Noise Removal Techniques using Data Analysis in Data Mining

Byalalli Rajeshri
Assistant Professor
Information Science and Engineering Department,
Guru Nanak Dev Engineering Collage, Bidar, India

Abstract **- Data mining is the process of extraction of relevant information from data warehouse. It also refers to the analysis of the data using pattern matching techniques. Presently, a very large amount of data stored in databases. This requires a need for new techniques and tools to aid humans in automatically and intelligently analyzing large data sets to acquire useful information. Removing objects that are noise is an important goal of data cleaning. Because data sets can contain large amount of noise, these techniques also need to be able to discard a potentially large fraction of the data. This paper presents, a different data cleaning methods to focus on removing noise includes in Data mining. Thus, if the goal is to enhance the data analysis as much as possible, these objects should be considered as noise, at least with respect to the underlying analysis.**

IndexTerms **– Data Mining, Data Analysis, Noise Removal.**

I. INTRODUCTION

Data mining is the process of extraction of relevant information from data warehouse. It also refers to the analysis of the data using pattern matching techniques. Presently, a very large amount of data stored in databases. This requires a need for new techniques and tools to aid humans in automatically and intelligently analyzing large data sets to acquire useful information. Removing objects that are noise is an important goal of data cleaning as noise hinders most types of data analysis. Most existing data cleaning methods focus on removing noise to enhance the data analysis. Thus, if the goal is to enhance the data analysis as much as possible, these objects should also be considered as noise, at least with respect to the underlying analysis.

Noise is "irrelevant or meaningless data". For most existing data cleaning methods, the focus is on the detection and removal of noise (low-level data errors) that is the result of an imperfect data collection process. The need to address this type of noise is clear as it is detrimental to almost any kind of data analysis. However, ordinary data objects that are irrelevant or only weakly relevant to a particular data analysis can also significantly hinder the data analysis, and thus these objects should also be considered as noise. If the goal is to use clustering to find the strong topics in a set of documents, then the analysis will suffer unless irrelevant and weakly relevant documents can be eliminated. Consequently, there is a need for data cleaning techniques that remove two types of noise. In some cases the amount of noise in a data set is relatively small. For example, it has been claimed that field error rates for business are typically around 5% or less if an organization specifically takes measures to avoid data errors. However, in other cases, the amount of noise can be large. For example, a significant number of false-positive protein interactions are present in current experimental data for protein complexes. To remove the noise through data analysis, their presented a several algorithms using different approaches.

The approaches like classification, estimation, prediction, association rules, clustering and description are used in data mining. The classification task is described by well-defined classes and training sets. The Brodley et al. [1] presented an algorithm based on classification. Estimation deals with a continuously valued outcome. Prediction can be thought of as a classification or estimation. Clustering is the task of segmenting a diverse group into a number of similar subgroups or clusters. Algorithm defined with [2], [3], [4],[6],[7] are cluster based algorithms. Based on all these methods several algorithms have been implemented as follows.

II. TECHNIQUES FOR DATA ANALYSIS AND NOISE REMOVAL

A Density-Based Algorithm for Discovering Clusters

M. Ester, et al. [2] presented the new clustering algorithm DBSCAN relying on a density-based notion of clusters which is designed to discover clusters of arbitrary shape. The key idea in DBSCAN is that for each point of a cluster, the neighborhood of a given radius has to contain at least a minimum number of points, i.e. the density in the neighborhood has to exceed some threshold. DBSCAN can separate the noise (out- liers) and discover clusters of arbitrary shape. It uses R*-tree to achieve better performance. But the average run time complexity of DBSCAN is *O (NlogN).* They performed an experimental evaluation of the effectiveness and efficiency of DBSCAN using synthetic data and real data of the SEQUOIA 2000 benchmark. The results of this experiments demonstrate that (1) DBSCAN is significantly more effective in discovering clusters of arbitrary shape than the well-known algorithm CLARANS, and that (2) DBSCAN outperforms LARANS by a factor of more than 100 in terms of efficiency.

