Distributed Data Mining and Multi Agent-Based Distributed Data Clustering

Annan Naidu Paidi[1]
Assistant Professor, CSE Dept., Centurion University, Orissa, India
annanpaidi@gmail.com

Abstract: ***The increasing demands to scale up to massive data sets inherently distributed over a network with limited band width and computational resources available motivated the development of distributed data mining (DDM). Multi-agent systems offer architecture for distributed problem solving. Distributed data mining algorithms specialize on one class of such distributed problem solving tasks—analysis and modelling of distributed data. This paper offers a perspective on distributed data mining algorithms in the context of multi-agents systems. It particularly focuses on distributed clustering algorithms and their potential applications in multi-agent-based problem solving scenarios. A framework for multi-agent based clustering is described whereby individual agents represent individual clusters. A particular feature of the framework is that, after an initial cluster configuration has-been generated, the agents are able to negotiate with a view to improving on this initial clustering.***

Keywords: *agent-system, DDC problem, Multi agent for Distributed clustering*

1. Introduction

Originated from knowledge discovery from databases (KDD), also known as data Mining (DM), distributed data mining (DDM) mines data sources regardless of their physical locations. The need for such characteristic arises from the fact that data produced locally at each site may not often be transferred across the network due to the Excessive amount of data and privacy issues. Recently, DDM [2] has become a critical component of knowledge-based systems because its decentralized architecture reaches every networked business .Data Mining still poses many challenges to their search community. The main challenges in data mining are:

[1] Data mining to deal with huge amounts of data located at different sites the amount of data can easily exceed the terabyte limit.

[2] Data mining is very computationally intensive process involving very large data sets. Usually, it is necessary to partition and distribute the data for parallel processing to achieve acceptable time and space performance.

Input data change rapidly. In many application domain data to be mined either is produced with high rate or they actually come in streams. In those cases, knowledge has to be mined fast and efficiently in order to be usable and updated;
Security is a major concern in that companies or other organizations may be willing to release data mining results but not the source data itself.

2. Distributed Data mining

Data mining technology has emerged as a means for identifying patterns and trends from large quantities of data. Data mining and data warehousing go hand-in-hand: most tools operate on a principal of gathering all data into a central site, then running an algorithm against that data (Figure 1). There are a number of applications that are infeasible under such a methodology, leading to a need for distributed data mining [4].

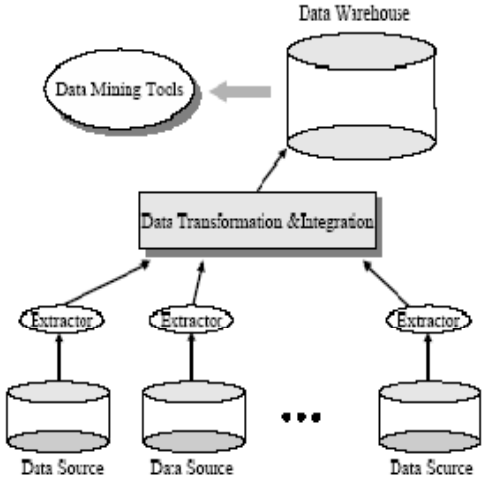

Figure 1: A datawarehouse Architecture

Distributed data mining (DDM) considers data mining in this broader context. As shown in figure (2), objective of DDM is to perform the data mining operations based on the type and availability of the distributed resources. It may choose to download the data sets to a single site and perform the data mining operations at a central location.

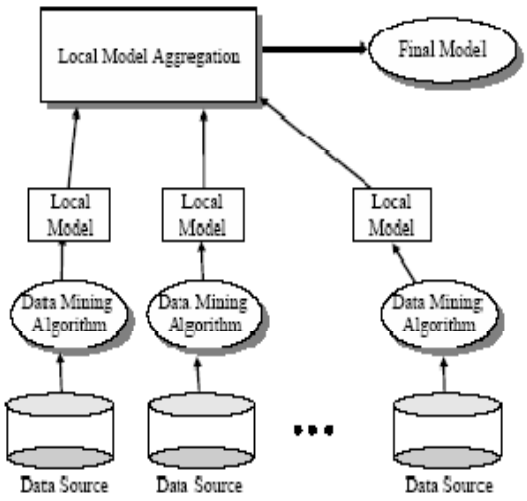

Figure 2: A Distributed Data Mining Framework

Data mining is a powerful new technology with great potential to help companies focus on the most important information in the data they have collected about the behaviour of their customers and potential customers. Data mining involves the use of sophisticated data analysis tools to discover previously unknown, valid patterns and relationships in large data set. These tools can include statistical models, mathematical algorithm and machine learning methods .It discovers information within the data that queries and reports can't effectively reveal.

3. Open Problems Strategy

Several systems have been developed for distributed data mining. These systems can be classified according to their strategy to three types; central learning, meta-learning, and hybrid learning.

