Spotting Outliers in Large Distributed Datasets using Cell Density based Approach

A.Rama Satish
Associate Professor, Dept. of CSE
DVR & Dr HS MIC College of Technology
Kanchikacherla,Krishna District,A.P., India

Dr.P.Bala Krishna Prasad
Principal
Eluru College of Engineering & Technology
Eluru , Krishna Distrct, A.P., India

ABSTRACT

Outliers are abnormal instances or observations. Detecting data outliers is a very important concept in Knowledge data discovery. Outlier detection has been studied in the context of a large number of research areas like large distributed systems, data mining, wireless sensor networks(WSN), health monitoring, environmental science, statistics, etc., Density based (DB) outlier detection techniques are robust in detecting outliers. In many applications, too much voluminous distributed data is generating every day. Finding deviating observations in the large distributed database rather than in any individual database is not a simple task. Integrating distributed database cause two major problems. First, render massive data from different databases. In addition, data integration may cause violation of data security and leakage of sensitive information. In this work we propose cell density based mechanism for outlier detection **(CDOD)** in large distributed databases. A centralized detection paradigm is used; it allows overcoming the expensive data integration and information leakage. The experimental results show robustness for finding outliers in large number of databases, instances and attributes.

Keywords

Data Mining, KDD,Large distributed databases, Density based outlier detection.

1. INTRODUCTION

Outlier detection is great significant research problem in data mining. This mainly aims to detect a specific number of data objects that are considerably dissimilar, exceptional and inconsistent with respect to the majority records in the input databases[3, 6]. Outliers arise due to machine level errors, changes in system behaviour, fraudulent behaviour, human errors, system faults, or simply through natural deviations in populations. Detection of potential outliers is important for identifying the errors and removes their contaminating effect on the dataset to make the data clean for processing. Outlier detection methods can be classified between univariate methods and multivariate method. Different approaches are devised based on different assumptions to detect outliers. The best way of detecting outliers in distributed databases is global versus local outlier detection approach. All data objects are considered as reference set in global approaches but the local approaches contains a (small) subset of data objects. The general design of outlier detection technique contains the primary ingredients of nature of data, outlier detection technique, knowledge disciplines, application domains, finally outliers.

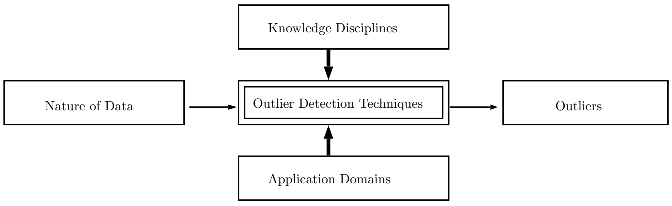

Fig. 1. A General Design of an Outlier Detection Technique

Figure 1 illustrated that any outlier detection technique has the following primary ingredients:

—Nature of data, nature of outliers, and other constraints and assumptions that collectively constitute the problem formulation.

—Application domain in which the technique is applied. Some of the techniques are developed in a more generic fashion but are still feasible in one or more domains while others directly target a particular application domain.

—The concept and ideas used from one or more knowledge disciplines.

In many applications, too much voluminous distributed data is generating every day. The increase in number of applications it is necessary to collect and store a large amount of data in multiple proprietary or distributed databases for knowledge discovery. Credit card transactions are scattered across a number of distributed community data centres[18]. Detecting irregular credit card spending patterns is the best example for outlier detection in large distributed database. These kinds of abnormalities are called *global outliers*. Integrating distributed database cause two major problems. First, render massive data from different databases. In addition, data integration may cause violation of data security and leakage of sensitive information. Finding deviating observations in the large distributed database rather than in any individual database is not a simple task.For the past decades, most of the existing outlier detecting research work is focused on the centralized outlier detection mechanism where all the data are stored and processed in a central manner. Optimizing or boosting techniques are required to reduce the communication overhead among the sites.

