International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol.2, Issue.4, July-Aug 2012 pp-1955-1957 ISSN: 2249-6645

Protection of Privacy in Distributed Databases using Clustering

Ganesh P.[1], KamalRaj R[2], Dr. Karthik S.[3]
[1,2,3]*Department of Computer Science EngineeringSNS College of Technology*

Abstract: Clustering is the technique which discovers groups over huge amount of data, based on similarities, regardless of their structure (multi-dimensional or two dimensional). We applied an algorithm (DSOM) to cluster distributed datasets, based on self-organizing maps (SOM) and extends this approach presenting a strategy for efficient cluster analysis in distributed databases using SOM and K-means. The proposed strategy applies SOM algorithm separately in each distributed dataset, relative to database horizontal partitions, to obtain a representative subset of each local dataset. In the sequence, these representative subsets are sent to a central site, which performs a fusion of the partial results and applies SOM and K-means algorithms to obtain a final result.

I. Introduction

In the recent years, there has been an increasing of data volume in organizations, due to many factors such as the automation of the data acquisition and reduced storage costs. For that reason, there has been also a growing interest in computational algorithms that can be used to extracting relevant information from recorded data.

Data mining is the process of applying various methods and techniques to databases, with the objective of extract information hidden in large amounts of data. A frequently used method is cluster analysis, which can be defined as the process of partition data into a certain number of clusters (or groups) of similar objects, where each group consists of similar objects amongst themselves (internal homogeneity) and different from the objects of the other groups (external heterogeneity), i.e., patterns in the same cluster should be similar to each other, while patterns in different clusters should not [1].

More formally, given a set of N input patterns: X = {x1, …, xN}, where each xj = (xj1, …, xjp) represents a p-dimensional vector and each measure xji represents a attribute (or variable) from dataset, a clustering process attempts to seek a K partition of X, denoted by C = {C1, …, CK}, (K <= N). Artificial neural networks are an important computational tool with strong inspiration neurobiological and widely used in the solution of complex problems, which cannot be handled with traditional algorithmic solutions [13]. Applications for RNA include pattern recognition, signal analysis and processing, analysis tasks, diagnosis and prognostic, data classification and clustering.

In some works, they presented a simple and efficient algorithm to cluster distributed datasets, based on multiples parallel SOM, denominated partSOM [5]. The algorithm is particularly interesting in situations where the data volume is very large or when data privacy and security policies forbid data consolidation into a single location.

This work extends this approach presenting a strategy for efficient cluster analysis in distributed databases using SOM and K-means. The strategy is to apply SOM algorithm separately in each distributed dataset, horizontal partitions of data, to obtain a representative subset of each local dataset. In the sequence these representative subsets are sent to a central site, which performs a fusion of the results and applies SOM and K-means algorithms to obtain the final result.

The remainder of the article is organized as follows: section 2 presents a brief review about distributed data clustering algorithms and section 3 describes the main aspects of the SOM. The proposed algorithm is presented in section 4 describes the methodology. Finally, section 5 presents conclusions and future research directions.

II. Bibliography Review

Cluster analysis algorithms groups data based on the similarities between patterns. The complexity of cluster analysis process increases with data cardinality and dimensionality. Cardinality :-(N, the number of objects in a database) and dimensionality:- (p, the number of attributes).Clustering methods range from those that are Largely heuristic method to statistic method. Several algorithms have been developed based on different strategies, including hierarchical clustering, vector quantization, graph theory, fuzzy logic, neural networks and others. A recent survey of cluster analysis algorithms is presented in Xu and Wunsch [1].

Searching clusters in high-dimensional databases is a non trivial task. Some common algorithms, such as traditional agglomerative hierarchical methods, are improper to large datasets. The increase in the number of attributes of each entrance does not just influence negatively in the time of processing of the algorithm, as well as it hinders the process of identification of the clusters. An alternative approach is divide database into partitions and to perform data clustering each one separately.

Some current applications have so large databases that are not possible to maintain them integrally in the main memory, even using robust machines. Kantardzic[2] points three approaches to solve that problem:

a) The data are stored in secondary memory and data subsets are clustered separately. A subsequent stage is needed to merge results;

b) Usage of an incremental grouping algorithm. Each element is individually stored in the main memory and associated to one of the existent groups or allocated in a new group;

c) Usage of a parallel implementation. Several algorithms work simultaneously on the stored data.

Two approaches are usually used to partition dataset: the first, and more usual, is to divide horizontally the database, creating homogeneous subsets of the data. Each algorithm operates on the same attributes. Another approach is to divide horizontally the database, creating heterogeneous

data subsets. In this case, each algorithm operates on the same registrations, but handle on different sets of attributes.

