ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

Mining Of Inconsistent Data in Large Dataset In Distributed Environment

M.Shanthini[1]

Department of Computer Science and Engineering, Syed Ammal Engineering College, Ramanathapuram, Tamilnadu, India[1]

ABSTRACT: Introduce a distributed method for detecting distance-based outliers in very large data sets. This approach is based on the concept of outlier uncovering solving set, which is a slight subset of the data set that can also be engaged for foreseeing new outliers. It is to be used both in parallel and distributed scenarios. Due to the use of multiple processor hierarchy, each one has been worked independently. The method exploits parallel computation in order to obtain vast time savings. Certainly, afar preserving the perfection of the result, the suggested outline exhibits admirable concerts. Since the academic point of view, for shared settings, the time-based cost of our system is estimated to be at any rate of three orders of a magnitude faster than the classical nested-loop like approach to spot outliers. Tentative results demonstrate that the system is efficient and that it's running time scales quite well for an increasing number of nodes. It is also a variant of the basic strategy which reduces the amount of data to be transferred in order to improve both the communication cost and the inclusive runtime. Prominently, the solving set figured by our approach in a distributed environment has the same quality as that produced by the corresponding centralized method.

KEYWORDS: Distance-based outliers, outlier detection, parallel and distributed algorithms

I. INTRODUCTION

OUTLIER detection is the data mining task whose goal is to isolate the observations which are considerably dissimilar from the remaining data [11]. This task has practical applications in several domains such as fraud detection, intrusion detection, data cleaning, medical diagnosis, and many others. Unsupervised approaches to outlier detection are able to discriminate each datum as normal or exceptional when no training examples are available. Among the unsupervised approaches, distance-based methods distinguish an object as outlier on the basis of the distances to its nearest neighbors [15], [19], [6], [4], [2], [20], [9], [3]. These approaches differ in the way the distance measure is defined, but in general, given a data set of objects, an object can be associated with a weight or score, which is, intuitively, a function of its k nearest neighbors distances quantifying the dissimilarity of the object from its neighbors. In this work, we follow the definition given in [4]: a top-n distance based outlier in a data set is an object having weight not smaller than the nth largest weight, where the weight of a data set object is computed as the sum of the distances from the object to its k nearest neighbors. Many prominent data mining algorithms have been designed on the assumption that data are centralized in a single memory hierarchy. Moreover, such algorithms are mostly designed to be executed by a single processor. More than a decade ago, it was recognized that such a design approach was too limited to deal effectively with the issue of continuous increase in the size and complexity of real data sets, and in the prevalence of distributed data sources [22]. Consequently, many research works have proposed parallel data mining (PDM) and distributed data mining (DDM) algorithms as a solution to such issue [14].

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

Today, the arguments for developing PDM and DDM algorithms are even stronger, as the tendency toward generating larger and inherently distributed data sets amplifies performance and communication insufficiencies. Indeed, when applied to very large data sets, even scalable data mining algorithms may still require execution times
that are excessive when compared to the stringent requirements of today's applications. Parallel processing of mining tasks could dramatically reduce the effect of constant factors and decrease execution times. Moreover, in mining data from distributed sources, the data set is fragmented into many local data sets, generated at distinct nodes of a network. A widely adopted solution entails the transfer of all the data sets to a single storage and processing site, usually a data warehouse, prior to the application of a centralized algorithm at the site. The advantages of such a solution are simplicity and feasibility with established technology.

II. RELATED WORK

An "outlier detection" based on concept of outlier detection solving set. It is based on PDM/DDM approach for computing distance based outliers. In Parallel Data mining aims at finding meaningful patterns or rules in large datasets. It is an interdisciplinary field, which combines research from areas such as machine learning, statistics, high performance computing, and neural networks. The field of distributed data mining (DDM) deals with this problem mining distributed data by paying careful attention to the distributed resources. Analyzing and monitoring these distributed data sources require a data mining technology designed for distributed applications. A common feature of most data mining tasks is that they are resource intensive and operate on large sets of data.

