Efficient Processing of Large Uncertain Data using Map Reduce Framework

M.Blessa Binolin Pepsi[1], P.Jothi[2], K.Malini[2], S.Maariswari[2]
[1]Assistant Professor (Senior Grade) [2]UG Scholar,
Department of Information Technology
Mepco Schlenk Engineering College, Sivakasi,
Tamilnadu, India.

Abstract: In this project, the dominant values are retrieved from the big dataset that has random missing values using map reduce framework. Retrieving missing values in the dataset is the difficult process when the dataset becomes large. Missing nodes will be randomly distributed in its dimensions. They are many algorithms have been introduced to recover dominant values but they are applicable only for small datasets. Top N dominating query is possible for Bitmap index guided algorithm but it has not been intended to recover the dominating values in the incomplete dataset. Instead this algorithm increases the performance of incomplete dataset. Skyband based algorithm and upper bound based algorithm have also been made possiblefor Top N dominating query but their performance is very weak. In order to overcome these issues this paper proposes the Map Reduced Enchanced Bitmap Index Guided Algorithm.For applying the Top N dominance query on incomplete dataset MRBIG algorithm uses the MapReduce framework for big datasets. Several nodes have been involved in the mapreduce computing process.Compared to the previous algorithm MRBIG algorithm has the faster processing time for finding the Top N dominating queries.

Keywords:Hadoop,mapreduce,bigdata,query processing, bitmap indexing.

I. Introduction

In a given dataset contain multiple dimension values, the values are required to find the most dominant values throughout the datasets. Based on the dominance definition better values can be expressed. The movie database with different movie rating given by the users is the sample of multi- dimensional dataset. Top-n dominance query retrieve the dominant values which dominate the values in the same dimension by using a scoring function. Based on the dominance definition the dominant values are distinguished.

The top-n dominance query can be established by using different algorithm and methods. To find the dominant values in the dataset pairwise comparison is the best approach. The pairwise comparison method is used to compare each two different items with same dimensions in the dataset. But this method is efficient for massive dataset and the processing time is larger. Therefore this approach is not the best way.

To apply the top-n dominance query with best performance several algorithms have been proposed. The sky band algorithms separate the values into different buckets based on the dimension values. The bucket method is easier to process. Each bucket contain the separate dominant values combine them together and then determine top-n values of the whole dataset. Another method is upper bound based algorithm which is based on bitwise comparison. The BIG algorithm improve the performance of top-n dominance query.

II.Related work

In this paper, Top N dominance, incomplete data, bitmap indexing and mapreduce are the related works that are discussed below.

Top N dominance

The better value in recovered by the Top N dominating query based upon the goodness term in each use case. The predefined definition clearly explains the dominance admittance which also gives the relationship of the best dominance. In the approach described in [6] , top-n dominating (TND) query gives the n items that dominate the maximal number of objects in a dataset.M. L. Yiu and N. Mamoulis[7]reviewed thequery is an important tool for decision support since it provides data analysts an intuitive way for finding significant objects. It combines the advantages of top-n and skyline queries without sharing their disadvantages:(i) the output size can be controlled,(ii) no ranking functions need to be

specified by users.. X. Han, X. Liu, J. Li, and H. Gao,[8] proposed that top-*n* query is an important operation to generate a set of interesting points in a large data space. To compute top-*n* results on massive data the TKAP algorithm is used which occupy low space for some datastructures.To retrieve a better pruning effect the information obtained by round-robin are adjusted by adaptive pruning operation**.**

Incomplete data

Missing values in the dimension is one of the important factors of the incomplete data. We must be in the position to manage the issues such as the presence of missing values.It requires some of the innovative methods to remove the missing values. F. Bu, Z. Chen, Q. Zhang, and X. Wang,[1] proposed the hierarchical clustering-based feature subset selection algorithm is designed to reduce the dimensions of the data set. Incompleteness is the nature of the dataset.Both the present and missing values are not compared to each otherand it also not meant for incompleteness.M. E. Khalefa[5] to remove a search space of huge numbers of multi-dimensional data items the skyline query algorithm is introduced, to a small set of items eliminate the items that are dominated by others.Relating Top N dominance with incomplete data will show the major difference between the MRBIG algorithm and the previously mentioned works.