Some recent works about distributed data clustering include Forman and Zhang [3] that describes a technique to parallels several algorithms in order to obtain larger efficiency in data mining process of multiple distributed databases. Authors reinforce the concern need in relation to reducing communication

Several organizations maintain geographically distributed databases as a form of increasing the safety of their information. In that way, if safety policies fail, the invader has just access to a part of the existent information. Vaidya and Clifton [18] approaches vertically partitioned databases using a distributed K-means algorithm. Jagannathan et al. [11] present a variant of K-means algorithm to clustering horizontally partitioned databases. Oliveira and Zaïane [9] proposed a spatial data transformation method to protecting attributes values when sharing data for clustering, called RBT, that and is independent of any clustering algorithm.

In databases with a large number of attributes, another approach sometimes used is to accomplish the analysis considering only a subset of the attributes, instead of considering all of them. An obvious difficulty of this approach is to identify which attributes are more important in the process of clusters identification. Some papers related with this approach have frequently used statistical methods as Principal Components Analysis (PCA) and Factor Analysis to treat this problem. Kargupta et al. [10] presented a PCA-based technique denominated Collective Principal Component Analysis (CPCA) for cluster analysis of high-dimensional heterogeneous distributed databases. The authors demonstrated concern in reducing data transfers taxes among distributed sites.

Other works consider the possibility to partition attributes in subsets, but considering each one of them in data mining process. This is of particular interest for the maintenance of whole characteristics present in initial dataset. He et al. [12] analyzed the influence of data types in clustering process and presented a strategy that divided the attributes in two subsets, one with the numerical attributes and other with the categorical ones. Subsequently, they propose to cluster separately of each one of the subsets, using appropriate algorithms for each one of the types. The cluster results were combined in a new database, which was again submitted to a clustering algorithm for categorical data.

III. Self-Organizing Map

The self-organizing feature map (SOM) has been widely used as a tool for visualization of high-dimensional data. Important features include information compression while preserving topological and metric relationship of the primary data items [14]. SOM is composed of two layers of neurons, input and output layers. A neighbouring relation with neurons defines the topology of the map. Training is similar to neural competitive learning, but the best match unit (c or BMU) is updated as well as they neighbors. Each input is mapped to a BMU, which has weight vectors most similar to the presented data.

A number of methods for visualizing data relations in a trained SOM have been proposed [17], such as multiple views of component planes, mesh visualization using projections and 2D and 3D surface plots of distance matrices. The U-matrix method [17] enables visualization of the topological relations of the neurons in an organized SOM. A gradient image (2D) or a surface plot is generated by computing distances between adjacent neurons. High values in the U-matrix encode dissimilarities between neurons and correspond to cluster borders. Strategies for cluster detection using U-matrix were proposed by Costa and Netto [16]. The algorithms were developed for automatic partitioning and labeling of a trained SOM network. The result is a segmented SOM output with regions of neurons describing the data clusters.

IV. Proposed Methodology

Distributed clustering algorithms usually work in two stages. Initially, the data are analyzed locally, in each unit that is part of the distributed database. In a second stage, a central instance gathers partial results and combines them into an overall result.

This section presents a strategy for clustering similar objects located in distributed databases, using parallel self-organizing maps and K-means algorithm. The process is divided in three stages.

a) Traditional SOM algorithm is applied locally in each one of the distributed bases, in order to elect a representative subset from input data;

b) Traditional SOM is applied again, this time to the representatives of each one of the distributed bases that are unified in a central unit;

c) K-means algorithm is applied over trained self-organizing map, to create a definitive result.

The proposed algorithm, consisting of six steps: step 1 applies local clustering in each local dataset (horizontal parties from the database) using traditional SOM. Thus, the algorithm is applied to an attribute subset in each of the remote units, obtaining a reference vector from each data subset. This reference vector, known as the codebook, is the self-organizing map trained.

In step 2, a projection is made of the input data on the map in the previous stage, in each local unit. Each input is presented to the trained map and the index corresponding to the closest vector (BMU) present in the codebook is stored in an index vector. So, a data index is created based on representative objects instead of original objects. Despite the difference from the original dataset, representative objects in the index vector are very similar to the original data, since maintenance of data topology is an important characteristic of the SOM.

In step 3, each remote unit sends its index and reference vector to the central unit, which is responsible for unifying all partial results. An additional advantage of the proposed algorithm is that the amount of transferred data is considerately reduced, since index vectors have only one column (containing an integer value) and the codebook is usually mush less than the original data. So, reducing data transfer and communication overload are considered by the proposed algorithm.

