International Journal of Contemporary Research in Computer Science and Technology (IJCRCST) e-ISSN: 2395-5325
Volume 4, Issue 5 (October '2018)

STUDY ON DISTRIBUTED DATA MINING TECHNIQUES AND METHODOLOGY

E.Prabakar Raj,
Assistant Professor,
Department of Computer Science,
Sengunthar Arts and Science College,
Tiruchengode,Tamilnadu,India.

R.Senthilkumar,
Assistant Professor,
Department of Computer Science,
Sengunthar Arts and Science College,
Tiruchengode,Tamilnadu,India.

Abstract: In recent years several approaches to knowledge discovery and data mining, and in particular to clustering, have been developed, but only a few of them are designed for distributed data sources. Distributed clustering model most closely related to statistics is based on distribution models. Clusters can then easily be defined as objects belonging most likely to the same distribution. A nice property of this approach is that this closely resembles the way artificial data sets are generated: by sampling random objects from a distribution. The aim of this paper is to explain Distributed Data Mining (DDM) started to gain attention during the late nineties. Al though it is still a young area of research, the body of literature on DDM constitutes a sizeable portion of the broader data mining literature. DDM in general deals with the problem of finding patterns in an environment where data is either naturally distributed, or could be artificially partitioned across computing nodes. It implies distribution of one or more of: users, data, hardware, or mining software. Centralized data mining systems do not address some the requirements of distributed environments, such as scalability and cooperation. Data mining in distributed environments is known as Distributed Data Mining (DDM), and sometimes as Distributed Knowledge Discovery (DKD).

Keywords: Data Mining, Distributed, Density models, Subspace models, Group models, k-means

I.INTRODUCTION

The notion of a "cluster" varies between algorithms and is one of the many decisions to take when choosing the appropriate algorithm for a particular problem. At first the terminology of a cluster seems obvious: a group of data objects. However, the clusters found by different algorithms vary significantly in their properties, and understanding these "cluster models" is key to understanding the differences between the various algorithms. Typical cluster models include: Connectivity models, Centroid models, Distribution models, Density models, Subspace models, Group models and Graph-based models [1-3] and [9]. A "clustering" is essentially a set of such clusters, usually containing all objects in the data set. Additionally, it may specify the relationship of the clusters to each other, for example a hierarchy of clusters embedded in each other. Data Clustering is one of the challenging mining techniques exploited in the knowledge discovery process. Clustering huge amounts of data is a difficult task since the goal is to find a suitable partition in a unsupervised way (i.e. without any prior knowledge) trying to maximize the similarity of objects belonging to the same cluster and minimizing the similarity among objects in different clusters. Many different clustering techniques have been defined in order to solve the problem from different perspective, i.e. partition based clustering, density based clustering, hierarchical methods and grid-based methods etc. In this paper we represent a survey of recent clustering approaches for data mining research.

1.1 Types of clustering algorithms

Clustering algorithms fall into a number of categories depending on their various aspects.

- Hard clustering, e.g. k-means, assigns each object exclusively to one cluster, thus creating a disjoint set of clusters. Probabilistic, e.g. expectation-maximization, and fuzzy clustering, e.g. fuzzy c-means, assigns for each object a degree of membership to each cluster, thus creating overlapping clusters.
- Hierarchical clustering, e.g. hierarchical agglomerative clustering, creates a dendo-gram of clusters such that clusters can contain sub-clusters. It works either bottom- up by merging clusters into larger clusters on the next level of the hierarchy, or top-down by splitting clusters into sub-clusters. Flat clustering, on the other hand, produces a flat set of clusters with no ordering or subsumption between them.
- Density-based clustering, e.g. DBSCAN [5], forms clusters by finding density-connected regions in the feature space.
- Neural network-based clustering, e.g. SOM [6], utilizes a neural network approach that automatically tunes the network weights such that similar objects tend to be close to each other.

II.DISTRIBUTED CLUSTERING

Han and Kamber [12] describe the need for parallel and distributed mining algorithms:

"The huge size of many databases, the wide distribution of data, and the com- putational complexity of some data mining methods are factors motivating the development of parallel and distributed data mining algorithms. Such algorithms divide the data into partitions, which are processed in parallel. The results from the partitions are then merged." With the continuous growth of data in distributed networks, it is becoming increasingly important to perform clustering of distributed data in-place, without the need to pool it first into a central location.