Bitmap indexing

Bitmap Indexing is a special type of database indexing that uses bitmaps.This technique is used for huge databases, when column is of low cardinality and these columns are most frequently used in the query.Incomplete data can be easily processed by the bitmap indexing.Bitmap indexing is dangerous in case of Bigdata.Data patterns are generated by using bitwise operation in bitmap indexing. K. Wu, E. J. Otoo[2] proposed Specialized compression schemes, like the byte-aligned bitmap code (BBC), are usually faster in performing logical operations than the general purpose schemes, but in many cases they are still orders of magnitude slower than the uncompressed scheme.Bitmap indexing highly compressed structure, making them fast to read and their structure makes it possible for the system to combine multiple indexes together.IBIG algorithm which uses the improved version of bitmap index table take place after conducting the compression on the Bitmap indexing

Mapreduce

Mapreduce is a programming model. This model used a parallel distributed algorithm on a cluster for processing large datasets. Mapper and Reducer are the two roles performed by Mapreduce. The map task is done by means of Mapper Class. The reduce task is done by means of Reducer Class. D. Lopez proposed a Mapreduce Ambient intelligence that provides advances in sensors and sensor networks, pervasive computing, and artificial intelligence to capture the real time climate data.The tool used in this paper is Apache Hadoop. A.S.Ashour [4] proposed a recursive naive algorithm is used for generating K-mers in mapping phase.

III. Objective

The goal of this work is to obtain the dominant values in incomplete dataset that has random distributed missing nodes in its dimension through query processing and MapReduce framework. In the preprocessing phase the missing values have been removed from the dataset and it will be loaded for retrieving dominant value.

The method must satisfy these requirements:

1. To recover the highest values in the given dataset.
2. To perform Map Reduce framework to improve the performance of applying Top N dominance queries in large incomplete dataset.

In this proposed technique,Incomplete data clustering plays an important role in the big data[1] and theclustering algorithm based on feature selection and partial distance strategy. To return top-n values with a highest domination score in huge data space use domination query.

IV. Problem Definition

It consists of two terms in the incomplete dataset R. The incomplete dataset contains p dimensions and q items. Our objective is to recover the highest values or dominant values in the dataset. Let us the consider the item q1 with four dimension with the missing values in it. It can be represented as the following,

Q1=(q1,q2,q3,q4)=(7,2,--,1)

In this missing value will be represented by using dash. If q1 is dominant over the q2 it will be represented as q1>q2.

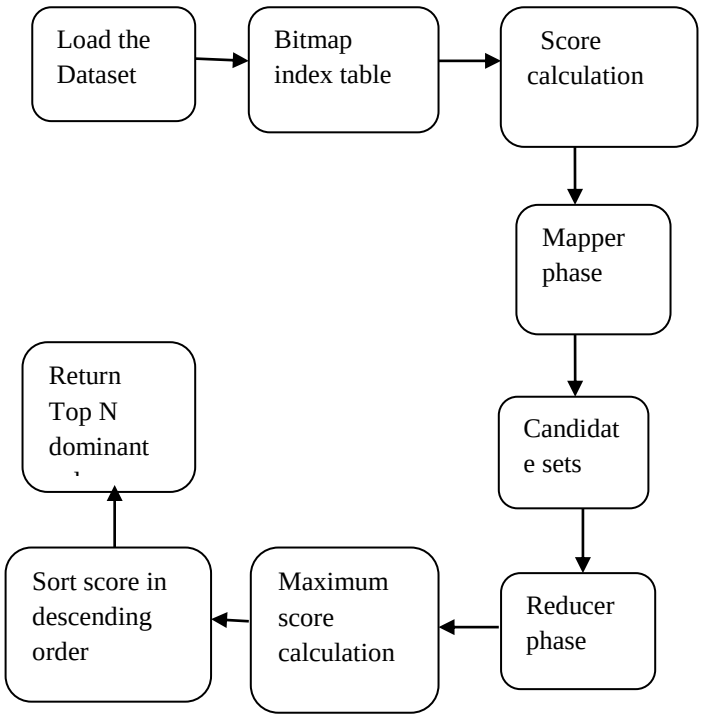
Dominance Definition: Consider the two items q1 and q2.The statement q1 dominates over q2 will perfect if and only if exclude all the missing values in the dimensions and check the dimension value of q1 is greater than the dimension value of q2.