Step 4 is responsible for receiving the index vector and the codebook from each local unit and combining partial results to remount a database based on received data. To remount each dataset, index vector indexes are substituted by the equivalent value in the

codebook. Datasets are combined juxtaposing partial datasets; however, it is important to ensure that objects are in the same order as that of the original datasets. Note that the new database is slightly different from the original data, but data topology is maintained.

In step 5, the SOM algorithm is again applied over the complete database obtained in step 4. The expectation is that the results obtained in that stage can be generalized as being equivalent to the clustering process of the entire database. The data obtained after the step 4 and that will serve as input in stage 5 correspond to values close to the original, because vectors correspondents in codebook are representatives of input dataset.

In step 6, K-means algorithm is applied over the final trained map, in order to improve the quality of the visualization results.

V. Conclusion

Self-organizing map is neural network concept, unsupervised learning strategy, has been widely used in clustering applications. However, SOM approach is normally applied to single and local datasets. In one of the research work, they introduced partSOM, an efficient strategy SOM-based to perform distributed data clustering on geographically distributed databases.

However, SOM and partSOM approaches have some limitations for presenting results. In this work we join partSOM strategy with an alternative approach for cluster detection using K-means algorithm.

References

[1] R. Xu and D. Wunsch II. "Survey of Clustering Algorithms", IEEE Trans. on Neural Networks, Vol.16, Iss.3, May 2005, pp. 645-678.

[2] M. Kantardzic, Data Mining: Concepts, Models, Methods, and Algorithms, IEEE Press, 2003.

[3] G. Forman and B. Zhang, "Distributed data clustering can be efficient and exact". SIGKDD Explor. Newsl. 2, Dec. 2000, pp 34-38.

[4] Chak-Man Lam, Xiao-Feng Zhang and W. K. Cheung, "Mining Local Data Sources for Learning Global Cluster Models", Web Intelligence, Proceedings IEEE/WIC/ACM International Conference on WI 2004, Iss. 20-24, Sept. 2004, pp. 748-751.

[5] F. L. Gorgônio and J. A. F. Costa, "Parallel Self-Organizing Maps with Applications in Clustering Distributed Data", International Joint Conference on Neural Networks (IJCNN'2008), (Accepted).

[6] A. Asuncion and D.J. Newman. UCI Machine Learning Repository,availableathttp://www.ics.uci.edu/~mlearn/ML Repository.html, Irvine, CA, 2007.

[7] J. C. Silva and M. Klusch, "Inference in distributed data clustering", Engineering Applications of Artificial Intelligence, vol. 19, 2006, pp. 363-369.

[8] P. Berkhin, "A Survey of Clustering Data Mining Techniques", In: J. Kogan et al., Grouping Multidimensional Data: Recent Advances in Clustering, New York: Springer- Verlag, 2006.

[9] S. R. M. Oliveira and O. R. Zaïane, "Privacy Preservation When Sharing Data For Clustering", In: Proceedings of the International Workshop on Secure Data Management in a Connected World, 2004.

[10] H. Kargupta, W. Huang, K. Sivakumar and E. Johnson, "Distributed Clustering Using Collective Principal Component Analysis", Knowledge and Information Systems Journal, Vol. 3, Num. 4, May 2001, pp. 422-448.

[11] G. Jagannathan, K. Pillaipakkamnatt and R. N. Wright,"A New Privacy-Preserving Distributed k-Clustering Algorithm", In: Proceedings of the 2006 SIAM International Conference on Data Mining (SDM), 2006.

[12] Z. He, X. Xu and S. Deng, "Clustering Mixed Numeric and Categorical Data: A Cluster Ensemble Approach", Technical Report, http://aps.arxiv.org/ftp/cs/papers/0509/0509011.pdf, 2005.

[13] S. Haykin, "Neural networks: A comprehensive foundation", 2nd ed., N. York: Macmillan College Publishing Company, 1999.

[14] T. Kohonen, Self-Organizing Maps, 3rd ed., New York: Springer-Verlag, 2001.

[15] J. Vesanto, "Using SOM in Data Mining", Licentiate's Thesis, Department of Computer Science and Engineering, Helsinki University of Technology, Espoo, Finland, 2000

[16] J. A. F. Costa and M. L. de Andrade Netto, "Clustering of complex shaped data sets via Kohonen maps and mathematical morphology". In: Proceedings of the SPIE, Data Mining and Knowledge Discovery. B. Dasarathy (Ed.), Vol. 4384, 2001, pp. 16-27.

[17] A. Ultsch, "Self-Organizing Neural Networks for Visualization and Classification", In: O. Opitz et al. (Eds). Information and Classification, Springer Berlin, 1993, pp.301-306.

[18] J. Vaidya and C. Clifton, "Privacy-preserving k-means clustering over vertically partitioned data". In: Proc. of the Ninth ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining, New York, 2003,pp.206-215.