STING: STatistical Information Grid-based method

Wang et al proposed a STatistical Information Grid-based method (STING) for spatial data mining [10]. They divide the spatial area into rectangular cells using a hierarchical structure. They store the statistical parameters (such as mean, variance, minimum, maximum, and type of distribution) of each numerical feature of the objects within cells. STING goes through the data set once to compute the statistical parameters of the cells, hence the time complexity of generating clusters is *O(N).* In STING, the hierarchical representation of grid cells is used to process. After generating the hierarchical structure, the response time for a query would be *O (K),* where K is the number of grid cells at the lowest level. Usually $K << N$, which makes this method fast. However, in their hierarchy, they do not consider the spatial relationship between the children and their neighboring cells to construct the parent cell. This might be the reason for the isothetic shape of resulting clusters, that is, all the cluster boundaries are either horizontal or vertical, and no diagonal boundary is detected. It lowers the quality and accuracy of clusters, despite the fast processing time of this approach.

WaveCluster: A Multi-Resolution Clustering Approach for Very Large Spatial Databases

WaveCluster [3] is a novel clustering approach based on wavelet transforms, which satisfies all the below requirements, using multi resolution property of wavelet transforms,

- Management of spatial data
- Clustering large spatial databases
- detecting clusters of arbitrary shape

We can effectively identify arbitrary shape clusters at different degrees of accuracy. And computational complexity of generating clusters of WaveCluster method is *O (N).* The results are not affected by outliers and the method is not sensitive to the order of number of input to be processed. Wavecluster is capable of finding arbitrary shape clusters with complex structures such as concave or nested clusters at different scopes and does not assume any specific shape for the clusters.

BIRCH: An efficient data clustering method for very large databases

Finding useful patterns in large datasets has attracted considerable interest recently, and one of the most widely studied problems in this area is the identification of *clusters,* or densely populated regions, in a multi-dimensional dataset. Prior work does not adequately address the problem of large datasets and minimization of I/O costs. BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) [4] uses a hierarchical data structure called CF-tree for incrementally and dynamically clustering the incoming data points. CF-tree is a height balanced tree which stores the clustering features. BIRCH tries to produce the best clusters with the available resources. They consider that the amount of available memory is limited (typically much smaller than the data set size) and want to minimize the time required for I/O. In BIRCH, a single scan of the dataset yields a good clustering, and one or more additional passes can (optionally) be used to improve the quality further. So, the computational complexity of BIRCH is *O (N).* BIRCH is also the first clustering algorithm to handle noise. Since each node in CF-tree can only hold a limited number of entries due to its size, it does not always correspond to a natural cluster [4]. Moreover, for different orders of the same input data, it may generate different clusters, in other words, it is order-sensitive. The performance comparisons of *BIRCH* versus *CLARANS,* a clustering method, show that *BIRCH* is consistently superior

Fast Outlier Detection in High Dimensional Spaces

F. Angiulli and C. Pizzuti [5] proposed a new definition of distance-based outlier that considers for each point the sum of the distances from its *k*nearest neighbors, called weight. Outliers are those points having the largest values of weight. In order to compute these weights, we find the *k* nearest neighbors of each point in a fast and efficient way by linearizing the search space through the Hilbert space filling curve. The algorithm consists of two phases; the first provides an approximated solution, within a small factor, after executing at most $d+1$ scans of the data set with a low time complexity cost, where d is the number of dimensions of the data set. During each scan the number of point's candidate to belong to the solution set is sensibly reduced. The second phase returns the exact solution by doing a single scan which examines further a little fraction of the data set. Experimental results show that the algorithm always finds the exact solution during the first phase after d & $d+1$ steps and it scales linearly both in the dimensionality and the size of the data set.

Mining Distance Based Outliers in Near Linear Time with Randomization and Simple Pruning Rule

Defining outliers by their distance to neighboring examples is a popular approach to finding unusual examples in a data set. Recently, much work has been conducted with the goal of finding fast algorithms for this task. Stephen D. Bay et al. [6] show a simple nested loop algorithm that in the worst case is quadratic, can give near linear time performance when the data is in random order and a simple pruning rule is used. They test the algorithm on real high-dimensional data sets with millions of examples and show that the near linear scaling holds over several orders of magnitude. The average case analysis suggests that much of the efficiency is because the time to process non-outliers, which are the majority of examples, does not depend on the size of the data set. .

LOF: Identifying Density-Based Local Outliers

For many KDD applications, such as detecting criminal activities in E-commerce, finding the rare instances or the outliers, can be more interesting than finding the common patterns. Existing work in outlier detection regards being an outlier as a binary property. The papers [7] contend that for many scenarios, it is more meaningful to assign to each object a *degree* of being an outlier. This degree is called the *local outlier factor* (LOF) of an object. It is *local* in that the degree depends on how isolated the object is with respect to the surrounding neighborhood. We give a detailed formal analysis showing that LOF enjoys many

desirable properties. Using real world datasets, we demonstrate that LOF can be used to find outliers which appear to be meaningful, but can otherwise not be identified with existing approaches. Finally, a careful performance evaluation of our algorithm confirms we show that our approach of finding local outliers can be practical.