Data partitioning is the answer to improve the performance of outlier detection and it fundamentally impacts the outlier detection techniques in outliers detecting process.There are two primary approaches *horizontal partitioning* and *vertical partitioning* in data partitioning of various distributed applications. Practically horizontal partitioning is the common type of data partitioning in many real-life applications which is mainly concentrated by the researcher. The host and process of horizontal partitioning different subsets of data have the same schema. The global database will form by the union of the database of all distributed sites. The above definition of global database can be proved in mathematical language, $D = D_1 \cup D_2 \cup \cdots \cup D_t$. The global database D is formed by the union of individual or local databases $D_1, D_2, \cdots, D_t$. In this case we have chosen independent and identical distributed *(IID)* data at different sites.

The problem of detecting global outliers based on cell density in a distributed environment can be formally formulated as follows. Let consider the global database D that is horizontally partitioned into t parts (i.e., $D = D_1 \cup D_2 \cup \cdots \cup D_t$) each data subset residing at a distributed processing site, represented respectively by $s_1, s_2, \cdots, s_t$. Map data into cell grid by taking equal interval. If the cell count is high then finding global outliers will become expensive task. To reduce the complexity and make it inexpensive, again map all the cells into master grid according to cell density. Find the global data summary. Find the global dense master cells that are to be declaring as possible cells of global outliers called global master cell. A data point p in master global cell is detected and returned as one of the final top n global outliers (n) is a user specified parameter, if there does not exist more than $(n-1)$ other data points p_i in master global cell that satisfy that $f(p) < f(p_i)\,(0 < i < n-1)$. Here, $f(\cdot)$ denotes the outlier-ness score function that is used to quantitatively measure the strength or probability of p for being an outlier and may have different application-dependent definitions. When dealing with this problem, high numbers of data instances are processed in a simple manner. Data integration is expensive and is thus disallowed.

To reduce the complexity and make finding global outliers in distributed databases is expensive we propose a new system, called CDOD (short for **C**ell **D**ensity based **O**utlier **D**etection). In this paper, we present a work addressing the above challenges. Specifically, we make the following brief notes that show contributions of CDOD:

—CDOD is an effective global outlier detection method in distributed databases.

—Mapping data based on their densities reduces the complexity of outlier detection in distributed databases.

—It is devised in a centralized environment without sensitive information leakage.

—Experimental results show that the efficiency of CDOD in finding global outliers in terms of large number of attributes instances and distributed sites is outstanding.

—It is efficient and ideal to deal with massive distributed databases.

RoadMap The sections in this research paper are organized as follows: Section 2 will describe literature survey, section 3 provides proposed solution. Experimental results of conducted tests on selected databases are reported in section 4 followed by the conclusion in section 5.

2. LITERATURE SURVEY

Majority of outlier detection techniques are categorized based on the distribution-based methods, the distance-based methods, the density-based methods, and the clustering-based methods. These techniques are developed for analysing centralized database.

Distribution-based approach is a statistical approach that assumes distribution or probability model for the given dataset and identifies outliers with respect to the model using a discordance test. A discordance test is used to detect whether a given object is an outlier or not. Statistical techniques are the best use if the data contains significant outliers. In this case we may need to consider the use of robust statistical techniques for outlier detection. The concept of proximity of object is considered in distance based approaches[9, 10, 14]. An object is said to be an outlier if the nearest neighbors (NN) of the object are far away. So the proximity of object is significantly deviates from the proximity of most of the other objects in the same data set. These outlier detection techniques have emerged as a scalable, viable, parameter-free alternative to the more traditional statistical approaches. In density-based methods[4, 8, 15, 19, 17] usually it is necessary to investigating not only the local density of the data being studied but also the local densities around its neighbors, which become more complex mechanism to model outlier-ness of data point. In an appropriate data representation, normal data is expected to cluster, and outliers are expected to be further away from the normal data. Clustering-based methods[1, 5, 7, 12, 16, 20], an object is said to be outlier if it does not belongs to any cluster, there will be a large distance between object and its closest cluster, and it belongs to small and sparse cluster.

