Distributed Data Mining for Multiple Sourced Heterogeneous Datasets: A Survey

Xing-ying LI, Shan-zi LI, Yi-xuan WU, Ai-jia HE, Xiao-ya HUANG and Xin ZHAO*
Beijing Normal University, Zhuhai, NO 18 Jinfeng Road, Tangjiawan, Zhuhai City, China
*Corresponding author

Keywords: Multiple sourced heterogeneous data sets, Distributed data mining, Distributed algorithms.

Abstract. In the information age of the 21st century, a large amount of information is collected and applied. However, due to the heterogeneity of system environment for data storage and computing, how to mine these distributed data sources has become a valuable research topic that attracted more and more attention. In this paper, we firstly presented the problem scenario and main challenges confronting with the problem of distributed data mining on multiple sourced heterogeneous data sets. Then, we surveyed research works related to the problem and elicited their main features on different technology domains to show current distributed solutions for different data mining algorithm categories. Finally, we reviewed in detail the research works and discussed the challenges remained in the distributed data mining problem for multiple sourced heterogeneous data sets.

Introduction

With the rapidly development and maturity of modern information technology like Internet, Cloud computing and Artificial intelligence, various of enterprise information systems is widely implemented in different enterprises, governments and application areas. While providing sufficient support for daily business operation or administrative activities of variable information systems, big business data is produced and accumulated by these information systems. These data which is various, huge scale and continuous updating offer enormous potential value to the strategic planning and management determination of enterprise or government constitute.

For example, as a large-scale industry, its different business sectors may located in different cities or even countries and rely on sundry information systems to support the daily routines of different departments. Therefore, for corporate headquarters to support their management decisions, it is necessary to collect data from different business departments and different systems to complete a data mining task. That the data sets used for the mining tasks are distributed geographically in different locations, based on different system or technology environments, and having different data storage architectures. For some official statistics tasks of administrative sectors, one of its data analyse task even need to rely on the data sets belonging to different enterprise or organizations (Such as transaction data for some e-commerce platforms). In this paper, we call this data sets which are distributed geographically and generated by different systems in a no-technology or storage environment or even belong to different enterprise or organizations MSHD (Multiple Sourced Heterogeneous Data sets).

Data Mining has been developed as an important research field for many years and is gradually maturing, but how to make full use of MSHD to realize efficient deep data analyse is still an open subject, facing many challenges. On one hand, to realize the data mining on MSHD, one possible idea is copy and integrate multiple data source data and implement centralized storage, so as to implement traditional data mining algorithms in a unified and consistent technology environment. However, this method requires a large amount of data transmission bandwidth and storage cost, and may faces data attribution problems. n the other hand, distributed data mining provides another possible idea for deep analysis and mining of MSHD. By distributing the computing progress required for data mining on multiple data source, most of the data processing is done locally and the
The Problem of Distributed Data Mining for Multiple Sourced Heterogeneous Data Sets

A Problem Scenario

With the development of all walks of life, the collection of data has become a problem. The systems of various companies are not only different in the time and process of construction and deployment, but also the affiliation companies, R&D companies and R&D technical methods of their respective systems are also far apart, which leads to the closed state of each system in the non-circulation, thus forming data cage. For example, for the NBS (National Bureau of Statistics), it’s required for them to make agriculture policies supporting by a large amount of agriculture products data, that’s why they need to collect statistics and economic indicators from various agriculture internet and enterprises, which makes a problem in the bandwidth and storage cost of data transmission. Besides, with the spring up of the e-commerce platform which becomes an important data source in recent years, it’s more difficult to collect these data, data attribution problems also follow.

The Issues Should be Considered

Mining Tasks and Algorithms. In this paper, we mainly consider association rules, decision trees, classification and clustering for distributed data mining algorithms. It is worth mentioning that the decision tree itself should be a classification algorithm, but because the decision tree is such an important and representative data mining algorithm, it even forms a relatively independent research direction and many algorithm variants. So in this paper we consider the decision tree as an independent algorithm.

The Distributed Computing Architectures. For distributed data mining, distributed computing technology plays a crucial role as the basis for mining. It not only provides design, implementation and runtime environment for distributed mining methods or algorithms, but also further determines the application scope of distributed data mining algorithms. According to our research results, the main distributed computing environments used in existing distributed data mining related research are: Fisher linear discriminant, Decision tree classifiers, Density-based, Grid services, Custom distribution framework, Web services, Hadoop frameworks.