HCleaner: hyperclique-based data cleaner

HCleaner, is a hyper clique-based data cleaner. It is based on the concept of hyperclique patterns which consist of objects that are strongly similar to each other. In particular, every pair of objects within a hyperclique pattern [8] is guaranteed to have a cosine similarity above a certain level. The cosine similarity measure is also known as the uncentered Pearson's correlation coefficient, a measure of association that describes the strength or magnitude of a relationship between two objects. HCleaner filters out all objects that do not appear in any hyperclique pattern. Experimental results show that using HCleaner generally leads to better performance as compared to the outlier based data cleaning alternatives.

Identifying mislabeled training data:

The Brodley et al. [1] uses consensus filters and majority vote filters to identify and eliminate mislabeled training samples. Their results show that if that the training data set is sufficiently large, then classification accuracy can be improved as more and more suspiciously labeled objects are removed. Cluster analysis provides an example of data cleaning for the elimination of weakly relevant or irrelevant objects.

A Rule Management System for Knowledge Based Data Cleaning

Louardi_Bradji, Mahmoud_Boufaida [9] proposed a rule management system for data cleaning that is based on knowledge. This system combines features of both rule based systems and rule based data cleaning frameworks. The important advantages of this system are threefold. First, it aims at proposing a strong and unified rule form based on first order structure that permits the representation and management of all the types of rules and their quality via some characteristics. Second, it leads to increase the quality of rules which conditions the quality of data cleaning. Third, it uses an appropriate knowledge acquisition process, which is the weakest task in the current rule and knowledge based systems. As several research works have shown that data cleaning is rather driven by domain knowledge than by data, they have identified and analyzed the properties that distinguish knowledge and rules from data for better determining the most components of the proposed system. The autonomy, extensibility and platform-independency of the proposed rule management system facilitate its incorporation in any system that is interested in data quality management.

III. CONCLUSION

Presently, a very large amount of data stored in databases. This requires a need for new techniques and tools for analyzing large data sets to acquire useful information. Removing objects that are noise is an important goal of data cleaning. This paper overviewed different data cleaning methods focus on removing noise. Thus, if the goal is to enhance the data analysis as much as possible, these objects should be considered as noise.

REFERENCES

[1] Carla E. Brodley and Mark A. Friedl. Identifying mislabeled training data. *Journal of Artificial Intelligence Research*, 11:131–167,1999

[2] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discoverying clusters in large spatial databases with noise.In *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining*, 1996.

[3] G. Sheikholeslami, S. Chatterjee, and A. Zhang. Wavecluster: A multi-resolution clustering approach for very large spatial databases. In *Proceedings of International Conference on Very Large Databases*, 1998.

[4] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch: an efficient data clustering method for very large databases. In *Proceedings of the 1996 ACM SIGMOD international conference on Management of data*, pages 103–114. ACM Press, 1996.

[5] F. Angiulli and C. Pizzuti. Fast outlier detection in high dimensional spaces. In *Proceedings of the Sixth European Conference on the Principles of Data Mining and Knowledge Discovery*, 2002.

[6] Stephen D. Bay and Mark Schwabacher. Mining distance-based outliers in near linear time with randomization and a simple pruning rule. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 29–38, New York, NY, USA, 2003. ACM Press.

[7] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jorg Sander. Lof:identifing density based local outliers. In *Proc. of the 200 ACM SIGMOD International Conference on management of Data*, 2000.

[8] Hui Xiong, Pang-Ning Tan, and Vipin Kumar. Mining strong affinity association patterns in data sets with skewed support distribution. In *Proceedings of the third IEEE International Conference on Data Mining*, pages 387–394, 2003.

[9] M. L. Lee, T. W. Ling, and W. L. Low. Intelliclean: A knowledge-based intelligent data cleaner. In *Proceedings of the sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2000.

[10] Wei Wang, Jiong Yang, and Richard Muntz. STING: A statistical information grid approach to spatial data mining. In Proceedings of the 23rd VLDB Conference, pages 186-195, Athens, Greece, 1997