Methods For Evaluating Iceberg Queries Using Decimal Index

Dr. A. Padmapriya,M.C.A.,M.Phil.,Ph.D[#1], T. Shanmugapriya[#2]
[#]*Department of Computer Science and Engineering, Alagappa University*
Karaikudi

Abstract—*Decision support and knowledge discovery system often compute aggregation values of interesting attributes by processing a huge amount of data in very large database and or warehouses. In particular iceberg query is a special type of aggregation query that compute aggregate values above user provide threshold. This paper proposed a decimal index to process Iceberg queries. Because it occupies less memory space and less processing time. We exploited the property of decimal index and developed a very efficient algorithm for processing iceberg query*

Keywords-*Iceberg Query, Counting co-occurrence, Bitmap index, decimal index*

1. INTRODUCTION

Data mining is the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cuts costs, or both. Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases.

Data mining software analyzes relationships and patterns in stored transaction data based on open-ended user queries. Several types of analytical software are available: statistical, machine learning, and neural networks.

1.1 Uses of data mining

Data mining brings a lot of benefits to businesses, society, governments, sales, marketing, insurance, health care, transportation and medicine and so on.

- ***Market segmentation:*** Identify the common characteristics of customers who buy the same products from your company.
- ***Fraud detection***: Identify which transactions are most likely to be fraudulent.
- ***Direct marketing***: Identify which prospects should be included in a mailing list to obtain the highest response rate.
- ***Banking/Finance:*** Used to identify customer loyalty by analyzing the data of customer purchasing activities.
- ***Interactive marketing***: Predict what each individual accessing a Web site is most likely interested in seeing.
- ***Health Care and Insurance***: The growth of insurance industry entirely depends on the ability of converting data into the knowledge, information or intelligence about customer, competitors and its markets.
- ***Market basket analysis***: Understand what products or services are commonly purchased together; e.g., beer and diapers [21] [22].

Data mining consists of two types of Counting Co-occurrences for frequent item set and iceberg query.

Iceberg queries are a special case of SQL queries involving GROUP BY and HAVING clauses, wherein the answer set is small relative to the database size. Iceberg queries have been recently identified as important queries for many applications. It computes an aggregate function over an attribute or set of attributes in order to find aggregate value above threshold.

The recent research [12] [17] has paid attention to iceberg problem. Iceberg problem in database means the relation between a lot of data and few results is similar to it between an iceberg and tip of one. Iceberg queries were introduced in[1].these queries have three properties,(1) Computing aggregate function (2) about large data (3) returning results above threshold. By the following cases it is necessary to compute them. One is when the amount of data is very large like data warehouse [12] [17].

The prototypical iceberg query the paper considers (can be easily extended to the other forms of iceberg queries) is:

```
SELECT     attr1, attr2, …., attrk, COUNT ( rest)
FROM       R
GROUP BY attr1, attr2, attrk
HAVING     COUNT (rest) >= T
```

Where R is a relation that contains attributes attr1, attr2, ..., attrk, rest and T is a threshold.

AVG, SUM, COUNT, MIN, MAX are aggregate operator the relation is grouped according to the value of the attribute and provide a condition. Grouping and having should be implemented together.

State (attr1)	Job (attr2)	Salary(rest)
Tamilnadu	Professor	1000
Delhi	Doctor	2000
Kashmir	Professor	1000

SELECT state, job, count (salary) from salary info group by job having count (salary)>=T.

The query Exection engine takes a query evaluation plan, execute that plan and return the answers to the query.

2. REVIEW OF RELATED WORK

A handful of researches are available in literature for iceberg queries. In recent times the evaluation of iceberg queries in distributed manner has attracted researchers significantly due to the demand of scalability and efficiently. Here we review the recent work available in the literature for evaluation of iceberg queries.

There are results which are showing that executing iceberg queries on data takes more time than finding the dataset. In this section we will list some of those methodologies recently appeared in the literature.