Data Storage Environment. With the development of economy, science and technology, information has become an indispensable tool for us, so a large amount of data is collected and stored in different platforms and different environments. In the research process, we conducted statistical analysis on the data distribution environment and the types of data sets mentioned in the article. In order to get close to the reality, we have deeply studied the articles using data in distributed environments. After research, we found that the data sets used in the related articles are mainly distributed data sets in distributed parallel systems, horizontal and vertical partitioned databases and other distributed environments.

The Distributed Implementations of Mining Algorithms. First we compared the relevant indicators of each algorithm mentioned in the research work, and obtained Table 1.
Table 1. Relevant indicators of each algorithm.

| Defect                      | Algorithm                  |
|-----------------------------|----------------------------|
|                             | Association rules algorithm | Decision tree algorithm | Classification algorithm | Clustering algorithm |
| Too many database scans     | √                           |                          |                           |                     |
| Generate a large number of  | √                           | √                        |                          |                     |
| candidate sets              |                             |                          |                           |                     |
| Low efficiency              | √                           | √                        | √                          | √                   |
| Poor mining performance     | √                           | √                        | √                          | √                   |
| Large amount of communication | √                      | √                        | √                          | √                   |
| Low model accuracy          | √                           | √                        |                           | √                   |
| Sensitive to the amount of  | √                           |                          | √                          |                     |
| data                        |                             |                          |                           |                     |
| High time complexity        | √                           | √                        | √                          | √                   |

From this table, we can find that basically every distributed algorithm has low efficiency, poor mining performance and other defects. Then we will give a general overview of the main problems in the distributed implementation of different algorithms, the solutions and their technical features, and the unresolved challenges in the following part.

The Results of Survey

There are four main distributed data mining algorithms: association rule algorithm, decision tree, clustering algorithm and classification algorithm. We firstly understand the development trend of distributed data mining and the above four algorithms we have mentioned in the past ten years. We set the search keywords separately: "Distributed and data mining", "Distributed and association rules algorithm", "Distributed and decision-making tree", "Distributed and classification algorithm" and "Distributed and clustering algorithm", respectively in the IEEE database, ACM database and Science Direct database and other databases published in the past ten years of relevant literature search and descriptive analysis of the search results, we can see from the Figure 1 that distributed data mining and four algorithms are generally showing an upward trend in the past decade. It shows that more and more scholars have begun to pay attention to this aspect of research.

![Figure 1. The development of distributed algorithms in recent 10 years.](image)

Then, we summarize the distributed implementations of various algorithms (published literature research work) in Table 2: the distributed computing technology, data storage environment, technical characteristics of the proposed solution, and the unsolved problems.
### Table 2. Published works summary.

| Research Works | The Distributed Computing Architectures | Data Storage Environment | Issues Considered | Solutions Proposed | Challenges Remained |
|----------------|----------------------------------------|---------------------------|-------------------|--------------------|---------------------|
| **Association rule Algorithm** | DHP algorithm[1] | Transaction database, Distributed parallel system | How to improve the candidate project set generation process | Use Hash technology | High time cost when the database is too large |
|  | CD algorithm[2] | Not mentioned | How to improve the association rule mining algorithm of Apriori algorithm | Stored on each processor and swap to globally count candidate sets for each processor | 1. High communication costs 2. High control complexity. |
| **Decision Tree Algorithm** | Fisher's linear discriminant [3] | Not mentioned | Uniformly distributed data sets | Multi-feature segmentation | Develop a tree pruning algorithm |
|  | ID3 Entropy Calculation Method[4] | C4.5 | How to improve classification performance | | |
| **Clustering Algorithm** | Density-Based Distributed Clustering Algorithm [5][6] | Density-based | Not mentioned | Propose the density-based distributed clustering algorithm | Not mentioned |
|  | K-Dmeans algorithm[7] | Not mentioned | Improve privacy protection and efficiency | Proposed a new algorithm based on the K-Dmeans algorithm | |
|  | Distributed clustering algorithm based on K-means | Parallel computing environment, Mapreduce framework[8] | Not mentioned | Easy to fall into the problem of local optimal solution | Introducing Canopy algorithm to initialize clustering center and apply K-means algorithm on canopy | Not mentioned |
|  | Grid-based Distributed Clustering Algorithm | Web service, grid service[9] | Heterogeneous datasets | Handling complex data sets is difficult | Built a new framework ADMIRE | |
|  |  | Maximal frequent itemsets | High synchronization and communication overhead, Low resource utilization | Proposed a grid-based distributed Max-Miner | Not mentioned |
The Current Distributed Data Mining Algorithms