Table 2.1: Types of data and clustering process distribution

Centralized data	Distributed data
Centralized clustering	CD-CC DD-CC
Distributed clustering	CD-DC DD-DC

1) Centralized Data - Centralized Clustering (CD-CC) this is the standard approach where the clustering process and data both reside on the same machine. Other distributed models can be mapped to CD-CC by pooling the distributed data into one location and performing centralized clustering on it.
2) Distributed Data - Centralized Clustering (DD-CC) Data is dispersed across a number of nodes, while the clustering process runs on a single machine. Web mining is an example of DD-CC, where a single machine crawls and mines web pages from a large number of remote web sites.
3) Centralized Data - Distributed Clustering (CD-DC) Data is stored in one location, while clustering processes run on different machines accessing the same data. This is a typical case of parallel processing, such as in compute clusters and grid computing.
4) Distributed Data - Distributed Clustering (DD-DC) The highest level of distribution, where both the data and the clustering process are distributed.

In general, centralized clustering usually implies high computational complexity, while distributed clustering usually aims for speedup but suffers from communication overhead. The general goal of distributed clustering is achieving a level of speedup that outweighs communication overhead. Adopting distributed clustering introduces another factor affecting algorithm scalability: number of nodes. Data privacy is also an issue in distributed clustering, since the participating sites may not be willing to share their data with peers. In general, there are two architectures in distributed clustering: peer-to-peer and facilitator-worker.

- In the peer-to-peer model all nodes perform the same task and exchange the necessary information to perform their clustering goals.
- In the facilitator-work model one node is designated as a facilitator, and all others are considered worker nodes. The facilitator is responsible for partitioning the task among the workers and aggregating their partial results.

The goal of distributed clustering can be either to produce globally or locally optimized clusters. Globally optimized clusters reflect the grouping of data across all nodes, as if data from all nodes were pooled into a central location for centralized clustering. As a result, at the end all nodes acquire the same clustering solution, but local data stays the same. Globally optimized clustering is suitable for speeding up clustering of large data sets by partitioning the data among many nodes. Both the peer-to-peer [7, 8, 10] and the facilitator- worker [11, 12, 13] models can be used to achieve globally optimized clustering.

On the other hand, locally optimized clusters create a different set of clusters at each node, taking into consideration remote clustering information and data at other nodes. This implies exchange of data between nodes so that certain clusters appear only at specific nodes. Locally optimized clusters are useful when whole clusters are desired to be in one place rather than fragmented across many nodes. It is also only appropriate when data privacy is not a big concern. Both the peer-to-peer model [14] and the facilitator-worker [15] models can be used to achieve locally optimized clustering.

III.DISTRIBUTED DATA MINING

Distributed Data Mining (DDM) started to gain attention during the late nineties. Al-though it is still a young area of research, the body of literature on DDM constitutes a sizeable portion of the broader data mining literature. DDM in general deals with the problem of finding patterns in an environment where data is either naturally distributed, or could be artificially partitioned across computing nodes. It implies distribution of one or more of: users, data, hardware, or mining software [16]. Centralized data mining systems do not address some the requirements of distributed environments, such as scalability and cooperation.

Data mining in distributed environments is known as Distributed Data Mining (DDM), and sometimes as Distributed Knowledge Discovery (DKD). The central assumption in DDM is that data is distributed over a number of sites, and that it is desirable to derive, through data mining techniques, a global model that reflects the characteristics of the whole data set.

Applications of DDM are numerous, and are usually manifested as distributed comput- ing projects. They often try to solve problems in mathematics and science. Specific areas and example projects include: astronomy (SETI@home), biology (Folding@home, Predic- tor@home), climate change (CPDN), physics (LHC@home), cryptography (distributed.net), and biomedicine (grid.org). Those projects are usually built on top of a common platform providing low level services for distributed or grid computing. Examples of those platforms include: Berkeley Open Infrastructure for Network Computing (BOINC), Grid.org, World Community Grid, and Data Mining Grid.
A number of challenges (often conflicting) arise when developing DDM methods:

- Communication model and complexity
- Quality of global mode
- Privacy of local data

It is desirable to develop methods that have low communication complexity, especially in mobile applications such as sensor networks, where communication consumes battery power. Quality of the global model derived from the data should be either equal or com- parable to a model derived using a centralized method. Finally, in some situations when local data is sensitive and not easily shared, it is desirable to achieve a certain level of privacy of local data while deriving the global model. Although not yet proven, usually deriving high quality models requires sharing as much data as possible, thus incurring higher communication cost and sacrificing privacy at the same time.

