C. What to do with Outliers

Try running your analysis with and without outlier and note the difference. Drop it ,if it is a mistake without loss of data. Usually it is of interest to analyst and can be used in applications**.**

D. Causes for Outliers

Malicious activity such as insurance or credit card or telecom fraud, Instrumentation error, changes in environment,human error ,poor data quality etc.

E. Types of Outliers

- Univariate outliers: These are the cases that have an unusual value for a single variable.
- Multivariate outliers: These are the cases that have an unusual combination of values for a number of variables. The value for any of the indvidual variables may not be a univariate outlier, but, in combination with other variables.
- Local outliers : These are observations inconsistent with their neighborhoods
- Global outlier (or point anomaly): Global outlier are Observations inconsistent with rest of the dataset
- Contextual outlier (or conditional outlier): It deviates significantly based on a selected context.
- Collective Outliers: A subset of data objects *collectively* deviate significantly from the whole data set, even if the individual data objects may not be outlier.

F. Outlier detection methods

Outlier detection refers to the problem of finding patterns in data that do not conform to expected normal behavior.There are two outlier detection methods transaction specific outlier detection method and non methods transaction specific outlier detection method.

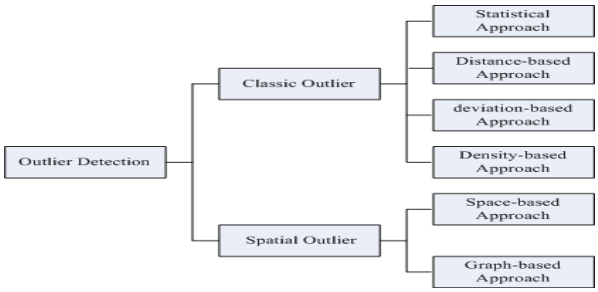

Figure 3. Non Transaction specific outlier detection method

In statistical approach the parameters are computed assuming all data points have been generated by a statistical distribution like Gaussian method.Outliers are points that have a low probability to be generated by the overall distribution.In depth based approach outliers are located at the border of the data space.Normal objects are in the center of the data space.In distance based approach[17] normal data objects have a dense neighborhood.Outliers are far apart from their neighbors, i.e., have a less dense neighbourhood.In density based approach an outlier is considerably different to the density around its neighbors i.e. outlier score.Classification[16] and clustering[19] are also non transaction specific methods.

Transaction specific outlier detection methods are association rule based outlier detection method and frequent pattern outlier detection method.

III. TRANSACTION SPECIFIC OUTLIER DETECTION METHOD

A. Association rule based outlier detection method

Association rule is an implication expression of the form X → Y, where X and Y are itemsets.Example: {Milk, Diaper} → {Beer}. Association rule mining: Given a set of transactions, find rules that will predict the occurrence of an item based on the occurrences of other items in the transaction.

According to Association rule based outlier detection method by Narita and Kitagawa[2][5],an outlier transaction is defined as a transaction that is expected to contain some items that actually did not appear there.Let T be a transaction dataset and a transaction in T is denoted as t. Let I be the set of all items. A set X ⊆ I is called an itemset. X's support is denoted by.

$$\text{support}(X) = \frac{|X \subseteq t|}{|T|}.$$

An itemset is frequent if its support is larger than or equal to a pre-defined support threshold min_sup.For two sets X ,Y ⊆ I and X ∩Y =φ , the rule X →Y means if X occurs then Y also occurs. The confidence of X →Y is defined as confidence (X →Y) :

$$\text{confidence}(X \to Y) = \frac{\text{support}(X \cup Y)}{\text{support}(X)}.$$

Narita and Kitagawa introduced the outlier degree toevaluate whether a transaction is an outlier or not.Let t be a transaction, R be the set of high confidence association rules, and t+ be the associative closure of t. The outlier degree of t is defined as od(t) :

$$od(t) = \frac{|t^+ - t|}{|t^+|}.$$

od derived using association rule to detect transactions that are outlier from a transaction database.The value of outlier degree should be between 0 and 1.An outlier transaction is a transaction that its outlier degree od(t) is greater than or equal to min_od, a predefined outlier degree threshold.min_od is minimal outlier degree.