3.1 Central learning strategy:

It is when all the data can be gathered at a central site and a single model can be build. The only requirement is to be able to move the data to a central location inorder to merge them and then apply sequential algorithms. This strategy is used when the geographically distributed data is small. The strategy is generally very expansive but also more accurate. The process of gathering data in general is not simply a merging step; it depends on the original distribution. For example, different records are placed in different sites, different attributes of the same records are distributed across different sites, or different tables can be placed at different sites, therefore when gathering data it is necessary to adopt the proper merging strategy. However, as pointed before this strategy in general is unfeasible .Agent technology is not very preferred in such strategy.

3.2 Meta-learning strategy:

It offers a way to mine classifiers from homogeneously distributed data. Meta-learning follows three main steps. The first is to generate base classifiers at each site using a classifier learning algorithms. The second step is to collect the base classifiers at central site, and produce meta-level data from a separate validation set and predictions generated by the base classifier on it. The third step is to generate the final classifier (meta-classifier) from meta-level data via a combiner or an arbiter. Copies of classifier agent will exist or deployed on nodes in the network being used. Perhaps the most mature systems of agent-based meat learning systems are: JAM system, and BODHI.

3.3 Hybrid learning strategy

It is a technique that combines local and centralized learning for model building is designed to support both learning strategies. In contrast to JAM and BODHI, Papyrus can not only move models from site to site, but can also move data when that strategy is desired. Papyruses a specialized system which is designed for clusters while JAM and BODHI are designed for data classification. The major criticism of such systems is that it is not always possible to obtain an exact final result, i.e. the global knowledge model obtained may be different from the one obtained by applying the one model approach (if possible) to the same data. Approximated results are not always a major concern, but it is important to be aware of that. Moreover, in these systems hardware resource definition for the most common agents that are used in MADM; the names might be different but they share the same functionalities in most cases.

4. Why Agents for DDM

The following arguments in favour or against the use of intelligent agents for distributed data mining [2].

Autonomy of data sources: A DM agent may be considered as a modular extension of a data management system to deliberatively handle the access to the underlying data source in accordance with given constraints on the required autonomy of the system, data and model. This is in full compliance with the paradigm of cooperative information systems.

Interactive DDM: Pro-actively assisting agents may drastically limit the amount a human user has to supervise and interfere with the running data mining process, e.g., DM agents may anticipate the individual limits of the potentially large search space and proper intermediate results.

Dynamic selection of sources and data gathering: In open multi-source environments DM agents may be applied to adaptively select data sources according to given criteria such as the expected amount, type and quality of data at the considered source, actual network and DM server load.

Scalability of DM to massive distributed data: A set of DM agents allow for a divide-and-conquer approach by performing mining tasks locally to each of the data sites. DM agents aggregate relevant pre-selected data to their originating server for further processing and may evaluate the best strategy between working remotely or migrating on data sources. Experiments in using mobile information

filtering agents in distributed data environments are encouraging [2].

Multi-strategy DDM: DM agents may learn in due course of their deliberative actions which combination of multiple data mining techniques to choose depending on the type of data retrieved from different sites and mining tasks to be pursued. The learning of multi-strategy selection of DM methods is similar to the adaptive selection of coordination strategies in a multi-agent system as proposed.

5. Distributed Data Clustering (DDC)

Data clustering [1] [9] is the task of partitioning a multivariate data set into groups maximizing intra-group similarity and inter-group dissimilarity. In a distributed environment, it is usually required that data objects are not transmitted between sites for efficiency and security reasons. An approach to clustering exploits the local maxima of a density estimate (d.e.) [5] to search for connected regions which are populated by similar data objects. a scheme for distributed clustering based on d.e. has been proposed, which we briefly recall. Every participating site computes data based on its local data only. Then, every site applies information theoretic regular multi-dimensional sampling to generate a finite, discrete, and approximate representation of the d.e., consisting of its values at a finite number of equidistantly spaced locations. The samples computed by all sites are transmitted and summed (by location) outside the originating site, e.g., at a distinguished helper site. The resulting list of samples, which is an approximate representation of the true global d.e., is transmitted to each participating site. Every site executes a density-based clustering algorithm to cluster its local data with respect to the global d.e., the values of which can be computed from the samples by means of a sampling series. Notice that a d.e. is not a band-limited function, therefore sampling produces aliasing errors, which increase as the number of samples decreases.

.