Dominance also refers to finding a better value (i.e the greater value) in the dataset. Strength of the value is also determined by the dominance definition. Let us consider the two item q1 and q2 with the dimension for better understanding,

q1=(4,0,-,1) and q2=(1,-,-,0)

By comparing above two items the dominance is given to the q1[p1] in comparison with q2[p1].q1 dominates over the q2 because is has the larger value in it. Result of the process will not be changed if the missing values are not considered.

V. Proposed System Design



The block diagram of the proposed system is given in Fig. 1

Skyband Based Algorithm:

To retrieve the top-n dominance query in the given dataset, by using a pairwise comparison the whole dataset can be compared and it is the best approach .The pairwise comparison method is applicable only for small datasets and for massive datasets causes poor performance .There are many errors in this approach, to evaluate single values in the dataset the pair wise comparison need long runtime.

1) Based on their missing values categorize the data into various parts by using normalization method
2) The data are arranged in a bucket from the given dataset based on their missing values pattern.
   For example: (-,1,3,8) and (-,2,4,4) due to having the missing values in the similar dimension both items are directed to the same bucket
3) For each bucket the candidate sets are created , all the values in the bucket are considered and the results will be a dominant value

TABLE1: Sample dataset with missing values

	p1	p2	p3	p4
q1	-	1	2	-
q2	2	3	-	-
q3	-	2	3	-
q4	1	2	-	-

In the above table,

q->items p->dimensions - ->missing value

TABLE2: Bucketing method

b1			
-	1	2	-
-	2	3	-

b2			
2	3	-	-
1	2	-	-

In table2, q1 and q3 have been bucketed into the same bucket b1 since they have the same missing data dimensions.

TABLE3: Candidate set

b1			
-	2	3	-
-	1	2	-

b2			
2	3	-	-
1	2	-	-

In table3, both p2 and p3 dimension values in the item q3 are greater than the dimension values in the item q1,it is similar to sorting method based on this

the candidate sets are created. The missing values are not considered.

The processing problems will be occurred due to large dataset, the sky band algorithm is applicable only for smaller datasets these are the defects in sky band algorithm.

Upper Bound Based Algorithm:

To obtain top-n dominance query on incomplete dataset upper bound based is one of the available algorithm. This algorithm is not based on the pairwise comparison. In the given dataset the top-n dominant value is retrieved based on the dominance definition. The goal of the dominance definition is to define the criteria that work uniquely. To find the final top-n dominance value the score is allotted. A score is a number that is assigned to each dataset. Only fewer calculations are taken by allocating the score values and it also eliminates the pair wise comparison.

The frequency of the items is identified based on the upper bound based algorithm. For complete data the values are compared with one another in the same dimension. For incomplete data each dimensions are considered independently and the compare with other values to find the frequency of values it dominates.

The real world problem is the upper bound based algorithm requires large processing procedure for massive datasets.

Bitmap index guided algorithm

The methods present in previous section have some disadvantage like inefficiency.In order to overcome that this algorithm take place.The previously discussed algorithm such as the skyband based algorithm uses the pairwise comparison method which is inefficient and applicable only for small datasets.Upper bound based algorithm will be efficient only if they is numerous comparison between values.The Top N dominant values can be generated by using the Bitmap index table in bitmap index guided algorithm.

For constructing a bitmap index table following conditions has to be satisfied.

First the Bitmap index table should be initialized with 0.

The values in the table will be modified based on the item present in it.

- If the missing value if found, leave the respective row without any changes.
- If the number if found,then the number 1 should be inserted in the respective row and following right rows will be filled accordingly.

Bitmap index table is not possible while dealing the large datasets which has thousands of rows and columns in it. Apart from limitations, this algorithm has best performance.In order to avoid the overabundance we are moving to the next algorithm.