Association rule based outlier detection method algorithm by Narita and Kitagawa

Steps:

1. Get association rules set R

from a transaction dataset T

2. Get outlier transactions set OT for each transaction t in Tget t's associative closure by checking Rcalculate outlier degree od(t)if od(t) >= min_od thenOT = OT ∪ {t}

Example : Consider purchase data at a store. Each row lists the commercial products a consumer bought on one shopping event. The first column corresponds to transaction IDs. The second column lists transactions that are represented as sets of commercial products (*items*). Also, Table 2 shows all possible association rules generated with the support threshold at 50% and the confidence threshold at 80%.

TABLE 1. A SAMPLE TRANSACTION DATASET

TID	Items
1	Bread, Jam, Milk
2	Bacon, Corn, Jam, Milk
3	Bread, Jam, Milk
4	Bacon, Bread, Corn, Egg, Milk
5	Bacon, Bread, Corn, Egg, Jam, Milk
6	Bread, Corn, Jam, Milk
7	Bacon, Bread, Egg, Milk
8	Bacon, Bread, Egg, Jam, Milk
9	Bread, Jam, Milk
10	Bacon, Egg, Milk

TABLE 2. ASSOCIATION RULE

Rule ID	Rule
1	{Jam}→{Bread}
2	{Jam, Milk}→{Bread}
3	{Jam}→{Bread, Milk}
4	{Bacon}→{Egg}
5	{Bacon, Milk}→{Egg}

For example consider transaction no 2, using formula

$$od(t) = \frac{|t^+ - t|}{|t^+|}.$$

Outlier degree=6-4/6=2/6=0.33

Consider min_od =0.3

According to defination , An outlier transaction is a transaction that its outlier degree od(t) is greater than or equal to min_od, a predefined outlier degree threshold where min_od is minimal outlier degree.

As 0.33>0.3.

Transaction no 2 is a outlier transaction.Similarly for transaction no 10,Outlier degree=6-3/6=3/6=0.5>0.3

Therefore transaction 3 is a outlier transaction.Thus in above dataset transaction no 2 and 10 are outlier transactions.

B. Improved Method

These algorithm is used to detect the item which induces a transaction to become abnormal. Here few changes are made to original algorithm.

Steps:

1. Get association rules set R from a transaction dataset T
2. Get outlier transactions set OT
   for each transaction t in T
   get t's associative closure by checking R
   calculate outlier degree od(t)
   if od(t) >= min_od then
   OT = OT ∪ {t}
3. Transfer transaction dataset T to T_{trans} by dividing T into unobserved frequent itemset and infrequent itemset.

Example :

TABLE 3. TRANSACTION DATASET

TID	Items
1	c, d, f, g
2	a, b, c, d, e, g
3	a, c, d, f
4	c, d, h, i
5	d, e, f, h
6	a, c, d, f, e, g
7	b, c, d, e, f
8	b, c, f, e, i
9	c, d, e, f, g, i
10	b, c, d, f
11	a, b, c, d
12	b, g
13	c, d, f, h
14	b, d, f, h
15	b, c, d, f
16	c, d, f, g

TABLE 4. ASSOCIATION RULE DERIVED FROM ABOVE TRANSACTION DATASET

Rule ID	Rule (confidence)
1	c → d (92.3%)
2	d → c (85.7%)
3	c, f → d (90.0%)
4	d, f → c (81.8%)

TABLE 5. TRANSFORMED TRANSACTION DATASET WITH UNOBSERVED PATTERN AND INFREQUENT ITEMSET

TID	Items
5	dc*, dfc*, e, h
8	cd*, cfd*, e, i
14	dc*, dfc*, h

In the above transaction dataset, transaction no 5,8,14 are outlier transactions.Here dfc*,cfd* are unobserved patterns and it is observed that 'c' should appear in transaction,but because of infrequent item 'h',it does not appear.Thus 'h' causes transaction to become outlier.