3.1 HOMOGENEOUS VS. HETEROGENEOUS DISTRIBUTED DATA

We can differentiate between two types of data distribution. The first is homogeneous, where data is partitioned

horizontally among the sites; i.e. each site holds a subset of the data. The second is heterogeneous, where data is partitioned vertically; i.e. each site holds a subset of the attribute space, and the data is linked among sites via a common key.

3.2 EXACT VS. APPROXIMATE DDM ALGORITHMS

A DDM algorithm can be described as either exact or approximate. Exact algorithms produce a final model identical to a hypothetical model generated by a centralized process having access to the full dataset. Figure 3.2 illustrates the hypothetical process that is modeled by an exact distributed clustering algorithm. The exact algorithm works as if the data subsets, D_i, from each node were brought together into one data set, D, first;

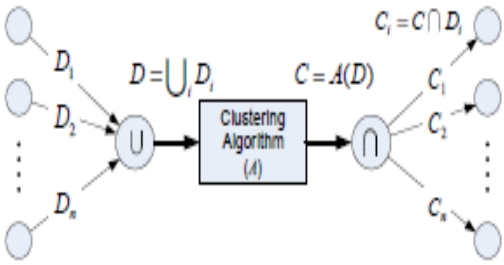

Figure 3.2: Exact Distributed Clustering Model

then a centralized clustering algorithm, A, had performed the clustering procedure on the whole data set. The clustering solutions are then distributed again by intersecting the data subsets with the global clustering solution. Approximate algorithms, on the other hand, produce a model that closely approximates a centrally-generated model. Most DDM research focuses on approximate algorithms as they tend to produce comparable results to exact algorithms with far less complexity [22].

3.3 COMMUNICATION MODELS

Communication between nodes in distributed clustering algorithms can be categorized into three classes (in increasing order of communication cost)

- Communicating models, which involves calculating local models that are then sent to peers or a central site. These models often are comprised of cluster centroids, e.g. P2P k-means [17], cluster dendograms, e.g. RACHET [18], or generative models, e.g. DMBC [20];
- Communicating representatives, in which nodes select a number of representative samples of the local data to be sent to a central site for global model generation, such as the case in the KDEC distributed clustering algorithm [19], and the DBDC algorithm [21]; and
- Communicating data, in which nodes exchange actual data objects; i.e. data objects can change sites to facilitate construction of clusters that exist in certain sites only, such as the case in the collaborative clustering scheme in [4], and the distributed signature-based clustering in [6].

3.4 DISTRIBUTED TEXT AND WEB MINING

Applications of DDM in the text mining area are rare, but usually employ a form of distributed information retrieval. Distributed text classification and clustering have received little attention. PADMA is an early example of parallel text classification [6]. The work presented by Eisenhardt et al [3] achieves document clustering using a distributed peer-to-peer network. They use the k-means clustering algorithm, modified to work in a distributed P2P fashion using a probe-and-echo mechanism. They report improvement in speed up compared to centralize clustering. Their algorithm is an exact algorithm, although it requires global synchronization at each iteration. A similar system can be found in [6], but the problem is posed from the information retrieval point of view. In this work, a subset of the document collection is centrally partitioned into clusters, for which "cluster signatures" are created. Each cluster is then assigned to a node, and later documents are classified to their respective clusters by com- paring their signature with all cluster signatures. Queries are handled in the same way, where they are directed from a root node to the node handling the cluster most similar to the query. Centralized mining can hardly scale to the magnitude of the data on the Web. Google, for example, is able to index the Web daily and respond to millions of queries per day because it employs a farm of distributed computing nodes that apply distributed algorithms for content indexing and query processing.

IV.CONCLUSION

Distribution-based clustering is a semantically strong method, as it not only provides you with clusters, but also produces complex models for the clusters that can also capture correlation and dependence of attributes. However, using this clustering method puts an extra burden on the user: to choose appropriate data models to optimize, and for many real data sets, there may be no mathematical model available the algorithm is able to optimize.