The above mentioned techniques are devised for centralized database analysis. They are failed to handle the distributed databases. Popular distributed database (DDB) definition is, it is a collection of multiple, logically interrelated databases distributed over a computer network. Recently a lot of research has focused on distributed database analysis, but still very limited, research work aiming to develop new outlier detection techniques exclusively for handling large distributed databases is discussed as follows.

Fabrizio Angiulli, described a distributed approach which addressing the distance based outlier detection in very large data sets and named it as *Distributed Solving Set*[2] based on the outlier detection solving set. In this researcher considered, a subset S of the data set D as the outlier detection solving set that includes sufficient number of objects from D. This makes top n outlier detection process easier by considering only the distances among the pairs in $S * D$. A novel object q is chosen to predict, if it is an outlier or not by comparing novel object q only with other objects in solving set S instead of considering all other objects in dataset D. Subset S contains at least the top n outliers and simultaneously solves outlier detection test.

Yaling Pei, Osmar R. Zaiane, proposed a new approach the basic idea of this method is to rank the data points based on their relative degree of density with respect to a fixed set of reference points. So the proposed outlier detection[13] that uses the relative degree of density with respect to a fixed set of reference points to approximate the degree of density defined in terms of nearest neighbours of a data point.

The researchers Muruganantham, AnkitaDubey mainly focused on detecting outlying observations in large database[11] to fond top n-distance-based outlying objects in the database. The objects detected as outliers having weight not smaller than the n^{th} largest weight, where the weight of a data set objects computed as the sum of the distances from the object to its k nearest neighbours.

Ji Zhang, tao, wang proposed new technique DISTributed OutlieR Detection[18] for global outlier detection from large distributed databases. Centralized detection approach is used which prevents information leakage. Implemented optimization enhancement strategies to speed up outlier detection process and reduce communication overhead. They have shown that the experimental results show the good performance.

The above mentioned research works are developed for finding outliers in centralized and distributed databases. Few algorithms [11, 18] are introduced for large distributed databases. There have been some research works on distributed outlier detection, in the sense of increase in attributes, instances and distributed sites.

3. PROPOSED WORK

In this section, we present our proposed **C**ell **D**ensity based **O**utlier **D**etection approach, for detecting user requested top n global outliers from large distributed databases. This section mainly contains the following: step wise description, diagrammatic representation and pseudo code of CDOD.

3.1 Step Wise Description of CDOD

Step 1 *Superimpose the data into cell grid (for distributed sites)* The incoming data is mapped into one appropriate cell in the grid; if the data falls into a cell that is not yet in the grid, then a new cell will be added into the grid. Index number will be changed. Density of the cell increased when a new data point is assigned to it. The major purpose of assigning data into grid structure is to obtain the density information of cells in the grid.

Step 2 *Superimpose the cells in grid into master grid (for distributed sites)* Superimpose the above created cells into master grid according to their densities. If the cell is falls into master cell that is already exist then mapping can be done otherwise a new master cell will be added into the master grid. The density of master cell increased when a new data is assigned to it. The major purpose of assigning cells into master grid structure is to obtain the local data summary.

Step 3 *Generating the global data summary (for mediator)* After completion of above step each dataset is partitioned into grid to obtain the global data summary. Cell index and cell density information are first transmitted to the mediator. So this prevents the data security violation. The mediator aggregates the density information of each master cell. i.e.,

$$density\left[mastercell\right] = \sum_{s=1}^{T} density\left[mastercell\right]^{s}$$

Each master cell generated in step 2 of each site is called as global populated cell if the following condition is true.

$$density\left[mastercell\right] > global\_avg\_density$$

where

$$global\_avg\_density = N/N_{populated\_master\_cells}$$

$N =$ sum of number of data points in each database.

$N_{populated\_master\_cells}$ = Sum of number of populated master cells in each database that contain atleast one element.