Data sources measuring in gigabytes or terabytes are now quite common in data mining. In Distributed Solving Set algorithm splitting the data set into various subsets of data sets. Each object in the data set has been worked as like a distributed manner. In this environment, it has been assigned to multiple processor hierarchy.

Distance-Based Detection and Prediction of Outliers

A distance-based outlier detection method that finds the top outliers in an unlabeled data set and provides a subset of it, called outlier detection solving set, that can be used to predict the outlier of new unseen objects, is proposed. The solving set includes a sufficient number of points that permits the detection of the top outliers by considering only a subset of all the pair wise distances from the data set. The properties of the solving set are investigated, and algorithms for computing it, with sub quadratic time requirements, are proposed.

Experiments on synthetic and real data sets to evaluate the effectiveness of the approach are presented. A scaling analysis of the solving set size is performed, and the false positive rate, that is, the fraction of new objects misclassified as outliers using the solving set instead of the overall data set, is shown to be negligible. Finally, to investigate the accuracy in separating outliers from inliers, ROC analysis of the method is accomplished. Results obtained show that using the solving set instead of the data set guarantees a comparable quality of the prediction, but at a lower computational cost. It is not a sufficient method for detecting outliers in large database.

Fast Mining of Distance-Based Outliers in High-Dimensional Dataset

Defining outlier by their distance to neighboring data points has been shown to be an effective non-parametric approach to outlier detection. In recent years, many research efforts have looked at developing fast distance- based outlier detection algorithms. Several of these efforts report log linear time performance as a function of the number of data points on many real life low dimensional datasets.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

However, these same algorithms are unable to obtain the same level of performance on high dimensional data sets since the scaling behavior is exponential in the number of dimensions. In this paper we present RBRP, a fast algorithm for mining distance- based outliers, particularly targeted at high dimensional data sets. RBRP is expected to scale log-linearly, as a function of the number of data points and scales linearly as a function of the number of dimensions. Our empirical evaluation verifies this expectancy and furthermore we demonstrate that our approach consistently output forms the state-of-the-art, sometimes by an order of magnitude, on several real and synthetic datasets. It relates to the time consuming process.

Mining Distance based Outliers from Large Databases in Any Metric Space

It considers a generic version of the problem, where no information is available for outlier computation, except for objects mutual distance. Let R be a set of objects. An object o ∈ R is an outlier, if there exist less than k objects in R whose distances to o are at most r. The values of k, r, and the distance metric are provided by a user at the run time. The objective is to return all outliers with the smallest I/O cost.

Prove an upper bound for the memory consumption which permits the discovery of all outliers by scanning the dataset 3 times. The upper bound turns out to be extremely low in practice, e.g., less than 1% of R. Since the actual memory capacity of a realistic DBMS is typically larger, we develop a novel algorithm, which integrates our theoretical findings with carefully- designed heuristics that leverage the additional memory to improve I/O efficiency. Our technique reports all outliers by scanning the dataset at most twice (in some cases, even once), and significantly outperforms the existing solutions by a factor up to an order of magnitude. Due to the identification of outlier in each object, it needs long time consuming.

Mining Top-n Local Outliers in Large Databases

Outlier detection is an important task in data mining with numerous applications, including credit card fraud detection, video surveillance, etc. A recent work on outlier detection has introduced a novel notion of local outlier in which the degree to which an object is outlying is dependent on the density of its local neighborhood, and each object can be assigned a **Local Outlier Factor** (LOF) which represents the likelihood of that object being an outlier. Although the concept of local outliers is a useful one, the computation of LOF values for every data objects requires a large number of &k-nearest neighbor's searches and can be computationally expensive. Since most objects are usually not outliers, it is useful to provide users with the option of finding only n most outstanding local outliers, i.e., the top-n data objects which are most likely to be local outliers according to their LOFs.