**Association Rule**

In view of the shortcomings of Apriori algorithm database scanning too many times and generating a large number of candidate sets, J.S. Park et al proposed a DHP algorithm based on hash optimization [1]. The algorithm mainly uses the algorithm of hash pruning technology to reduce the amount of data transmission, and obtain the support of each item set directly from the Hash table after building the Hash table in the candidate item set. Thereby reducing the amount of data in the large project set, and effectively reducing the amount of data transmission and the number of database scans, the performance bottleneck problem of the Apriori algorithm is greatly optimized.

For distributed data mining, Agrawal et al. proposed a parallel improved algorithm of the Apriori algorithm in a distributed environment—the CD algorithm [2]. The algorithm stores the global candidate itemsets and the frequent itemsets on each processor and exchanges them so that the candidate set of each processor is globally counted, thereby obtaining a global frequent itemsets. Although the CD algorithm has better performance, we found that it has high communication cost and high control complexity.

**Decision Tree**

In order to solve the problem that most existing distributed data methods for generating a single feature segmentation decision tree are inefficient, Jie Ouyang et al. proposed a method that uses multi-feature segmentation to generate a compact tree rather than a single feature segmentation decision tree generated by most existing distributed data methods [3]. Their method is based on the
Fisher linear discriminant function, which can process multiple classes in the data. Experimental results show that the consistency of the estimated scatter matrix significantly improves the classification performance. Although the communication cost of implementing heterogeneous environment algorithms is directly proportional to the total number of data instances, empirically, their approach can achieve centralized performance while saving more than half of communication costs.

Similarly, in order to solve the problem of low efficiency and accuracy of the decision tree classifier, Chenggang Li et al. extended different decision tree classifiers for processing uncertain data, using the sum of the number of meta-components based on probability theory to develop a tree pruning algorithm [4]. By pruning, they found that the accuracy of the classifier and the efficiency of building the decision tree were improved. However, in the case of uniform distribution, increasing the sampling density of the uncertain attribute value does not contribute much to the improvement of accuracy, but the calculation cost is higher.

Compared with the above two improved methods, the methods of JieOuyang et al. and Chenggang Li et al. can effectively improve the classification performance of decision trees. However, on the one hand, the increase in communication costs, and on the other hand, the increase in computational cost, both of them bring new problems while improving the classification tree performance. This shows that scholars are still working hard to solve the problem of classification performance of decision trees.

Clustering

Density-based Distributed Clustering Algorithm. Aiming at the shortcomings of distributed clustering algorithm DBDC with large traffic and low efficiency, Zheng Miaomiao et al. proposed a density-based distributed clustering algorithm DBDC* [5]. The algorithm combines the Bayesian information criterion BIC in the local screening of representative points, and obtains a small number of BIC core points that accurately reflect the local site data distribution, which effectively reduces the data traffic in the distributed clustering process, and comprehensively considers the distribution of data of each site in global clustering. The experimental results show that the efficiency of the algorithm DBDC* is better than that of DBDC, and the clustering effect is better.

Similarly, based on the existing distributed density clustering algorithm DBDC, Ni Wei-Wei et al. proposed a distributed clustering algorithm LDBDC based on local density and introduced the concept of local density clustering and density attractor, which can effectively solve the problem of noise-containing data and data distribution anomalies, as well as the incompatibility of high-dimensional data [6]. Theoretical analysis and experimental results show that the algorithm is superior to other existing algorithms in cluster quality and efficiency.

Distributed Clustering Algorithm Based on K-means. In order to solve the problem of the inefficiency of the existing distributed clustering algorithm, a new distributed clustering algorithm is proposed by Li Xiaowu et al. based on the K-\text{D}means algorithm [7]. The algorithm uses the density function value between data objects to optimize the initial cluster center of the site, thus greatly reducing the number of iterations of the cluster. At the same time, each slave station only needs to transmit its local clustering feature parameters to the primary site, which effectively reduces the traffic in the distributed clustering process and protects the independence of each site. The experimental results show that the algorithm is effective and feasible, and is also suitable for high-dimensional data, and is superior to K-Dmean in efficiency and clustering quality.