Step 4 *Generating user requested top n local outliers (for distributed sites)* In each local database,for each data object in master grid k_ODF is calculated. i.e.,

$$k\_ODF\left(p\right) = \sum_{i=1}^{k} Dist\left(p, centroid\left(C\right)\right)/k$$

Where, k is the nearest global dense cell to object p, k_ODF is k-outlier degree factor which measures the strength of outlier-ness of each data point in each global populated cells. Sort the data objects according to their k_ODF values, generate and transmit top n local outliers to the mediator.

Step 5 *Generate user requested top n global outliers (mediator)* Mediator generates the top n global outliers from collected top n local outliers by merging them and they are returned to the end user as requested top n global outliers.

In this algorithm, it is clear that there is no communication between sites and users. User directly send request to the mediator. The mediator acts as intermediate node which finally generates the user requested top n global outliers.

3.2 Diagramatic Representation of CDOD

Figure 2 shows the detection of user requested top n global outlying observations in a distributed environment. It mainly consists two parts. First one is for distributed sites; second one is for mediator.

The main objective of **CDOD** is to find the global outliers without information leakage. All distributed sites must have the computation capability. In first step for each distributed site, map the data objects into a grid structure. When a new data object is assigned then change index number and density of cell and map formed cells into master grid structure by considering the density information of each cell in all sites. The main purpose of mapping data into grid is it enables fast processing and takes less time to handle data in cell. Processing and accessing time mainly depends on the number of cell and quantized space. The density information is transmit to the mediator for finding global data summary. At mediator side, mediator calculates global average density of global database and densities of master cells in all sites for finding global dense master cell. The global dense master cell has the possible global outlying objects. Find the centroid of global dense master cell and transmit it to each distribute site. For each distributed site calculate the k_ODF (outlier degree factor) for each object in global populated master cells by finding the k-nearest centroids.

Sort and transmit the k_ODF information along with corresponding data object to the mediator. At mediator side, generate top n global outliers by merging them. Return the top user requested n global outliers to the user.

3.3 Pseudo Code

Pseudo code given in Algorithm 1 clearly shown that what happens at the mediator side and the distributed sites side. In this each site and mediator have the computation capability. Lines 1-25 and 38-49 represents the code which is executed at distributors site. Lines 26-37 and 50-51 represents the code executed at mediator site.

4. EXPERIMENTAL RESULTS

To evaluate our proposed global outlier detection technique, both synthetic and real-life datasets are used. Experiments are performed on 20 PCs with 3.0 GHz CPU and 4 GB RAM, which runs Windows 7 Ultimate OS to construct distributed environment initially. MATLAB R2013a is used to devise proposed algorithm. Our experiments focus on testing the scalability and effectiveness