BITMAP INDEX TABLE:

Items	p1	-	1	2	3	p2	-	1	2	3	p3	-	1	2	3	p4	-	1	2	3
q1	-	0	0	0	0	1	0	1	1	1	2	0	0	1	1	-	0	0	0	0
q2	2	0	0	1	1	3	0	0	0	1	-	0	0	0	0	-	0	0	0	0
q3	-	0	0	0	0	2	0	0	1	1	3	0	0	0	1	-	0	0	0	0
q4	1	0	1	1	1	2	0	0	1	1	-	0	0	0	0	-	0	0	0	0

Algorithm1: Bitmap Index Guided

1. calculating the top-n dominating values
2. create movie and user
3. for each row
4. for each column
5. var1 ← the first value in the row
6. var2 ← index of the var1
7. movie ← join the column value with var2
8. user ← join the column value with var 2 + 1
9. create the non-dominating function in each row
10. movie∩ movie*
11. user∩ user*
12. continue the procedure up to final column
13. alpha = movie ∗ −user ∗ −nonDominant
14. beta = count the rated movie
15. rating = alpha+beta
16. count the maximum rated movie and maximum rated users
17. continue the procedure upto final row
18. sort the rated movie data and rated users data in descending order
19. return the top-n dominating values

MRBIG: Mapreduce Enhanced Bitmap Index Guided Algorithm

In order to handle the large datasets, MapReduce framework has been introduced. MapReduce frame work has two section called mapper phase and the reducer phase. Mapper task is the first phase of processing that processes each input record and generates an intermediate key-value pair. Reducer takes the output of the Mapper process each of them to generate the output. The final result of the reducer can be calculated by grouping the result of mapper. Time inexpedient is one of the drawbacks for mapreduce framework. One of the general examples of the Mapreduce framework is the word count problem. First the dataset will be splitted into different sections. Mapper starts performing the word counting. The number of appearances of the will be shown. The result will be sorted and shuffled. The reducer joins all the results and combines with the mapper section to produce final output.

Simple programming methods are used to deal with the large datasets. This is the proposed algorithm for finding the Top N dominant value in big dataset.

Consider the dataset k with p dimensions and q items. The size of the dataset will be huge compared to datasets previously taken. Two dimensional matrix is constructed to accommodate the rows and columns. The n_i present in the Table 1 shows range of all the present values in the dimension i.e., .Dynamic bitmap index table can be created with the help of the dataset values. To generate the bitmap index table need the count of the columns. Based on the dominance definition, scoring method is used to evaluate the values based on the power. It compares the values by the score. According to the dominance, higher scores are better.

MRBIG algorithm deals with three internal sets which help us to obtain Top N dominant values in the dataset for each object j. This algorithm has two important set [R] and [S].The set which is not better than j is [R] and it is defined as the set of objects. The set of the object which is worse than object j is [S].The set of object which is not dominated by the object j is [nonDomin].The main objective of the MRBIG algorithm is to find the [R*] and [S*] and to calculate score for Top N values. The process of the MRBIG algorithm is faster because it divides the sets and sent it to different computational nodes for further processing.

Algorithm 2: MapReduce Enhanced Bitmap Index Guided Algorithm

1. create [movie*] and [user*]
2. for each row n_i in $\{n_1,n_2,\ldots..,n_n\}$ continue
3. Mapper Class:
4. call the map function(each column split by dimensions)
5. call the bitmap functions(pass the column values)
6. for each column dim_jcontinue
7. create [$movie_j$*] ,[$user_j$*] and [$nonDomin_j$]
8. for each row
9. candi ← index(the first value in the row)
10. [movie] ← join the column value with candi
11. [user] ← join the column value with candi + 1
12. end
13. end
14. Reducer Class:
15. for each row
16. $movie_j$∩ movie*
17. $user_j$∩ user*
18. end
19. lamda=[user*] – [movie*]
20. for each row in lamda
21. Ω = count the lamda if lamda is greater than n_i
22. nonDomin← $lamda_i$
23. end
24. alpha=[lamda - nonDomin]

25. beta=count(movie* - Ω)
26. rating = alpha + beta
27. count the maximum rated movie and maximum rated users
28. continue the procedure upto final row
29. sort the rated movie data and rated users data in descending order
30. return the top-n dominating values

VI. Performance Analysis

Using real and synthetic datasets, the performance of the BIG and MRBIG algorithm is compared. The missing rate standard deviation, the standard deviation for each dimension, different values for the inserted numbers are the parameters involved in the Movielens dataset.
Table: Artificial Dataset Information

Number of users	Number of movies	Number of users	Number of movies
250	3000	5500	100
500	3000	5500	1000
1500	3000	5500	2000
2500	3000	5500	3000