5.1 Multi-agent Based distributed clustering

To address these issues, a multi-agent system for distributed clustering of text documents was developed [2] [3] [9]. This approach does not depend on a global term frequency count, and is essentially a hybrid HAC and K-means clustering approach. To implement this multi-agent clustering system, several types of agents are used. At the lowest level, there are sub-cluster agents. Each sub-cluster agent represents a set of documents that are very similar. Above these, there are cluster agents. Each cluster agent represents a set of sub-cluster agents whose document sets have some similarities, but are not as similar as documents in a sub-cluster agent's document set. Above the cluster agent is the master cluster agent. These master cluster agents manage a set of cluster agents. There is not any implied relationship between the set of cluster agents (and their associated documents) managed by a master cluster agent. The master cluster agents are used to move cluster agents between other master cluster agents on different computer systems and therefore achieve better load balancing. Each computer in the distributed clustering system has only one master cluster agent. Finally, there are document multiplexer agents in the system. The

5.2The DDC Problem

We define the problem of homogeneous distributed data clustering for a clustering algorithm A as follows. Let S = {~xi |i = 1. . . N} _ Rn be a data set of objects. Let Lj, j =1, . . . ,M, be a finite set of sites. Each site Lj stores one dataset Dj , and it will be assumed that S =SMj=1Dj . The DDC problem is to find a site clustering Cj residing in the data space of Lj , for j = 1, . . . ,M, such that

i. Cj = {C\Dj : C 2 A(S)} (correctness requirement)

ii. Time and communications costs are minimized (efficiency requirement)

iii. At the end of the computation, the size of the subset of S which has been transferred out of the data space of any site Lj is minimized (privacy requirement).

The traditional solution to the homogeneous distributed data clustering problem is to simply collect all the distributed datasets Dj into one centralized repository where their union Sis computed, and the clustering C of the union S is computed and transmitted to the sites. Such approach, however, does not satisfy our problem's requirements both in terms of privacy and efficiency [7].

Therefore propose a different approach i.e multi agent based distributed data clustering.
document multiplexer agents accept new documents (and their representative vectors) and help to insert them into the clustering [8] [10].

Incorporating a new document into the distributed clustering system requires several steps. First, a document vector representing the document must be created. The document vector is then used to evaluate how the document compares to the documents that already exist in the clustering system. Finally, either the document is given to a software agent representing a set of document very similar to it, or if the document is unlike any currently in the system, a new software agent is created to represent it [6].

6. Conclusions

The distributed agent-based clustering worked surprisingly fast for large document sets. In addition, it allows for distributed processing of documents, and a hybrid k-mean and hierarchical clustering result. With this approach, it will be possible to cluster massive amounts of textual information in relatively short amounts of time, due to the scalability of the agent architecture. I plan to explore further the scalability of the agent architecture

presented in this paper.

References

[1] Joachim M. Buhmann, Data Clustering and Learning, TheMIT Press (c), 2002.

[2] Vuda Sreenivasarao, Multi Agent-Based Distributed Data Mining: An Over View, 2009-2010 IJRIC.

[3] Joel W. Reed, Thomas E. Potok and Robert M. Patton, A Multi-Agent System for Distributed Cluster Analysis.

[4] Matthias Klusch, Stefano Lodi, Gianluca Moro, Distributed Clustering Based on Sampling Local Density Estimates.

[5] Abdelhamid Bouchachia Distributed Data Clustering, Universit¨at Klagenfurt, Institute f¨ur Informatik-Systeme Universit¨atsstrasse 65, A-9020 Klagenfurt, Austria.

[6] Klusch, M., Lodi, S., & Moro, G. (20 Issues of agent-based distributed data mining. Proceedings of the International Joint Conferen Autonomous Agents & Multi ag Systems, AAMAS 2003, July 14-1 2003Melbourne,Victoria,Australia, (1034-1035). New York, NY: ACM.

[7] Elth Ogston , Benno Overwinter, Maarten van Steen, and Frances Brazier Method for Decentralized Clustering in Large Multi-Agent Systems Department of Computer Science, Vrije Universities Amsterdam.

[8] Santhana Chaimontree, Katie Atkinson and Frans Coenen, A Multi-Agent Based Approach To Clustering: Harnessing The Power of Agents, Department of Computer Science, University of Liverpool, Liverpool, L69 3BX, UK.

[9] E. Ogston, M. van Steen, F. Brazier and B. Overeinder, Data Clustering by Large-Scale. Adaptive Agent Systems, Vrije Universiteit Amsterdam Technical Report IR-CS-014.

[10] Md Faizan Farooqui, Md Faizan Farooqui, Dr. Md Rizwan Beg,A Comparative study of Multi Agent Based and High-Performance Privacy Preserving Data Mining, International Journal of Computer Applications (0975 – 8887)Volume 4– N o.12, August 2010.

Author Profile

Annan Naidu Paidi received B.Tech (IT) and M.Tech (CSE) Degrees from JNUTH and JNTUK, Andhra Pradesh. I have Six years experience in teaching field. At Present working as Assistant Professor of CSE in Centurion University, Odisha. I have published two research papers in international journal. My Research Interests are Data Mining, Computer Networks and Cryptography.