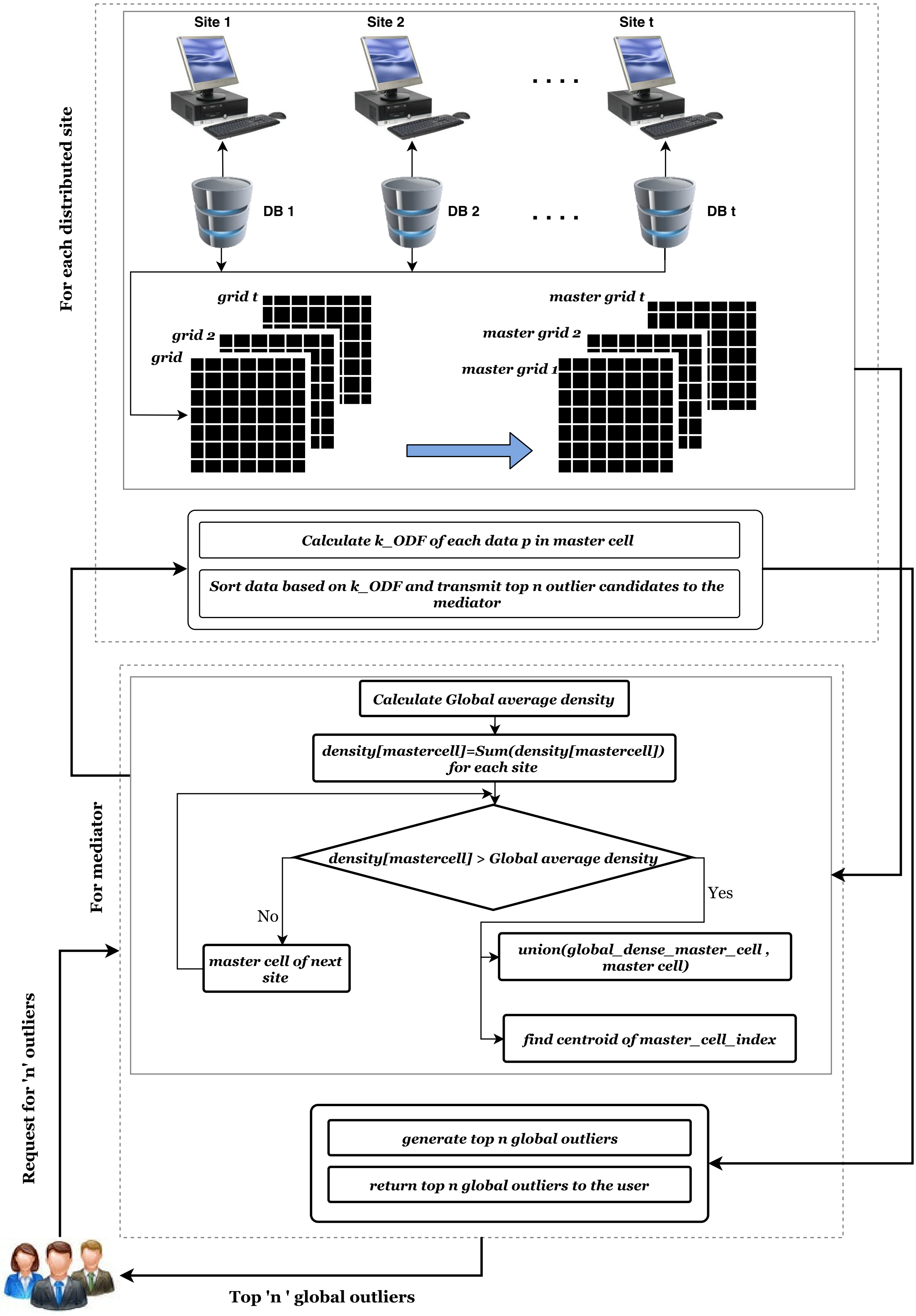
For each distributed site
Site 1
Site 2
Site t
DB 1
DB 2
DB t
grid t
grid 2
grid
master grid t
master grid 2
master grid 1
Calculate k_ODF of each data p in master cell
Sort data based on k_ODF and transmit top n outlier candidates to the mediator
For mediator
Calculate Global average density
density[mastercell]=Sum(density[mastercell]) for each site
density[mastercell] > Global average density
No
Yes
master cell of next site
union(global_dense_master_cell , master cell)
find centroid of master_cell_index
generate top n global outliers
return top n global outliers to the user
Request for 'n' outliers
Top 'n ' global outliers

