2023 IEEE 6th International Conference on Information Systems and Computer Aided Education (ICISCAE) | 979-8-3503-1344-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICISCAE59047.2023.10392962

Research and Implementation of Parallel CART Algorithm Based on Distributed Database

Jie Wang
School of Artificial Intelligence
The Open University of Guangdong
Guangzhou, China
gdwangjie2008@163.com

Abstract—**This paper presents a parallel CART method for multivariate association query. The amount of communication information is reduced by performing two semi-joins. This method adopts a multi-node parallel method, which greatly speeds up the execution efficiency of the system. The sequence of operations generated by this algorithm has global optimization characteristics. One of its important development trends is to realize effective data query by optimizing the database. The simulation experiment proves that the method has fast processing speed, high node utilization rate and strong practicability. This method is suitable for large data processing.**

Keywords—Distributed database, parallel CART algorithm, query processing, optimization algorithm

I. Introduction

The distributed database system belongs to a kind of computer network technology, which is based on the centralized database technology, but it differs from the centralized database in that it can be distributed and stored in different locations of the network. Due to the different storage locations, its data processing capabilities will also have certain differences. Although the scattered and redundant data in the DB makes the query and optimization problems in the DB more complicated, distributed query and optimization also play an important role in the actual application of distributed databases [1]. There are two purposes of searching for the best in the DB. The first purpose is to reduce the search overhead. The second is to shorten the query response time as much as possible, which is especially important for distributed database systems. Since it has multiple computers, the dispersion and redundancy of data also enables queries to be performed in parallel. In this way, the response time of the query can be shortened and the processing efficiency of the query can be improved. 1) Decompose the global query into subqueries on the local database. 2) Compress the data of the local query to multiple locations, thereby reducing the cost of data transmission. 3) Send the intermediate results of each website to the inquiry website to obtain the final inquiry result. In 1), as long as a query request of a user is equivalent to a sub-question of a local database, a corresponding solution can be obtained. 3) You can refer to some algorithms in the centralized database. But 2) Because a large amount of data needs to be processed, the processing and optimization of its query is an important aspect to improve its query efficiency. As for query processing and optimization, an appropriate method is used to minimize the information required for communication so as to speed up the query and reduce the system cost. A commonly used solution is to use semi-connected technology to reduce its calculation load, thereby reducing communication overhead and improving system response speed. This project intends to use the parallel CART decision tree (SPC-DT) method in the Spark environment to improve its classification accuracy and learning efficiency. (1) Using the method of vertical data segmentation, each node only undertakes its own calculation work, thereby reducing the information exchange between nodes and effectively reducing the communication cost. (2) For adjacent attributes, the Fayyad algorithm is used to determine the classification of its edge points, thus improving the learning speed of the decision tree.

II. The process model of executing data query in a distributed environment

The database structure construction in a distributed environment is shown in Figure 1 (the picture is quoted in Distributed database environments). Part of the data model of a database in a decentralized environment, including a model of a three-tier model [2]. This model is known as a centralized database. In the distributed database, the global data model can help the local data model to realize the conversion to the whole model. The global data model consists of global external, global concept, fragmentation model, etc., which mainly refers to the user view in the global data model, and the global concept model is a related logical subset. A shard pattern is an image relationship. It's between related shards and global relationships. Although each fragment belongs to the global relationship, it can be decomposed into multiple fragments in the global context. Remote query is to realize remote communication of single-point data. To reduce the costs associated with communication when performing queries when performing a corresponding redundancy allocation for the associated data, the selected data should be from the associated node closest to the querying node. The so-called global query refers to the realization of multi-point data query [3]. After selecting the corresponding query object, the connection of the binary operation can be determined by using the access path and related algorithms, so as to determine the execution node, and the communication cost, query speed and execution efficiency must be integrated in the whole process consider. At the same time, it is also necessary to determine the execution site of the relevant operations in the query process.