The relational database system like ORACLE, SQL SERVER, and MYSQL are using general aggregation algorithms [10] [23] to answer the iceberg queries. Many practical application including data warehousing [1], market-basket analyses [21] rely on iceberg query. Iceberg queries were Introduced in [16] and iceberg CUBE problem introduced in [12].The recent research [16] [12] has paid attention to iceberg problem. Iceberg problem in database means relation between a lot of data and few results is similar to it between an iceberg and the tip of one.

Recently, [11] a variant of the problem, called iceberg data cube computation was introduced by BUC. In order to meet similar objectives, in [12] proposed “multifeature cubes”. When computing such cubes, aggregates not satisfying a selection condition specified by user (similar to the clause having in SQL) are discarded.

From the previous works, it is known that static index pruning techniques can reduce the size of an index (and the underlying collection) while providing comparative effectiveness performance with that of the unpruned case [3, 5].

In author presented a strategy to efficiently answer joint queries on both structured and text types of data. The records in data warehouses are usually extracted from other database systems and therefore contain only what is known as structured data [7,8, 20]. A large amount of text document is inadequate for processing efficiently joint queries over structured and text data.

In general, these strategies appear wasteful since they do not take the threshold predicate into account, that is, they are not output sensitive. In case of an iceberg query involving a join of multiple base relations, the iceberg relation ***I*** is derived from the base relations ***B*** using one of the efficient join algorithms: sort-merge join, hybrid-hash join, and others mentioned in [10].

For characterizing cuboids, author state an equivalence between our representation and the result of the aggregate formation defined by [2] which is chosen because it is on one hand the original definition of the aggregation operator in the relational algebra [14][15].

The number of tuples satisfying the query is very less compared to the size of the database,[13] coin the term N-iceberg(Negative) queries for such a type of queries [14] proposed an algorithm to evaluate N-iceberg queries and compare them with ORACLE and traditional sorting algorithms, with very little main memory.

With the rapid increase of the databases and data repositories sizes, new types of queries have been emerged where the output is significantly small compared to the input. Iceberg queries have been recently identified as important queries for many applications belonging to this category. These applications can be found in data mining [1],information retrieval [18], decision support and data warehouse [4], web mining and top k queries [7, 8]. The iceberg queries are formally introduced by Fang et al. [9]. Detailed application examples have been also presented in [10]. These queries have been extended to data cubes in [4].

3. MOTIVATION BEHIND THE APPROACH

The data storing and retrieving are playing a major role in the data clustering and data ware housing techniques. The effectiveness of data retrieving of the method is limited or less amount of time. Iceberg query retrieving data from defined database is done with the help of database queries.

Today’s bitmap indices can be applied on all types of attributes.Studes have shown that compressed bitmap indices occupy less space than the raw data. Bitmap are provides better query performance. Nowadays bitmap index is supported in many commercial database systems (e.g., ORACLE, Informix) and so on. A bit map index is a data structure used to efficiently access large database.Generally,the purpose of an index is to provide pointers to row in a table containing given key values. In a common index, this is achieved by storing a list of records for each key corresponding to the row with that key value. So in this research paper overcome some complex in iceberg query bitmap index.

4. PROPOSED WORK

In proposed work have been following four modules

i. Dataset Data collection
ii. Bitmap index
iii. Size modification
iv. Decimal index

4.1. Dataset data collection

In this research work example data set is salary_info. It contains three column consist of state, job, rest.state contains employee working place information and job consist of professor, doctor…and salary gives salary information. So the dataset have collection of information about specific threshold. In that thesis consist of salary detail. So retrieve the salary detail than it is complex one. So now use in aggregation function.