Fig. 2. Diagramatic Representation of CDOD

Algorithm 1 Pseudo code for CDOD

```
Input: Distributed databases $D_1, D_2, ..., D_t$ at distributed sites $S_1, S_2, ..., S_t$, respectively. Request for top ’n’ global outliers.
Output: Top ’n’ global outliers.
disp(’top n data objects identified as outliers’) {for distributor sites begin}
n=str2double(input(’input n’,’s’))
for s=1:t do
    load data from site s
    for each data point $p\epsilon s$ assign into cell in grid s do
        if cell already exists then
            map
        else
            create new cell and map
        end if
        $density\,[cell\_index]^s = density\,[cell\_index]^s + 1$
    end for
end for
for s=1:t do
    for each $[cell\_index]^s\ \epsilon\quad grid$ ’s’ do
        according to density, assign each $[cell\_index]^s$ into master_cell in master grid s
        if master_cell exist then
            map
        else
            create new master_cell and map
        end if
        $density\,[master\_cell\_index]^s = density\,[master\_cell\_index]^s + 1$
    end for
    transmit density information to the mediator
end for{for distributor sites end}
$global\_dense\_master\_cell = []$ {for mediator begin}
for s=1:t do
    for each populated master_cell $[master\_cell\_index]$ do
        $density\,[master\_cell] = \sum density\,[master\_cell]^s$
    end for
    $global\_avg\_density = N/N_{populated\_master\_cell}$
    if $density\,[master\_cell] > global\_avg\_density$ then
        $global\_dense\_master\_cell = union\,(global\_dense\_master\_cell, [master\_cell])$
        $centroid\,(master\_cell\_index)^s = mean\,(master\_cell\_index)$
        transmit centroid of master cell to each site s
    end if
end for{for mediator end}
{for distributor sites begin}
for s=1:t do
    for each data $p\epsilon\,[master\_cell\_index]$ do
        $temp = []$
        for j=1:k do
            $distance = dist\,(p, centroid\,([master\_cell\_index]^s))$
            $temp = [temp; distance]$
        end for
        k_ODF(p)=sum(temp)
    end for
    k_ODF(p)=sort_fun(k_ODF(p))
    transmit top n outlier from the sorting list from s to the mediator
end for{for distributor sites end}
{finally mediator return top n outliers}
merge and generate the top n outliers
return the top n global outliers
```

of CDOD. To test on synthetic datasets, we deploy a data generator to produce datasets with a controlled number of outliers which consists large number of attributes and data instances. The KDD Cup99 network intrusion detection dataset from online free UCI Machine learning data repository is used as real-life dataset. In all cases the execution time and number of attributes, instances and distributed sites are recorded for testing efficiency and scalability. The following tables presents the data objects along with their

Table 1. Merged data objects along with their k_ODF values

Site 1				. . .	. . .	**Site t**				. . .
Attr 1	**k_ODF**	**Attr 2**	**k_ODF**	. . .	. . .	**Attr 1**	**k_ODF**	**Attr 2**	**k_ODF**	. . .
0.200	2.2417	2.8000	0.2208	. . .	. . .	0.200	2.2417	2.8000	0.2208	. . .
4.800	1.4000	4.9000	1.5000	. . .	. . .	4.800	1.4000	4.9000	1.5000	. . .
3.100	0.4292	2.0000	0.4417	. . .	. . .	3.100	0.4292	2.0000	0.4417	. . .
1.600	0.8417	7.7000	4.3000	. . .	. . .	1.600	0.8417	7.7000	4.3000	. . .
0.200	2.2417	2.8000	0.2208	. . .	. . .	0.200	2.2417	2.8000	0.2208	. . .
5.400	2.0000	6.7000	3.3000	. . .	. . .	5.400	2.0000	6.7000	3.3000	. . .
3.400	0.4000	2.0000	0.4417	. . .	. . .	3.400	0.4000	2.0000	0.4417	. . .
1.500	0.9417	6.3000	2.9000	. . .	. . .	1.500	0.9417	6.3000	2.9000	. . .
0.400	2.0417	2.7000	0.1875	. . .	. . .	0.400	2.0417	2.7000	0.1875	. . .
5.200	1.8000	4.9000	1.5000	. . .	. . .	5.200	1.8000	4.9000	1.5000	. . .
. .	. .	. .	. .	. .	. .	. .	. .	. .	. .	. .