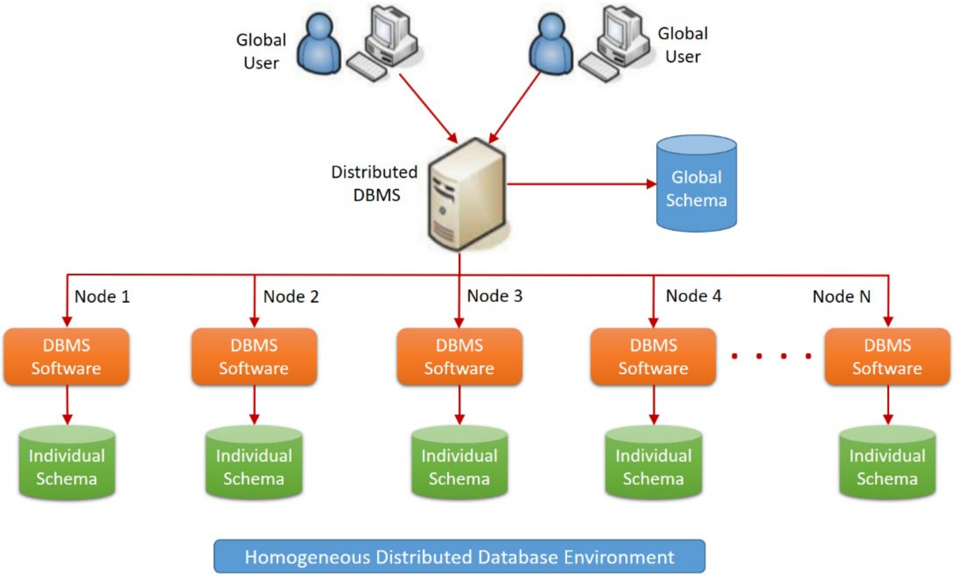
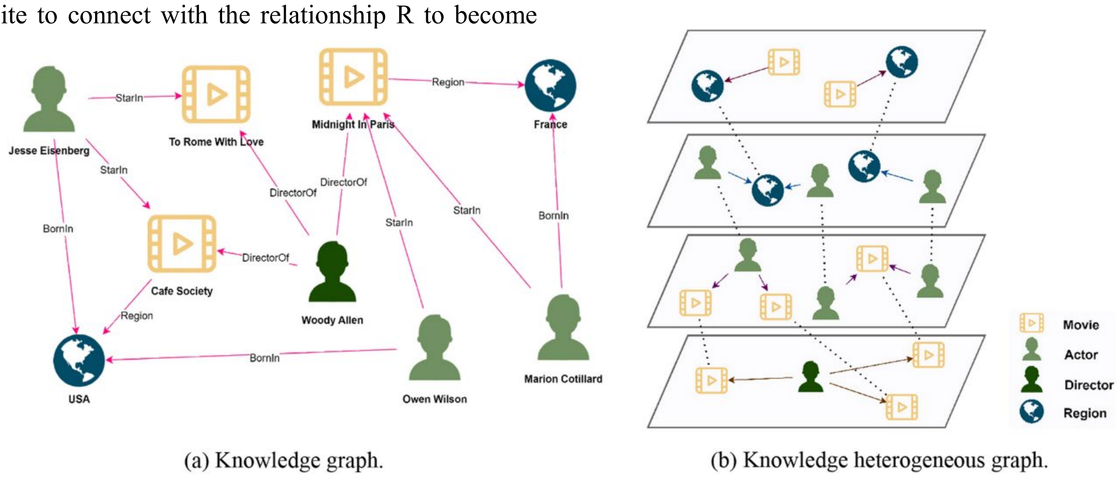


Fig. 1. The structure of the database in a distributed environment

The binary split reduction algorithm is a new method based on the semi-connected method, which aims to reduce the data transfer overhead in the system and make some operations in the system have good parallelism [4]. Let the connection attribute of any two relations R and S to be connected be X, and the relations R and S are in sites A and B respectively. The join was done after semi-join reduction. The attribute X is passed to the B site through the relationship R, and the relationship S is simplified into the relationship So through the semi-join, and then So is passed to the A site to connect with the relationship R to become Co. The method of bipartite reduction is to transfer the X attribute of So to A, and connect it to R, and complete the half of the two halves to obtain a complete half-connection, and then use the bipartite condition to connect Ro of station A to station B Separate So, transmit the data with the same conditions to the same place and connect them, so as to get the connected results scattered on the two places. The process is shown in Figure 2 (the picture is quoted in Iterative heterogeneous graph learning for knowledge graph-based recommendation).



(a) Knowledge graph.

(b) Knowledge heterogeneous graph.