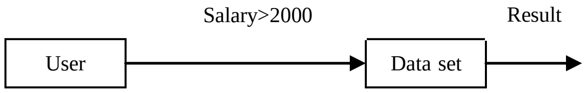
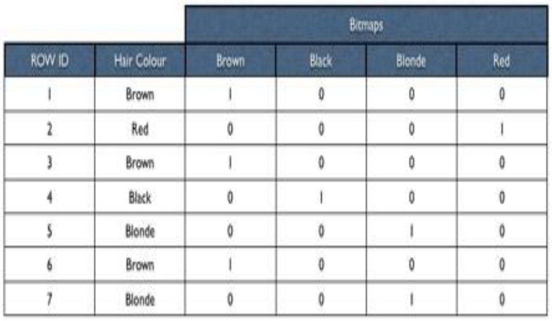


Fig: Data collection

4.2. Bitmap index

Bitmap index used to gather the data very easily. Bit map index consist the value of 1 and 0.in existing bitmap index stored data use in column and row. So it occupies more spaces and processing time always more. And also bitmap index stored data type is CHAR ().in which method storage each value needed into 2 BYTE character's memory always large.



		Bitmaps			
ROW ID	Hair Colour	Brown	Black	Blonde	Red
1	Brown	1	0	0	0
2	Red	0	0	0	1
3	Brown	1	0	0	0
4	Black	0	1	0	0
5	Blonde	0	0	1	0
6	Brown	1	0	0	0
7	Blonde	0	0	1	0

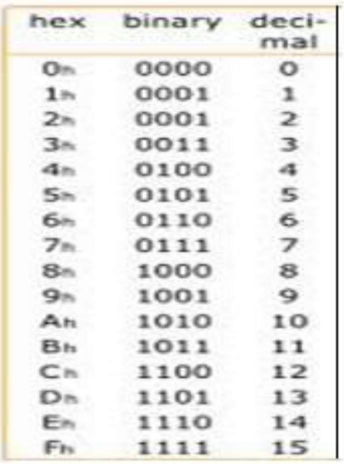
Table1: Bit Map index sample table

4.3. Data Type Conversion

Character data type data value storing processes need 2 Byte data size. So character data type converts to integer. Because integer data type get in 4 byte data size value. Convert character to integer than it occupies less memory and more data size spaces.

4.4. Decimal index Creation

Decimal index used to convert the bit value indo integer value. So it occupies the less memory spaces. Each row consists of equal integer value. So bit value converts in to decimal integer value than it have single row only so table memory space is less.



hex	binary	deci-mal
0_h	0000	0
1_h	0001	1
2_h	0001	2
3_h	0011	3
4_h	0100	4
5_h	0101	5
6_h	0110	6
7_h	0111	7
8_h	1000	8
9_h	1001	9
A_h	1010	10
B_h	1011	11
C_h	1100	12
D_h	1101	13
E_h	1110	14
F_h	1111	15

Table2: Binary value Convert into decimal value

4.5 Results

The main objective of Iceberg queries is to retrieve data quickly. Query optimization is the refining process in database administration and it help bring down speed of execution. Data mining techniques are often measured by their speed. The reason behind this is, it is faster and the tool can run on larger data set. Iceberg queries are generally very expensive to compute since they require several scans of relations.

The common objectives for any iceberg query using decimal index is as mentioned below

- Speed of execution is increased
- Ability to process large data set
- Reduce the number of scans in data set

5. CONCLUSION

This paper gives brief introduction about data mining uses of data mining. The need for ice berg queries and algorithm employed for evaluation of iceberg queries. The objectives of iceberg queries are studied in this paper. The paper propped a decimal index based iceberg query evaluation method. The main goal of using decimal index is it occupies less memory space and also speed up the query evaluation process.

6. REFERENCES

[1] Agrawal, R. and Srikant, R. "Fast Algorithms for Mining Association Rules." Proceedings of the 20th Int'l Conference on Very Large s Databases (VLDB '94), September 1994.

[2] A. C. Klug. "Equivalence of Relational Algebra and Relational Calculus Query Languages Having Aggregate Functions". Journal of ACM, 29(3):699–717, 1982.