However, if the pruning is not done carefully, finding top-n outliers could result in the same amount of computation as finding LOF for all objects. In this paper, we propose a novel method to efficiently find the top-n local outliers in large databases. The concept of "micro-cluster" is introduced to compress the data. An efficient micro-cluster-based local outlier mining algorithm is designed based on this concept. As our algorithm can be adversely affected by the overlapping in the micro-clusters, we proposed a meaningful cut-plane solution for overlapping data. The formal analysis and experiments show that this method can achieve good performance in finding the most outstanding local outliers.The computation of LOF values for each object, it needs a large number of nearest neighbors.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

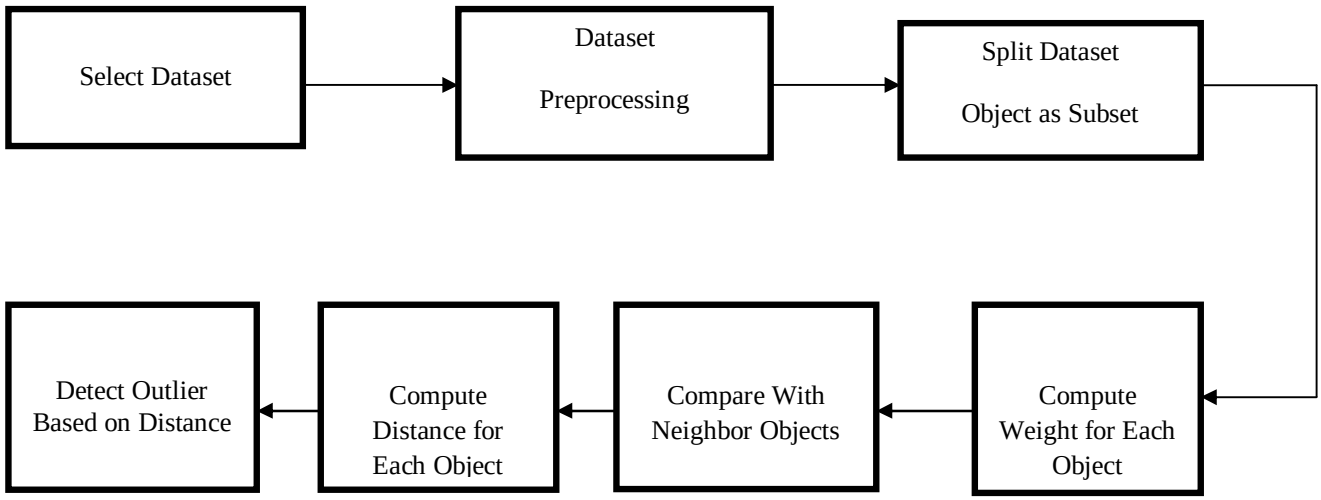
(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

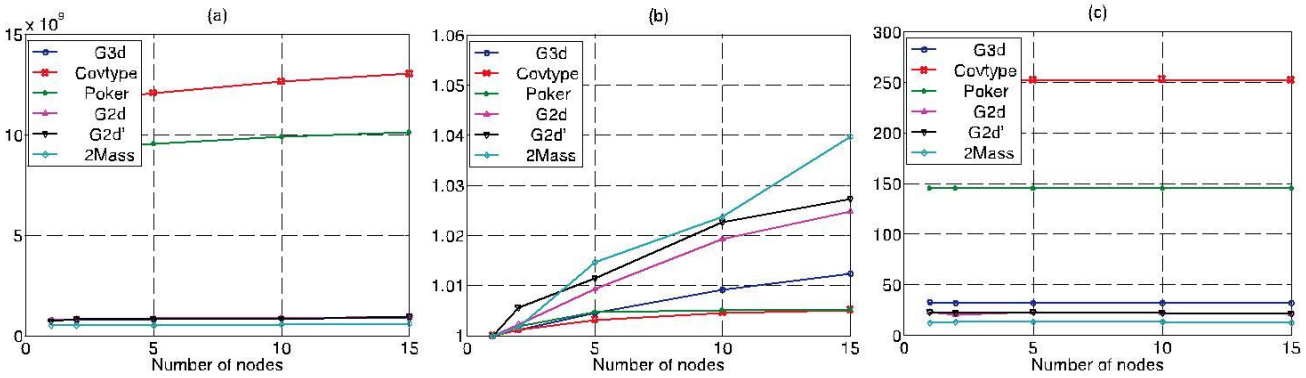
Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

III. ARCHIETECTURE





The main challenge in the data mining is Outlier detection. Outlier detection is the process of isolating the observations which are dissimilar from the data set. Distance based methods identifies an object as outlier. It detects based on the distances of its nearest neighbors. In the given data set of objects, an object can be associated with a weight or score. Function of k nearest neighbor's distances identifies the dissimilarity of the object from its neighbors. Top-n based outlier in a data set is an object. It's having a weight not smaller than the largest weight of the object. The weight of the data objects has been computed as from the sum of the distances from the nearest neighbor objects.