C. Frequent Pattern based Outlier Detection Method

Frequent pattern based outlier detection method is a new method to detect outlier by discovering frequent patterns from dataset.The outliers are defined as the data transactions that contain less frequent patterns in their itemsets. a measure called FPOF (Frequent Pattern Outlier Factor)[3] is used to detect the outlier transactions and the FindFPOF algorithm to discover outliers. [3]

Definition 1: (*FPOF-Frequent Pattern Outlier Factor*) Let $D = \{t_1, t_2, \ldots, t_n\}$ be a database containing a set of n transactions with items I. Given a threshold *minisupport*, $FPS(D, minisupport)$ is the set of all frequent patterns. For each transaction t, the *Frequent Pattern Outlier Factor of t* is defined as:

$$FPOF(t) = \frac{\sum support(X)_{X \subseteq t, X \in FPS(D, minisupport)}}{\| FPS(D, minisupport) \|} \quad (1)$$

A transaction t contains more frequent patterns, its FPOF value will be big, which indicates that it is unlikely to be an outlier. In contrast, transactions with small FPOF values are likely to be outliers. Obviously, the FPOF value is between 0 and 1.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Definition 2: For each transaction t, an itemset X is said to be *contradictive* to t if $X \not\subset t$. The *contradict-ness* of X to t is defined as:

$$Contradict\text{-}ness\ (X,\ t) = (||X|| - ||\ t\ \cap\ X||) * support\ (X)$$

Firstly, the greater the support of the itemset X, the greater the value of contradict-ness of X to t, since a large support value of X suggests a big deviation. With definition 2, it is possible to identify the contribution of each itemset to the outlying-ness of the specified transaction.

Algorithm: FindFPOF

Input: D // the transaction database

minisupport // user defined threshold for the permissible minimal support

top-n // user defined threshold value for top n fp-outliers top-k // user defined threshold value for top k contradict frequent patterns

Output: The values of FPOF for all transactions //indicates the degree of deviation The top-n FP-outliers with their corresponding TKCFPs

01 begin
02 Mining the set of frequent patterns on databaseD using minisupport
03 /* the set of all frequent patterns is donated as: FPS (D, minisupport) */
04 foreach transaction t in D do begin
05 foreach frequent pattern X in FPS (D, minisupport) do begin
06 if t contains X then
07 FPOF (t) = FPOF (t)+ support (X)
08 end if
09 end
10 f=FPOF(t)/FPS(D,minisupport)
11 return f
12 end
13 return top k outliers that minimize FPOFoutlierScore
14 end

Using the outlier factor FPOF, we can determine the degree of a record's deviation. The algorithm first gets the frequent patterns from the database using an existing association rule mining algorithm with a given minisupport . Then, for every transaction in the database, the value of FPOF is computed according to Definition 1 . Finally, the top-n FP-outliers are output with their corresponding top-k contradict frequent patterns.The FindFPOF algorithm has three parts: 1) mining the frequent patterns from the database; 2) computing the value of FPOF for each transaction; and 3) finding the top k outliers with minimum FPOF score.

The computational cost of the frequent-pattern mining algorithm is donated as O (FP). We remark that many fast frequent-pattern mining algorithms are available [19-21] and so the computation complexity of Part 1 will be acceptable. As to Part 2, two "for loops" are required. Therefore, the computational cost of this part is O (N*S), where N is number of the transactions in the database and S is the size of the set of frequent patterns. The overall computational complexity of the FindFPOF algorithm is: O(FP+N*S)

Example: Consider a ten-record customer dataset. We are interested in dimensions Age-range, Car, and Salary-level, which are useful for analyzing the latent behavior of the customers. Assume that the minimum support is set to 0.5, we can get the set of all frequent patterns as shown in Table 7

TABLE 6. CUSTOMER DATA

RID	Age-Range	Car	Salary-level	FPOF	Top 1 Contradict Frequent Patterns
1	Middle	Sedan	Low	0.35	{Young}, {High}
2	Middle	Sedan	High	0.35	{Young}, {High}
3	Young	Sedan	High	0.27	{Middle, Sedan}, {Low}, {Middle}
4	Middle	Sedan	Low	0.35	{Young}, {High}
5	Young	Sports	High	0.17	{Middle, Sedan}
6	Young	Sports	Low	0.17	{Middle, Sedan}
7	Middle	Sedan	High	0.35	{Young}, {Low}
8	Young	Sports	Low	0.17	{Middle, Sedan}
9	Middle	Sedan	High	0.35	{Young}, {Low}
10	Young	Sports	Low	0.17	{Middle, Sedan}

TABLE 7. FREQUENT PATTERN

ID	Age-Range	Support
1	{Middle}	0.5
2	{Young}	0.5
3	{Sedan}	0.6
4	{Low}	0.5
5	{High}	0.5
6	{Middle, Sedan}	0.5

1. For transaction no 1 FPOF is calculated using def 1 as: FPOF()=(0.5+0.5+0.5+0.6)/6=2.1/6=0.35
2. The transactions with less FPOF values contains less frequent patterns and are more likely to be outlier. Thus transactions 3,5,6,8,10 are considered as outlier transaction wrt FPOP values.