Due to the uncertainty of the initial center point, the k-means algorithm is easy to fall into the problem of local optimal solution. LI Xiao-yu et al. proposed a distributed improved algorithm based on Hadoop [8]. The algorithm initializes the clustering center of K-means algorithm by introducing Canopy algorithm, and combines MapReduce distributed computing model to give parallelized design method and strategy of improved algorithm. The experimental results show that the algorithm improves the accuracy and stability of the clustering algorithm, reduces the number of clustering. In addition, the algorithm also combines the actual text clustering application scenario, and the application of the algorithm in text clustering is given.
Grid-based Distributed Clustering Algorithm. In the article by Nhien-An Le-Khac et al., they mentioned that when we faced complex, data-rich, high-space heterogeneous data sets, Grid can be a good computing and storage platform, providing a very effective computing support for the deployment of distributed data mining [9]. So they set up a new framework ADMIRE based on grid for a large number of data sets. This framework not only utilizes the standard grid system (OGSA/WSRF), but also uses DGET to manipulate and store big data. The ADMIRE structure mainly achieves the convenience of big data mining in distributed data mining through three levels: interface layer, core layer and grid data layer. The experimental results show that the framework can efficiently improve the problems.

In the article by CONGNAN LUO et al., they made a more efficient data mining algorithm GridDMM (Grid-based Distributed Max-Miner) [10]. They dig the largest frequent itemsets from the database on the data grid by optimizing the synchronization and communication of the data network. Compared with similar algorithms, their algorithms require lower synchronization and communication overhead, which saves the resources needed for data mining.

In addition, for the performance degradation of execution time in cluster analysis of data mining, Giulia Bruna et al [11] proposed a new distributed system framework MicroClAn to evaluate and compare which quality indicators in clustering algorithms are most fully utilized for the performance degradation of execution time in cluster analysis of data mining. This framework uses a distributed environment to optimize execution time and evaluate several scheduling strategies. Moreover, they also utilized the most appropriate scheduler in a distributed environment to calculate their time to achieve their goals.

Classification

In the article by Jorge Calvo-Zaragoza et al., the Prototype Selection (PS) algorithm achieves a faster nearest neighbor classification by retaining only the best prototypes of the training set. In turn, these programs often reduce the accuracy of performance [12]. In this work, they proposed a new multi-label classification task strategy to solve this problem of reduced accuracy without using all training sets. To this end, the PS algorithm is used as a quick recommendation system that retrieves the most probable classes. Then, only the prototypes from the initial training set belonging to the suggested category are considered, and the actual classification is performed. The results show that this strategy provides a large number of compromise solutions that fill the gap between PS-based classification efficiency and traditional Knn accuracy. In addition, the solution not only achieves the performance of conventional Knn under the best conditions, but also surpasses the latter in the case of noise, proving that this is a more robust method.

In order to solve the difficulty of analyzing different types, large amount of data, and actually storing data sets in different geographical locations to find outliers, Zhen-Yu Chen et al [13] proposed a new algorithm that can identify outliers in the entire data without moving all the data to a single location. Their approach suggests focusing only on very small samples of different subsets of data from different locations. Through analysis and demonstration and experimental verification, the algorithm has high accuracy compared with complete concentration, and communication cost is only a small part.

For how to optimize the query in DDB, EnderSevinç et al proposed a new genetic algorithm (GA)-based query optimizer (New Genetic Algorithm (NGA)) and compared its performance with random and optimal (exhaustive) algorithms [14]. They are working on developing a system for designing efficient distributed databases (DDBs) with the allocation of data to distributed nodes. At the same time limit the run-time of NGA and to achieve additional improvements of GA-generated solutions we run a local optimization heuristic to obtain local optimal solutions. Although their NGA formulas do not achieve optimal results, improvements have been achieved compared to previous algorithms based on genetic algorithms. In the future, there is the possibility of developing a DDB query optimization algorithm with heterogeneous LAN/WAN links and bandwidth, and it can also be applied to the data warehouse field.
To train and deploy models on a distributed infrastructure, Elias De Coninck et al. proposed DIANNE, a modular framework for deep learning applications [15]. By splitting a neural network into its elementary operations and implementing these as services that can be linked on the fly, neural networks can be distributed across multiple, hetero-geneous devices. In addition, it supports a variety of styles of model and data parallelism by providing a variety of learning and evaluation procedures.