As for the parameter m, we have already seen that it is always inversely proportional to the number of iterations. Thus, by increasing m the number t of iterations is lowered. We recall that both the temporal cost and the amount of data to be transferred are proportional to tm. Thus, as long as tm remains constant, the parameter m has little impact on the performances of the algorithm. The following table reports t and the execution time for various values of m when n ¼ 10, k ¼ 50, and ' ¼ 15. It can be seen that, for Poker, the product tm is practically unchanged, and in fact the algorithm exhibited

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

in all of the three cases a similar execution time. Conversely, as for 2Mass, the value of the product tm is not constant and the relative variation of the execution times is more evident.

Dataset	t			Time [sec]		
	m			m		
	50	100	200	50	100	200
Poker	287	145	74	86.9	83.3	76.3
2Mass	17	12	9	7.7	7.7	10.6

Dataset \ ℓ	2	5	10	15
PSC	2.0	1.8	9.2	14.3
G2d	1.7	1.1	8.2	12.2
5G2d	1.8	1.5	8.8	12.8
10G2d	1.8	4.5	9.6	14.8

It is based on the algorithms such as distributed solving set algorithm, cost of the distributed solving set algorithm and lazy distributed solving set algorithms. It reduces data transformations and communication cost. From the comparison between the objects Cj, Ci, and C and D, we can easily estimate the outlier in the datasets. Outlier detection is fully based on the computation of distances between the objects. Final process is to display the outliers in the data set based on the distance of the data objects

It consists of a main cycle executed by a supervisor node, which iteratively schedules the following two tasks: 1) the core computation, which is simultaneously carried out by all the other nodes; and 2) the synchronization of the partial results returned by each node after completing its job. The computation is driven by the estimate of the outlier weight of each data point and of a global lower bound for the weight, below which points are guaranteed to be non outliers. The above estimates are iteratively refined by considering alternatively local and global information.

The core computation executed at each node consists in the following steps:

1. receiving the current solving set objects together with the current lower bound for the weight of the top nth outlier,
2. comparing them with the local objects,
3. extracting a new set of local candidate objects (the objects with the top weights, according to the current estimate) together with the list of local nearest neighbors with respect to the solving set and, finally,
4. determining the number of local active objects, that is the objects having weight not smaller than the current lower bound.

The comparison is performed in several distinct cycles, in order to avoid redundant computations. The above data are used in the synchronization step by the supervisor node to generate a new set of global candidates to be used in the following iteration, and for each of them the true list of distances from the nearest neighbors, to compute the new (increased) lower bound for the weight.

IV. COMPONENTS

DATASET PREPROCESSING

In the dataset processing module, collect the given dataset based on the process. A distributed strategy has been used. For that, data set is fragmented into many local data sets is called as subset of dataset. From these datasets we can

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

mining the outliers by using distributed solving set algorithm, solving set algorithm and lazy distributed solving set algorithm. In this process, extract the data from data set and insert it into the database.

SUBSET OF DATASET

After split the datasets into local datasets, solving set algorithm compares all dataset objects with a selected subset of the overall dataset. The subset of the dataset is called as candidate objects. It stores their k nearest neighbor with respect to the candidate sets. From these, we can easily obtain the true weights of each data object.

PROCESSING OF INPUT PARAMETERS

The distributed solving set algorithm process the input parameters. To get the inputs such as number of local nodes l, the size of the local dataset Di is represented as di.Then get the distance among the objects in the dataset D. For weight calculation, to find the number of neighbor objects and find the weight of each object.