IV. EXPERMENTAL RESULTS

In these section we are comparing two transaction specific methods association rule based and frequent pattern based outlier detection method and find efficient strategy for detecting outlier transactions.

The figure shown below gives the accuracy comparison for Synthetic. We have given (min sup, min conf) = (0.05%, 70%) as the proper parameter set for AR Method, while min sup= 0.025% is the most proper for FI Method. Then, our method obviously outperforms FI Method. The best Fmeasure is recorded as 69.3% when d rate = 79.8% and d prec = 61.2% for AR Method; for FI Method it is 5.7% when d rate = 25.7% and d prec = 3.2%. FI Method can rarely detect true outliers in Synthetic. This is because associations between items in Synthetic are not very strong and are insufficient for only FIs to extract regularities and features from this data. Thus our method can detect more true outliers. AR Method has good detection performance.

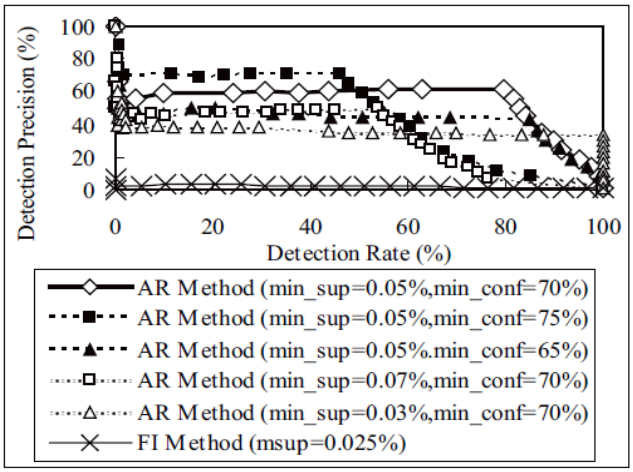

Figure 4. Accuracy Comparison

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

$$d\_rate = \frac{\text{\# of detected true outliers}}{\text{\# of all true outliers}}$$

$$d\_prec = \frac{\text{\# of detected true outliers}}{\text{\# of detected transactions as outliers}}$$

FindFPOF algorithm efficiency deteriorate with increase of large frequent itemsets generated as compared to association rule based method. Another drawback of FindFPOF algorithm is the speed of detecting the outliers in the data set. First for discovering the frequent patterns in the data set, at least two scans are needed and then after that, the data set needs to be scanned again to calculate the value of the FPOP measurement for each data object. Second for the approaches which are similar to FPOP measurement still targets the entire data set.

A drawback of the Find FPOF algorithm is that the size of the extracted frequent patterns is huge, because FindFPOF algorithm finds all the frequent patterns in the data set. Another drawback is, using the Apriori algorithm[6] as algorithm for finding the frequent patterns, which is time-consuming.

The table below depict the outliers detected by each algorithm using the real datasets.AR is having higher accuracy than FPOF method (for k=40, FPOF detects 31 outtliers and AR method detects 33).

TABLE 8. ACCURACY COMPARISON

(a) Breast Cancer

k	Greedy	AVF	FPOF	AR
4	4	4	3	3
8	8	7	7	7
16	15	14	14	15
24	22	21	21	21
32	29	28	27	28
40	33	32	31	33
48	37	36	35	37
56	39	39	39	39

TABLE 6.2 **Runtime in seconds for the simulated datasets with varying data size, *n*, from 1K to 800K data points**

Data Size *(thousands)*	AR	AVF	FPOF	Greedy
1	0.27	0.00	0.81	4.58
10	2.72	0.03	8.13	44.72
30	8.53	0.06	24.02	134.30
50	14.31	0.09	40.19	222.88
100	26.42	0.19	81.06	445.39
200	52.75	0.39	165.08	891.28
300	79.39	0.58	241.61	1337.06
400	106.14	0.80	323.97	1781.78
500	131.75	0.94	404.45	2233.74
600	158.70	1.16	484.00	2678.73
700	184.94	1.33	564.80	3127.22
800	212.08	1.56	667.55	3568.55

A framework with improvement in AR method is used to identify outlier transactions as well as find the item that induces the transaction to become outlier.