In order to solve the problem of parallelism of artificial neural network system, Hamidreza Mahini et al [16], provided an XML-based framework for implementing ANN on the Globus Toolkit platform for multi-purpose grid management software, using a grid to simulate a neural network to achieve high degree of parallelism in implementing ANN. Advantage of the model are its independency to the model and pattern of implemented Neural network, flexibility, scalability and using the well-known Globus Toolkit that helps hide the distribution and usage details of its services as its infrastructure.

Conclusion and Discussion

According to our research, the development of distributed mining algorithms is gradually increasing, and there may be better development in the next few years. From the application of distributed data mining algorithms, the distributed data mining algorithm has been widely used in various fields. This paper only selects several articles to elaborate. From this we can understand that each algorithm has defects such as low efficiency, poor mining performance and high time complexity, and scholars are constantly striving to improve these algorithms: Some of them have successfully improved the algorithm, and some have introduced new defects in the improvement of the old defects of the algorithm. This proves that there are many places worthy of research in distributed data mining, and in fact, distributed data mining under multi-source heterogeneous data sets does have a lot of exploration space and exploration value. With the deep research of various scholars, we believe that distributed data mining algorithms can be better developed and used in the future.

Acknowledgement

This paper is supported by Guangdong province Science & Technology Plan (project Number: 2016A040403029) and Guangong province Philosophy & Social Science "12th five" plan Interdisciplinary project (project number: GD15XGL02)

References

[1] J.S. Park, M.S. Chen, and P. S. Yu, An effective hash-based algorithm for mining association rules[J]. ACM SIGMOD Record, 1995, Vol. 24 (2), pp. 175-186.

[2] Agrwal R, Shafer J C. Parallel mining of association rules[J].IEEE Transactions on Knowledge and Data Engineering, 1996, Vol.8 (6), pp. 962-969.

[3] Jie Ouyang, Nilesh Patel, Ishwar Sethi. Induction of multiclass multifeature split decision trees from distributed data[J]. Pattern Recognition, 2009, 42(9).

[4] Chenggang Li, Liping Huang, Ling Tian. Efficient Building Algorithms of Decision Tree for Uniformly Distributed Uncertain Data[C].Natural Computation (ICNC), 2011 Seventh International Conference.

[5] Zheng MiaoMiao, Ji GenLin. An improved density based distributed clustering[J]. Journal of Nanjing University (Natural Sciences), 2008, (05), pp. 536-543.

[6] Ni WeiWei, Chen Geng, Wu YingJie, Sun ZhiHui. Local Density Distributed Clustering Algorithm[J].Journal of Software, 2008,(09), pp. 2339-2348.
[7] Li Xiaowu, Shao Jianfei, Liao Xiuling. A distributed clustering algorithm based on K-means algorithm[J]. Journal of Guilin University of Electronic Technology, 2011, (06), pp. 460-463.

[8] Li Xiao-yu, Yu Li-ying, Lei Hang, Tang Xue-fe. The Parallel Implementation and Application of an Improved K-means Algorithm[J]. Journal of University of Electronic Science and Technology of China, 2017, (01), pp. 61-68.

[9] Lekham N A, Kechadi M, Carthy J. Admire framework: Distributed data mining on data grid platforms[J]. 2015:67-72.

[10] Luo C, Pereira A L, Chung S M. Distributed Mining of Maximal Frequent Itemsets on a Data Grid System[M]. Kluwer Academic Publishers, 2006.

[11] Bruno G, Fiori A. MicroClAn: Microarray clustering analysis[J]. Journal of Parallel & Distributed Computing, 2013, 73(3):360-370.

[12] Jorge Calvo-Zaragoza n, Jose J. Valero-Mas, Juan R. Rico-Juan. Improving kNN multi-label classification in Prototype Selection scenarios using class proposals[J]. Pattern Recognition, 2015, Vol.48 (5), pp. 1608-1622.

[13] Das K, Bhaduri K, Votava P. Distributed anomaly detection using 1-class SVM for vertically partitioned data[J]. Statistical Analysis & Data Mining, 2011, 4(4):393-406.

[14] Ender, Ar A. An Evolutionary Genetic Algorithm for Optimization of Distributed Database Queries[M]. Oxford University Press, 2011.

[15] Coninck E D, Bohez S, Leroux S, et al. DIANNE: a modular framework for designing, training and deploying deep neural networks on heterogeneous distributed infrastructure[J]. Journal of Systems & Software, 2018, 141.

[16] Mahini H, Mahini A, Ghofrani J. XDANNG: XML based Distributed Artificial Neural Network with Globus Toolkit[J]. Computer Science, 2009.