ESTIMATE WEIGHT OF EACH OBJECT IN DATA SET

By using the solving set algorithm, we can easily estimate the weight of each dataset object.In this compare with the neighbor objects, if the weight of the object is lower than the greatest weight of the candidate objects. It is said to be non active. That is these objects cannot belong to the top-n outliers. The algorithm stops when Cj becomes empty and Cj is the union of candidate set objects.

ESTIMATE DISTANCE OF THE DATA OBJECT

In this module, to compute the distance between the two objects. Computing distances can be compared in three stages. In the first stage, compare each object in Cj with all other objects in Ci and updates the distance between them .Second stage is compare the objects in Ci with the objects of C and the objects of Ci compare with D. It can be estimated by using the cost of the distributed solving set algorithm. From the lazy distributed solving set algorithm, reduces the number of distances for each node. It starting from the smallest ones, it is sent by each local node to the supervisor node. Each supervisor node collects the additional distances.

OUTLIER DETECTION

Outlier detection for data mining is often based on distance measures, clustering and spatial methods. And only multivariate analysis is performed. From the comparison between the objects Cj, Ci, and C and D, we can easily estimate the outliers in the datasets. Outlier detection is fully based on the computation of distances between the objects. Final process is to display the outliers in the data set based on the distance of the data objects.

V. CONCLUSION AND FUTURE WORKS

To summarize a learned lesson, we started from an algorithm founded on a compressed form of data (the solving set) and derived a parallel/distributed data version by computing local distances and merging them at a coordinator site in an iterative way. The "lazy" version, which sends distances only when needed, showed the most promising performance. This schema could be useful also for the parallelized version of other kinds of algorithms, such as those based on Support Vector Machines. Additional improvements could be to find rules for an early stop of main iterations or to obtain a "one-

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 1, March 2014

Proceedings of International Conference On Global Innovations In Computing Technology (ICGICT'14)

Organized by

Department of CSE, JayShriram Group of Institutions, Tirupur, Tamilnadu, India on 6th & 7th March 2014

shot" merging method of the local information with some approximation guarantees.

REFERENCES

[1] F. Angiulli, S. Basta, S. Lodi, and C. Sartori, "A Distributed Approach to Detect Outliers in very Large Data Sets," Proc. 16th Int'l Euro-Par Conf. Parallel Processing (Euro-Par), pp. 329-340, 2010.

[2] F. Angiulli, S. Basta, and C. Pizzuti, "Distance-Based Detection and Prediction of Outliers," IEEE Trans. Knowledge and Data Eng., vol. 18, no. 2, pp. 145-160, Feb. 2006.

[3] F. Angiulli and F. Fassetti, "Dolphin: An Efficient Algorithm for Mining Distance-Based Outliers in very Large Datasets," Trans. Knowledge Discovery from Data, vol. 3, no. 1, article 4, 2009.

[4] F. Angiulli and C. Pizzuti, "Outlier Mining in Large High-Dimensional Data Sets," IEEE Trans. Knowledge and Data Eng., vol. 2, no. 17, pp. 203-215, Feb. 2005.

[5] A. Asuncion and D. Newman, UCI Machine Learning Repository,
2007.

[6] S.D. Bay and M. Schwabacher, "Mining Distance-Based Outliers in Near Linear Time with Randomization and a Simple Pruning Rule," Proc. Ninth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), 2003.

[7] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey," ACM Computing Survey, vol. 41, no. 3, pp. 15:1-15:58, 2009.

[8] H. Dutta, C. Giannella, K.D. Borne, and H. Kargupta, "Distributed Top-K Outlier Detection from Astronomy Catalogs Using the DEMAC System," Proc. SIAM Int'l Conf. Data Mining (SDM), 2007.

[9] A. Ghoting, S. Parthasarathy, and M.E. Otey, "Fast Mining of Distance-Based Outliers in High-Dimensional Datasets," Data Mining Knowledge Discovery, vol. 16, no. 3, pp. 349-364, 2008.

[10] S.E. Guttormsson, R.J. Marks, M.A. El-Sharkawi, and I. Kerszenbaum, "Elliptical Novelty Grouping for on-line Short-Turn Detection of Excited Running Rotors," Trans. Energy Conversion, vol. 14, no. 1, pp. 16-22, 1999.