Fig. 2. Connection after bisection

III. Algorithm Design

A. CART decision tree algorithm

A classification method of the Gini index is proposed, which adopts the classification method of the CART decision tree and determines the classification nodes according to the principle of least weight (Figure 3 cited in Energies 2022, 15(18), 6620). The decision tree generated by the CART method can be divided into two types according to its type, namely classification tree and regression tree. Ginni is a measured dataset

$$Gini(D)=1-\sum_{\beta=1}^{i} r_{\beta}^{2} \tag{1}$$

Among them, $i(1<i\leq\mathbb{N})$ is the number of different values in data set D, and r_{β} represents the probability of taking a category β among all categories. $Gini(D)$ represents the probability of randomly sampling 2 instances from dataset D whose class labels belong to different classes. The smaller the value of $Gini(D)$, the higher the

purity of the data set. The Gini index of the current attribute B is defined as:

$$Gini\_index(D,B)=\sum_{e=1}^{N}\frac{|D^e|}{|D|}Gini(D^e) \qquad (2)$$

$e(0<e\le\mathbb{N})$ represents the value category of the current attribute B, and D^e represents all instances in D that take the value of e on the current attribute B. When training the tree model, select the node where the attribute with the smallest Gini index after division is located as the split point.

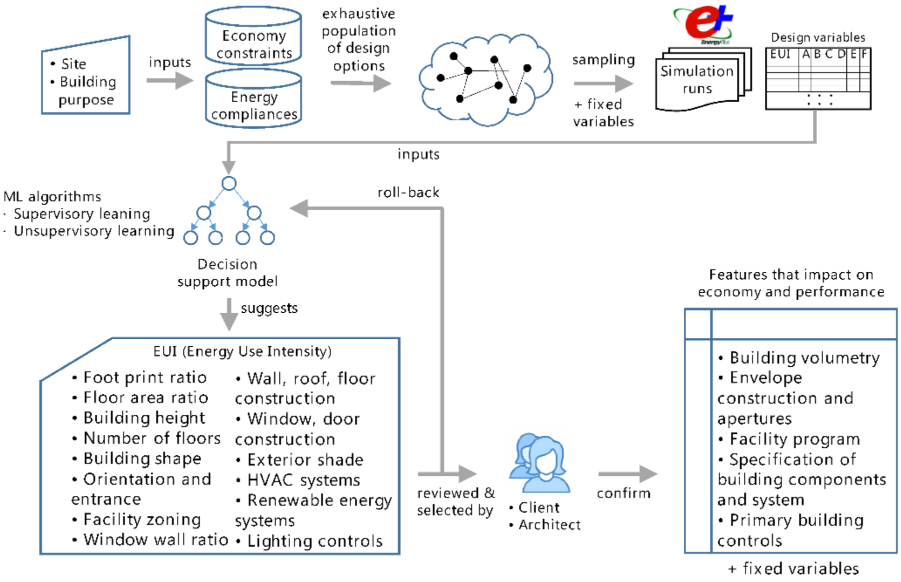


Fig. 3. CART decision tree algorithm

B. Spark Parallel Distributed Computing Architecture

Apache Spark is a distributed computing architecture based on AMPLab of Berkeley University. It is mainly used to process massive amounts of data. The Spark architecture is centered on SparkCore, with the RDD API as the basic program abstraction. Spark reuses the data stored in the space and the intermediate results obtained by calculation, thereby reducing the frequency of HDFS reading and writing data and improving the operating efficiency of the system. Spark has multiple components and interfaces in multiple languages, allowing users to meet their requirements at one time, thus bringing them a good experience. The Spark cluster is based on a master/slave architecture [5]. The master performs central coordination and schedules all slaves. The master is called a driver node, and the slave nodes are called executors. The two communicate with each other and work together. Figure 4 shows the parallel distributed computing architecture of Spark (the picture is referenced in A hybrid multi-objective whale optimization algorithm for analyzing microarray data based on Apache Spark).