V.REFERENCES

[1]. K. Aas and L. Eikvil. Text categorisation: A survey. Technical Report 941, Norwegian Computing Center, June 1999.

[2]. H. Ahonen, O. Heinonen, M. Klemettinen, and A. I. Verkamo. Applying data min- ing techniques for descriptive phrase extraction in digital document collections. In Advances in Digital Libraries, pages 2–11, 1998.

[3]. H. Ahonen, O. Heinonen, M. Klemettinen, and A. I. Verkamo. Finding co-occurring text phrases by combining sequence and frequent set discovery. In R. Feldman, editor, 16th International Joint Conference on Articial Intelligence IJCAI-99 Workshop on Text Mining: Foundations, Techniques and Applications, pages 1–9, 1999.

[4]. M. W. Berry, S. T. Dumais, and G. W. O'Brien. Using linear algebra for intelligent information retrieval. SIAM Review, 37(4):573–595, 1995.

[5]. Souptik Datta, Chris Giannella, and Hillol Kargupta. K-means clustering over peer-to-peer networks. In Workshop on High Performance and Distributed Mining (HPDM05). SIAM International Conference on Data Mining (SDM05), 2005.

[6]. Souptik Datta, Chris Giannella, and Hillol Kargupta. K-means clustering over a large, dynamic network. In SIAM International

Conference on Data Mining (SDM06), pages 153–164, 2006.

[7]. Martin Ester, Hans-Peter Kriegel, Jrg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In 2nd In- ternational Conference on Knowledge Discovery and Data Mining (KDD-96), pages 226–231, Portland, Oregon, 1996. AAAI Press.

[8]. R. Feldman, M. Fresko, Y. Kinar, Y. Lindell, O. Liphstat, M. Rajman, Y. Schler, and O. Zamir. Text mining at the term level. In Principles of Data Mining and Knowledge Discovery, pages 65–73, 1998.

[9]. K. Hammouda and M. Kamel. Data mining in e-learning. In Samuel Pierre, editor, E-Learning Networked Environments and Architectures: A Knowledge Processing Perspective, pages 374–404. Springer, 2006.

[10]. K. Hammouda and M. Kamel. Distributed collaborative web document clustering using cluster keyphrase summaries. Information Fusion, Special Issue on Web Infor- mation Fusion, 2007. In Press.

[11]. Jiawei Han and Micheline Kamber. Data mining: concepts and techniques. Morgan Kaufmann, 2nd edition, 2006.

[12]. S. J. Hong and S. M.Weiss. Advances in predictive model generation for data mining. Technical Report RC-21570, IBM Research, 1999.

[13]. Eshref Januzaj, Hans-Peter Kriegel, and Martin Pfeifle. DBDC: Density based dis- tributed clustering. In EDBT, pages 88–105, 2004.

[14]. Erik L. Johnson and Hillol Kargupta. Collective, hierarchical clustering from dis- tributed, heterogeneous data. In Large-Scale Parallel Data Mining, Workshop on Large-Scale Parallel KDD Systems, SIGKDD, pages 221–244. Springer-Verlag, 2000.

[15]. Hillol Kargupta, Weiyun Huang, Krishnamoorthy Sivakumar, and Erik Johnson. Distributed clustering using collective principal component analysis. Knowledge and Information Systems, 3(4):422–448, 2001.

[16]. S. Kaski, T. Honkela, K. Lagus, and T. Kohonen. Creating an order in digital libraries with self-organizing maps. In Proceedings of WCNN'96, World Congress on Neural Networks, pages 814–817, September 1996.

[17]. J. Kay, N. Masionneuve, K. Yacef, and O. Zaiane. Determination of factors influenc- ing the achievement of the first-year university students using data mining methods. In Proceedings of the Workshop on Educational Data Mining at the 8th International Conference on Intelligent Tutoring Systems (ITS 2006), 45–52.

[18]. S. Krishnaswamy, S. W. Loke, and A. Zaslavsky. Cost models for heterogeneous distributed data mining. In Proceedings of the 12th International Conference on Software Engineering and Knowledge Engineering (SEKE), pages 31–38, Chicago, IL, 2000.