Table 2. User requested top n global outliers along with k_ODF values

Site 1				. . .	. . .	**Site t**				. . .
Attr 1	**k_ODF**	**Attr 2**	**k_ODF**	. . .	. . .	**Attr 1**	**k_ODF**	**Attr 2**	**k_ODF**	. . .
3.2000	2.3410	7.7000	4.3000	. . .	. . .	6.0000	2.6000	3.8000	2.5000	. . .
0.2000	2.2417	5.2100	3.8000			6.0000	2.6000	4.3000	2.4500	
0.2000	2.2417	6.7000	3.3000	. . .	. . .	5.4000	2.0000	5.8000	2.4000	. . .
0.4000	2.0417	6.3000	2.9000			3.2000	1.8000	5.7000	2.3000	
5.4000	2.0000	2.4100	1.8000	. . .	. . .	5.1000	1.7000	5.5000	2.1000	. . .
5.2000	1.8000	4.9000	1.5000			1.2000	1.2417	5.1000	1.7000	
4.8000	1.4000	4.9000	1.5000	. . .	. . .	4.5000	1.1000	5.0000	1.6000	. . .
2.3000	1.0060	2.0000	0.4417			1.5000	0.9417	2.0000	0.4417	
1.5000	0.9417	2.0000	0.4417	. . .	. . .	1.6000	0.8417	2.1000	0.3417	. . .
1.6000	0.8417	2.8000	0.2208			3.0000	0.3958	2.8000	0.2208	
..	..	..	..	..	..	..	..	..	..	..

k_ODF values. At each distributed site side, each distributed site generates local outliers and transmits them to the mediator to generate global outliers. The results provided in Table 1 several attributes and their k_ODF values of each distributed site are displayed. To generate user requested top n global outliers merge all the received local outliers and generate top n global outliers. The Table 2 provides the results of generated top n global outliers.

We measure the scalability of CDOD by taking large number of data instances and large number distributed sites. The number of data instances from site ranges from 500,000 to 50,00,000.The number of distributed sites ranging from 100 to 1000. Figure 4 and figure 5 show the scalability with respective to database size 2-D graph and ribbon graph respectively. Figure 6 and figure 7 show the scalability with respective to distribute sites 2-D graph and ribbon graph respectively. In the above mentioned four graphs it is clearly shown that the speed improvement while running CDOD.

5. CONCLUSION

In distributed environment, every data object in local database is requested every time when it is required in global outlier detection process. This may leads to data security violation problem. Centralized approaches in distributed data analysis, one dedicated site managing all, to generate global data summary is called mediator. Centralized approaches are not scalable and single point of failure. Proposed technique **CDOD** is applied to distributed datasets by increasing the number of attributes, data instances and number of distributed sites. The experimental results shows that proposed cell density based outlier detection in large distributed

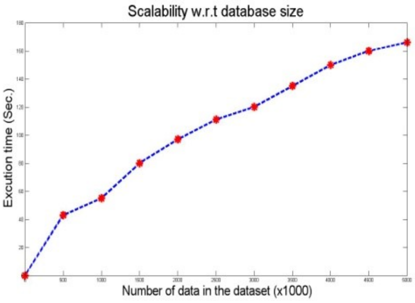

Fig. 3. Scalability w.r.t database size in 2-D graph

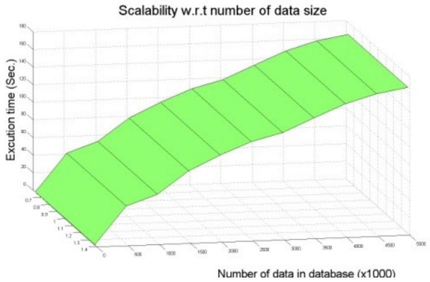

Fig. 4. Scalability w.r.t database size in ribbon graph

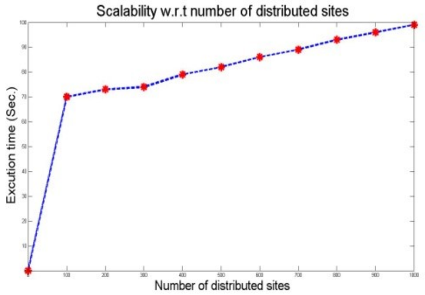

Fig. 5. Scalability w.r.t distributed sites in 2-D graph

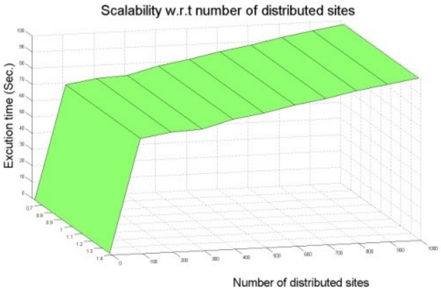

Fig. 6. Scalability w.r.t distributed sites in ribbon graph

environment for detecting user requested top n global outlier is effective and scalable.