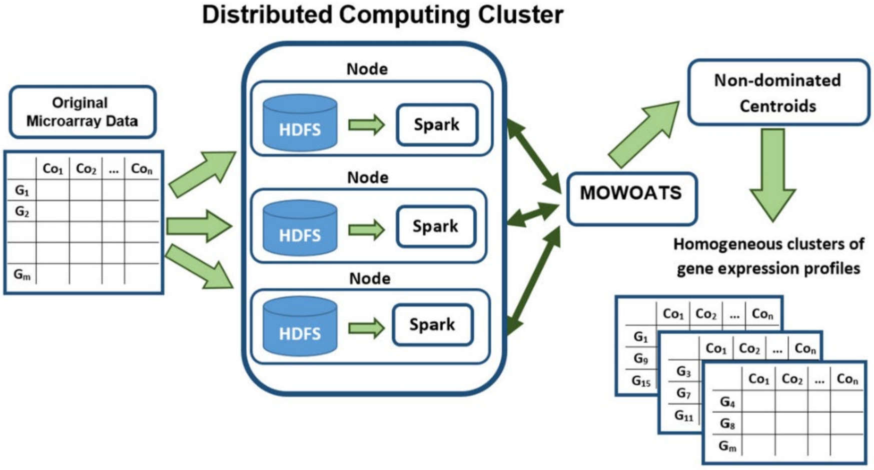


Fig. 4. Spark parallel distributed computing architecture

C. Fayyad Algorithm

In the process of generating the decision tree model, the continuous attribute is discretized by dividing the attribute range of the data set D into two sub-ranges. Suppose its sub-datasets are D_1 and $D_2, D_1 \in D, D_2 = D - D_1$ respectively. In order to evaluate its average class entropy, Fayyad defines the method of calculating the weighted average of its result class entropy to determine the boundary point category. It is defined as follows:

Definition 1 Boundary point: data set D, sort attribute value B in it in ascending order. D value C in attribute B is a boundary point if and only if in dataset D there exist 2 subsets D_1 and $D_2, D_1 \in D, D_2 = D - D_1$ such that $B(D_1) < W < B(D_2)$, and there does not exist any subset $s \in D$ such that $B(D_1) < B(s) < B(D_2)$. $B(s)$ represents the value of attribute B in dataset D.

$$Entropy(D_1, D_2, \cdots, D_n) = -\sum_{\beta=1}^{n} r_\beta \log r_\beta \qquad (3)$$

Where r_β represents the probability of obtaining class β in data set D.

Definition 3Fayyad boundary point category determination theorem: the information entropy obtained by W as the division point is defined as $Entropy(B, W; D)$:

$$Entropy(B, W; D) = \frac{|D_1|}{D} Entropy(D_1) + \frac{|D_2|}{D} Entropy(D_2) \qquad (4)$$

Fayyad proved that the split point W that minimizes the average class entropy is the bisection point of the current attribute B when building a decision tree.

Fayyad boundary point judgment condition: Assume that W is in a certain interval n_j of continuous attribute B that belongs to the same category F_k, and $n_j \geq 2, W_1$ and W_2 are two boundary points of n_j instances [6]. Assume S instances whose attribute value in attribute B is less than W_1, U instance whose value in attribute B is greater than W_2 , where $0 \leq S, U \leq N - n_j$ is. Suppose the category of the S instances on the left is $F_i, i = 1, \cdots, k$, and the number of each category is S_i, similar to the left, the category of the U instances on the right is also $F_i, i = 1, \cdots, k$, and the number of each category is U_i, where $0 \leq S_i \leq S, 0 \leq U_i \leq U, \sum S_i = S$ and $\sum U_i = U$. Assume there are n_c instances whose values in attribute B are between W_1 and W_2 . Fayyad proved that the average class information entropy achieves the minimum value when $n_c = 0$ or $n_c = n_j$.

IV. System detection

Define the corresponding relational schema R and turn it into a limited attribute set $\{A_1, A_2, \ldots, A_n\}$. For the instance R existing in the relational schema, it belongs to a corresponding mapping, and the mapping relationship is from the schema R to the data domain. The mapping relationship can be expressed as $\{t_1, t_2, \ldots, t_n\}$. If B is in the attribute of R, the specific representation is $t.B$. It can also be known that the following conditions must be satisfied $R.B = S.C$, and by defining the corresponding query statement q, the optimal connection strategy can be found [7]. For two pieces of data, although there are unequal numbers of connection conditions between the two, it is not necessary to consider the characteristics of the query graph. If you want to reduce the response time in distributed query, you can use spanning tree $QGTreeq$. For the spanning tree, its related costs are closely related to the size of the connection operation relationship, so it is necessary to find the sequence with the best conditional expression. The minimum cost of the single-condition spanning tree $QGTreeq$ is: $QGTreeq = \min\{Cost(cs)\} cs \in CS$. For the query conditional expression, if other forms of the query statement are used during the derivation process, it can also be directly regarded as a conditional expression with redundant characteristics. After implementing comprehensive analysis for $QGTreeq$ redundant expressions, the optimal conditional expression sequences can be obtained, which are $\{R.A, T.D\}, \{T.E, S.C\}, \{T.E, U.F\}$ respectively, and the corresponding total cost is 8k. But in the process of distributed query optimization, it is not enough to just consider $QGTreeq$, and it is necessary to use optimization algorithms to achieve this goal. The CHAIN algorithm can be used to obtain the lowest cost and the connection sequence of the corresponding relationship (Table I). The corresponding query optimization results are shown in Table II.