Table: Sample Dataset

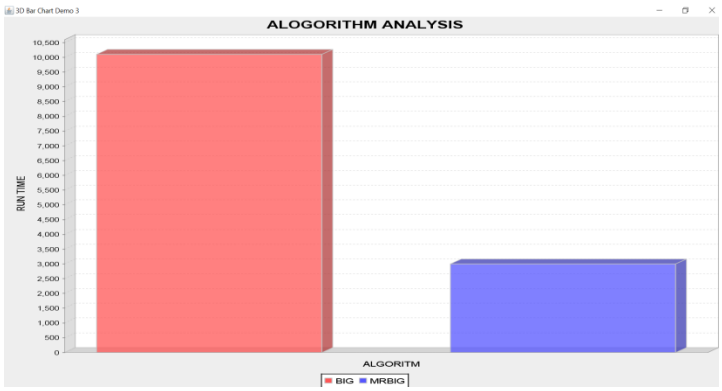
Raw Input

User UniqueId	Movie uniqueId	Ratings
100	1	1
100	2	2
100	3	3
200	4	4
200	5	5
80	5	1
80	3	4

Processed Input

	1	2	3	4	5
100	1	2	3	-	-
200	-	-	-	4	5
80	-	-	4	-	1

The Movielens dataset consists of user identification, movie identification and the rating information.As shown in the processed input the user 100 has submitted three ratings for the movies 1,2,3.but not have given rating for the movies 4 and 5.The rating for the movie 4 and 5 are the missing values and the corresponding row will not exist any more.The removal of special characters and seperators are done by the formatting process.The preprocessing method take place to make the data analysis accurate by comparing the uniform dataset with the same format of it.



.

Fig 2: Comparing BIG and MRBIG algorithm. Blue bars show BIG and pink bars shows the MRBIG.

In order to get a desired top N value, multiple execution of the MRBIG algorithm is required.As shown in the figure,when the size of the dataset increases BIG algorithm slows down the process while the MRBIG algorithm handles the large dataset and speed up the process.

VII. Conclusion

In this paper,proposed algorithm is to find the Top N dominant values using MapReduce framework model.This process involves query processing,Bitmap indexing,MapReduce and store the Top N dominant value.The key advantage of the skyband based algorithm and upper bound based algorithm are BIG and MRBIG algorithm to give exact result.

VII. Reference:

[1]. F. Bu, Z. Chen, Q. Zhang, and X. Wang, ``Incomplete big data clustering

algorithm using feature selection and partial distance," in *Proc. 5th Int.*
Conf. Digit. Home, 2014, pp. 263_266.

[2]. K. Wu, E. J. Otoo, and A. Shoshani, ``Compressing bitmap indexes for
faster search operations," in *Proc. 14th Int. Conf. Sci. Statist. Database*
Manage., 2002, pp. 99_108.

[3]. S. Kamal, S. H. Ripon, N. Dey, A. S. Ashour, and V. Santhi,
``A mapreduce approach to diminish imbalance parameters for

big deoxyribonucleic acid dataset," *Comput. Methods Prog.*
Biomed., vol. 131, pp. 191_206, Jul. 2016.

[4]. M. S. Kamal, S. Parvin, A. S. Ashour, F. Shi, and N. Dey, ``De-bruijn graph
with mapreduce framework towards metagenomic data classi_cation," *Int.*
J. Inf. Technol., vol. 9, no. 1, pp. 59_75, Mar. 2017.

[5]. M. E. Khalefa, M. F. Mokbel, and J. J. Levandoski, ``Skyline query
processing for incomplete data," in *Proc. IEEE 24th Int. Conf. Data Eng.*,
Washington, DC, USA, Apr. 2008, pp. 556_565.

[6]. X. Miao, Y. Gao, B. Zheng, G. Chen, and H. Cui, ``Top-*k* dominating
queries on incomplete data," in *Proc. IEEE 32nd Int. Conf. Data*
Eng. (ICDE), May 2016, vol. 28. no. 1, pp. 1500_1501.

[7]. M. L. Yiu and N. Mamoulis, ``Multi-dimensional top-k dominating
queries," *VLDB J.*, vol. 18, no. 3, pp. 695_718, 2009.

[8]. X. Han, X. Liu, J. Li, and H. Gao, ``TKAP: Ef_ciently processing topk
query on massive data by adaptive pruning," *Knowl. Inf. Syst.*, vol. 47,
no. 2, pp. 301_328, 2016.