6. REFERENCES

[1] Rakesh Agrawal, Johannes Gehrke, Dimitrios Gunopulos, and Prabhakar Raghavan. *Automatic subspace clustering of high dimensional data for data mining applications*, volume 27. ACM, 1998.

[2] Fabrizio Angiulli, Stefano Basta, Stefano Lodi, and Claudio Sartori. A distributed approach to detect outliers in very large data sets. In *Euro-Par 2010-Parallel Processing*, pages 329–340. Springer, 2010.

[3] Vic Barnett and Toby Lewis. *Outliers in statistical data*, volume 3. Wiley New York, 1994.

[4] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *ACM sigmod record*, volume 29, pages 93–104. ACM, 2000.

[5] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.

[6] Douglas M Hawkins. *Identification of outliers*, volume 11. Springer, 1980.

[7] Alexander Hinneburg and Daniel A Keim. An efficient approach to clustering in large multimedia databases with noise. In *KDD*, volume 98, pages 58–65, 1998.

[8] Wen Jin, Anthony KH Tung, and Jiawei Han. Mining top-n local outliers in large databases. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 293–298. ACM, 2001.

[9] Edwin M Knorr and Raymond T Ng. Finding intensional knowledge of distance-based outliers. In *VLDB*, volume 99, pages 211–222, 1999.

[10] Edwin M Knox and Raymond T Ng. Algorithms for mining distancebased outliers in large datasets. In *Proceedings of the International Conference on Very Large Data Bases*, pages 392–403. Citeseer, 1998.

[11] Ankita Dubey Muruganantham B. Outlier detection using distributed mining technology in large database. *International Journal of Computer Science and Engineering*, 2(2):6–11, 2015.

[12] Raymond T Ng and Jiawei Han. Efficient and effective clustering methods for spatial data mining. In *Proc. of*, pages 144–155, 1994.

[13] Yaling Pei, Osmar R Zaiane, and Yong Gao. An efficient reference-based approach to outlier detection in large datasets. In *Data Mining, 2006. ICDM'06. Sixth International Conference on*, pages 478–487. IEEE, 2006.

[14] Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. Efficient algorithms for mining outliers from large data sets. In *ACM SIGMOD Record*, volume 29, pages 427–438. ACM, 2000.

[15] Jian Tang, Zhixiang Chen, Ada Wai-Chee Fu, and David W Cheung. Enhancing effectiveness of outlier detections for low density patterns. In *Advances in Knowledge Discovery and Data Mining*, pages 535–548. Springer, 2002.

[16] Ji Zhang, Wynne Hsu, and Mong Li Lee. Clustering in dynamic spatial databases. *Journal of intelligent information systems*, 24(1):5–27, 2005.

[17] Ji Zhang, Meng Lou, Tok Wang Ling, and Hai Wang. Hos-miner: a system for detecting outlying subspaces of high-dimensional data. In *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pages 1265–1268. VLDB Endowment, 2004.

[18] Ji Zhang, Xiaohui Tao, and Hua Wang. Outlier detection from large distributed databases. *World Wide Web*, 17(4):539–568, 2014.

[19] Ji Zhang and Hai Wang. Detecting outlying subspaces for high-dimensional data: the new task, algorithms, and performance. *Knowledge and information systems*, 10(3):333–355, 2006.

[20] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch: an efficient data clustering method for very large databases. In *ACM SIGMOD Record*, volume 25, pages 103–114. ACM, 1996.