TABLE I. Example data of the CHAIN algorithm

i	**1**	**2**	**3**	**4**
Ri	30	10	10	20
Part(*Ri*)	U	C	D	*V*
ti	3	2	1	4
selectivity				
*σ*12	*σ*23	*σ*34		
0.2	0.1	0.2		

TABLE II. Related experimental results

i	*j*	*Cost*(*i,j*)	*cs*(*i,j*)	*Rij*	*tRij*
1	1	344	*cl*	63	5
2	3	31	*c*2	10	3
3	4	281	*c*2	42	5

Use the random number acquisition and use the random number detection library interface function to detect the obtained data, judge and output the detection results [8]. The most commonly used in these data is DIGITAL_CAPTURE_TEST (), which can directly call the random number detection library function. Perform tests and print test results. Back to the SETUP development environment, use the Random_Test test method as a random number detection class. Select the "Random Test" icon to conduct a "Random Data Collection" probe and output the test results. The results of the tests are shown in Figure 5.

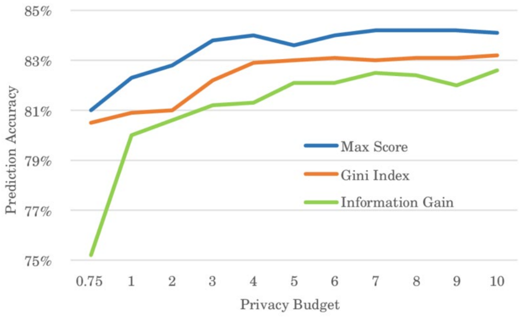


Fig. 5. Distributed database query efficiency curve based on CART decision tree algorithm

V. Conclusion

In the remote network, the connection sequence pairs generated by using the minimum spanning tree algorithm can minimize the total cost of the pre-estimation. In the local network, the improved minimum spanning tree algorithm can generate as many parallel connection sequence pairs as possible, and the algorithm Can be used repeatedly to generate multiple parallel linker pairs. This keeps the total cost as low as possible. A CART method based on the principle of edge point classification is proposed to reduce the time of edge point classification. A new method based on edge point classification is proposed. At the same time, the training time is shortened by vertically partitioning the data set to reduce network communication costs. Due to the query requirements on each destination website, the physical fragmentation of the database on each destination website can get a query strategy close to the best.

References

[1] Du Xiaofang, Chen Yihong, Wang Denghui, et al. Parallel CART decision tree algorithm on big data platform. Journal of China West Normal University: Natural Science Edition, vol.42, pp.69-71, February 2021.

[2] Sun Jinmang, Yu Zhongqing, Wang Haiya. Online Fault Diagnosis of Rolling Bearings Based on Machine Learning Algorithm. Journal of Qingdao University (Natural Science Edition), vol. 034, pp.15-22, February 2021.

[3] Cheng Ping, Yan Lu. Research on Performance Evaluation of Enterprise R&D Projects Based on CART Decision Tree Algorithm. Accounting Monthly, vol.8, pp.30-37,April 2022.

[4] Xiao Ling, Liu Jihong, Yao Jianchu. Research and Application of Distributed Database System. Computer Engineering, vol. 8, pp.33-35,January 2021.

[5] Duan Xiaocong. Research on encrypted storage of distributed database based on wireless sensor network clustering strategy. Journal of Sensor Technology, vol. 35, pp.1728-1732, December 2022.

[6] Liang Yi, Chen Youyong, Dong Xiaoyu, Zhang Fulin. Design and Implementation of Distributed Database Synchronization Middleware System. Microcomputer Applications, vol.38, pp.190-192,November 2022.

[7] Liu Yali. Research on Multi-layer Data Synchronization Mechanism of Distributed Database Based on ETL and XML Technology. Electronic Design Engineering, vol. 8, pp.30-38,June 2022.

[8] Miao Yan, Wang Heping. Research on Distributed Real-time Database Based on Access Integration Algorithm. Electronic Design Engineering, vol. 30, pp.127-130,January 2022.