[3] Altingovde, I. S., Ozcan, R., Ulusoy, Ö.: "Exploiting query views for static index pruning in web search engines". In: Proc. of CIKM'09. (2009) 1951-1954

[4] Beyer, K. and Ramakrishnan, R. "Bottom-up Computation of Sparse and Iceberg CUBEs." Proceedings of 1999 ACM SIGMOD Int'l Conference on Management of Data, pp. 359-370, 1999.

[5] Carmel, D., Cohen, D., Fagin, R., Farchi, E., Herscovici, M., Maarek, Y. S., Soffer, A.," Static index pruning for information retrieval systems. In: Proc. of SIGIR'01. (2001)

[6] Comer, D.: The ubiquitous B-tree. Computing Surveys 11(2), 121–137 (1979)

[7] Chaudhuri, S. and Gravano, L. "Evaluating Top-A: Selection Queries." Proceedings of the 25th Int'l s on Very Large Databases (VLDB '99), pp. 399-410, 1999.

[8] Donjerkovic, D. and Ramakrishnan, R. "Probabilistic Optimization of Top n Queries." Proceedings of the 25th Int'l Conference on Very Large Databases (VLDB'99), pp. 411-422, 1999.

[9] Fang, M., Shivakumar, N., Garcia-Molina, H., Motwani, R. and Ullman, J. "Computing Iceberg

Queries Efficiently." Proceedings of the 24th Int'l Conference on Very Large Databases (VLDB '98), 1998.

[10] G. Graefe, "Query Evaluation Techniques for Large Databases", ACM Comput. Surv., 25, 2, 73–170, June 1993.

[11] Kevin S. Beyer and Raghu Ramakrishnan "Bottom-up computation of sparse and iceberg cubes". In Proc. of the Int. Conf. on Management of Data (ACM SIGMOD),pages 359-370, 1999.

[12] K.Beyer and R.Ramakrishnan,"Bottom-Up Computation of sparse and iceberg CUBEs",In Proc.of the ACM SIGMOD Conf., Pages 359-370,1999.

[13] Leela krishna poola"Efficiently evaluating N-iceberg queries".

[14] L. Cabibbo and R. Torlone. "A Framework for the Investigation of Aggregate Functions in Database Queries". In C. Beeri and P. Buneman, editors, ICDT'99, Jerusalem, Israel, LNCS vol. 1540, pages 383–397.

[15] L. Libkin, L. Cabibbo" the aggregation operator in the relational algebra ".Springer Verlag, 1999.

[16] L. Libkin. Expressive Power of SQL. In ICDT'01, London, UK, LNCS vol. 1973, pages 1–21. Springer Verlag, January 2001.

[17] M.Fang,N.Shivakumar,H.Garua-Molina,R.Motwani,and J.D.Ullam,"Computing iceberg queries Efficiently",In Proc.of 24th VLDB conf..,Pages 299-310,1998.

[18] R. Ng, A. Wagner and Y. Yin, "Iceberg-cube Computation with PC Clusters", Proc. of ACM SIGMOD Conf., 2000.

[19] Salton, G. "A Theory of Indexing." Society for Industrial and Applied Mathematics, 1975.

[20] Selinger et al., "Access Path Selection in a Relational Database Management System", Proc. of ACM SIGMOD Conf., 1979.

[21] Shoshani, A.: "OLAP and statistical databases: similarities and differences. In: Principles Of Database Systems (PODS)", pp. 185–196 (1997)

[22] S. Brin, R. Motwani, J.D. Ullman, and S. Tsur".Dynamic itemset counting and implication rules for market basket data". In Proc. of the Int. Conf. on Management of Data (ACM SIGMOD), pages 255-264, 1997.

[23] W. P. Yan and Larson, "Data Reduction through EarlyGrouping", In CASCON, page 74, 1994.