V. CONCLUSION

In this paper, we target transaction databases and propose the detection of transactions that are likely to be outliers. Defining the concept of associative closure using association rules with high confidence, we derive a formula for outlier degree. Thus from these paper we conclude that the transaction specific method Association rule based outlier detection method is more efficient than frequent pattern outlier detection method using FPOF score for outlier transaction detection.

Speed of FindFPOF method is slow as first for discovering the frequent patterns in the data set. FindFPOF method uses the Apriori algorithm as algorithm for finding the frequent patterns, which is time-consuming. Association rule based method is having higher accuracy than FindFPOF method as shown from the experimental results. Since infrequent items will induce incorrect calculation on outlier degrees, the proposed method modified the definition of transaction's association closure by removing the infrequent items before the calculation of outlier degrees. The experimental results provide evidences to verify that the proposed algorithm is more efficient in both accuracy and precision rates.Thus association rule based method is more efficient than frequent pattern based outlier detection method.

REFERENCES

1. Li-Jen Kao, Yo-Ping Huang*, "An Efficient Strategy to Detect Outlier Transactions for Knowledge Mining."IEEE 2011
2. K. Narita and H. Kitagawa, "Outlier detection for transaction Databases using association rules," in Proc. of the 9th Int. Conf. on Web-Age Information Management, Zhangjiajie, Hunan, pp.373-380, July 2008
3. Z. He, X. Xu and S. Deng, "Fp-outlier: Frequent pattern based outlier detection," Computer Science and Information System, vol. 2, no. 1, pp.103-118, June 2005.
4. Koufakou1 E.G. Ortiz1 M. Georgiopoulos1 G.C. Anagnostopoulos2 K.M. Reynolds, "A Scalable and Efficient Outlier Detection Strategy for Categorical Data" IEEE 2007
5. Li-Jen Kao, Yo-Ping Huang*, "Association Rules Based Algorithm for Identifying Outlier Transactions in Data Stream", IEEE 2012
6. J. Han, J. Pei and Y. Yin, "Mining frequent patterns without candidate generation," in Proc. of ACM SIGMOD Int. Conf. on Management of Data, Dallas, Texas, USA, pp.1-12, May 2000.
7. VARUN CHANDOLA, Outlier Detection : A Survey
8. Hans-Peter Kriegel, Peer Kröger, Arthur Zimek, Outlier Detection Techniques
9. S. Ramaswamy, R. Rastogi and K. Shim, "Efficient algorithms for mining outliers from large data sets," in *Proc. of ACM SIGMOD Int. Conf. on Management of Data*, Dallas, Texas, USA, pp.427-438, May 2000.
10. K. Das and J.G. Schneider, "Detecting anomalous records in categorical datasets," in *Proc. of the Second Int. Conf. on Knowledge Discovery and Data Mining*, San Jose, California, USA, pp.220-229, August 2007.
11. D. Burdick, M. Calimlim and J. Gehrke, "Mafia: A maximal frequent itemset algorithm for transactional databases," in *Proc. of the 17th Int. Conf. on Data Engineering*, Heidelberg, Germany, pp.443-452, April 2001.
12. R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *VLDB*, pages 487–499, 1994.
13. A. Arning, R. Agrawal, and P. Raghavan. A linear method for deviation detection in large databases. In *KDD*, pages 164–169, 1996.
14. G. Grahne and J. Zhu. Efficiently using prefix-trees in mining frequent itemsets. In *FIMI*, 2003.
15. K. Narita and H. Kitagawa. Detecting outliers in categorical record databases based on attribute associations. In *APWeb*, 2008.
16. Dr. Shuchita Upadhyaya, Karanjit Singh, "Classification based outlier detection techniques"
17. Knorr E.M., Ng R.T., Tucakov V., "Distance based method: algorithm and applications"
18. Sridhar Ramaswamy, "Efficient algorithms for mining outliers from large datasets"[19] Rajendra Pamula,"Outlier detection method based on clustering"