International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611

Efficient Approximate Processing of Queries in P2P Networks

Suraj N. Arya[1], Rajesh V. Argiddi[2]

[1, 2]Solapur University, Walchand Institute of Technology, Seth Walchand Hirachand Marg, Ashok Chowk, Solapur-413006, India

Abstract: ***Peer-to-peer (P2P) databases are most probably used nowadays on the Internet for distributing and sharing documents, applications, and other organizational data. Finding the answers to large-scale ad hoc queries as aggregation queries on databases often gives rise to different challenges. Finding the exact solutions consumes a large amount of time and also is difficult to implement since P2P databases are often dynamic and distributed. In this paper, an approach for approximately answering ad hoc queries in dynamic and distributed databases is presented. Generally, the data is distributed across different peers in a distributed environment, and majority of the times, within each peer, the data is correlated up to a large extent. This fact is taken use of and an approach for processing queries in such environment is described in this work. Also, the results of the work are presented thereafter.***

Keywords: Peer-to-peer databases, distributed query processing, ad hoc queries, aggregation queries.

1. Introduction

Peer-to-Peer (P2P) databases are used up to a great extent these days for sharing of files and partitioning and data distribution over the network. A P2P network comprises of a number of peer nodes which share data and resources with other peer nodes. It is also possible to establish a hierarchy amongst the nodes, if there is such a requirement. Usually, in typical client-server models, major role is played by the server by providing services to different clients and also processing the queries which are initiated from the clients. Generally, in a P2P network, there is no any central authority for doing such administrative tasks. This results in poor coordination or lack of coordination amongst the different peer entities. Also, there is no performance bottleneck due to absence of failures as there is no central authority entity. Such networks can be regarded as scalable, and dynamic, and because of this, no huge difference is made in performance if a few nodes join and/or leave the network.

The important applications of P2P databases include those involving tasks like sharing of files and retrieval of relevant data. Those applications involving such types of databases include such queries as aggregation queries. Processing the aggregation queries has a large scope in applications of areas such as decision support, data analysis and mining, etc. Aggregation queries may also be used up to a large extent in sensor networks for detection of temperature and anomalies. These may also be used in various Intrusion Detection Systems.

The system described in this paper aims to keep updated information about the peer nodes in the network and also retrieve and analyze the results for aggregation queries at the target nodes. It is expected that the results for the aggregation queries be generated quickly so that they may support real-time systems. If the relevant information about the different active peers in the peer-to-peer network and the relevant information about the databases on each peer are known, then approximate results could be generated with lower risks. Aggregation queries may have the following simple general form:

```
SELECT  aggreg_oper (col_name)
FROM    T
WHERE  selection_condition;
```

In the above query, T denotes the name of the table that may be distributed over the P2P system, most probably making use of horizontal partitioning. The aggreg_oper is any aggregation operation such as MAX, MIN, COUNT, AVG, SUM, etc. The col_name denotes the name of that column on which the required aggregation operation is to be applied. Such type of query needs to be transformed into multiple queries such that each of them may have to be operated on different partitions of the original database.

The architecture of the system built is explained first with the help of the modules that it contains. Then, the results of the system are shown and explained.

2. Literature Review

Various sampling-based adaptive techniques were presented for the answering of queries in P2P database system of the ad hoc aggregation type. Comparatively, lesser number of messages was required for sending over the network and in accordance to it, tunable parameters were expected to be provided so as to maximize the performance for the various network topologies [1]. An efficient system was built which made possible efficient searches of large numbers of data providers on the internet. Every data source or data provider can become an autonomous node in a large peer-to-peer network [2]. Indices were utilized on each node and the relevant queries were accordingly directed from any node where the query was submitted to the respective relevant sources. Such an approach was found to have a high degree of feasibility especially in those applications involving a large peer-to-peer network. A system that made use of an adaptive two-phase sampling approach was developed which was based on random walks of the P2P graph [3]. It also made utilization of the block-level sampling techniques.

Firstly, query routing and then accordingly, its processing

form the main problems that arise because of the absence of a global catalog in a P2P network. Relevant solutions were proposed for efficiently handling these problems and accordingly generating the results for the queries that are initiated by the user [4]. A protocol was proposed for the participants in a network to build P2P networks in a distributed fashion that resulted in connected networks of a constant degree [5]. This resulted in an efficient search and data exchange in the network. Also, global knowledge about all the peer entities in the network was not a compulsory requirement that is to be known previously.

A framework was developed so as to classify the current and future P2P network technologies. The main task that was included in this work was identifying the important basic characteristics of the P2P network applications [6]. The infrastructure that may be developed on this idea may focus on computing in the P2P network. A Decision Support System named Aqua was designed for providing fast approximate answers to the aggregation queries [7]. Such queries are generally applied in OLAP applications. Special statistical summaries called synopses of the original data are computed in advance and then stored in the database. Approximate answers were calculated by rewriting the queries so that they could be run on the computed synopses. Also, the system keeps the synopses consistent with the underlying data, as the contents of the original database may have undergone changes due to transactions getting executed and accordingly the contents getting updated.

An efficient range of query processing schemes was proposed [8]. Query processing algorithms were proposed for single-attribute and multiple-attribute range queries respectively within a bounded delay. The ability to answer aggregation queries approximately and efficiently proves to be of larger benefit for data mining and decision support tools. The techniques realize and recognize the importance of taking into consideration the variance in the data distribution [9]. This work may then be implementable on a database system and may turn out to be of the superior quality in the generation of the approximate results.

3. Methodology

3.1 Algorithm

The following is the algorithm of the built system on which it works:

1) Start.
2) The system was provided with the detailed information about the active nodes present in the system as input in the form of a file named ActiveNodes.txt.
3) The peer network was then generated for executing the aggregation user queries.
4) A set of rules formed due to efficient study and analysis is utilized for allowing user for business analysis and constraints.
5) Approximate results were generated along with the final error estimate after completion of the business analysis.
6) Stop.

3.2 Approximate Query Processing System

The system has been designed in such a way that it becomes simpler to keep the information updated regarding the different nodes in network. Then, it is expected at the query node to retrieve and then further analyze the generated results for the query initiated. The details of the current network, different locations of the nodes and the databases at all of the peer nodes in a network, query execution along with the collection of their results, etc. is possible to be maintained. The system is designed for the aggregate queries‟ processing over the different peer nodes within a network. The following diagram depicts the architecture of the system built:

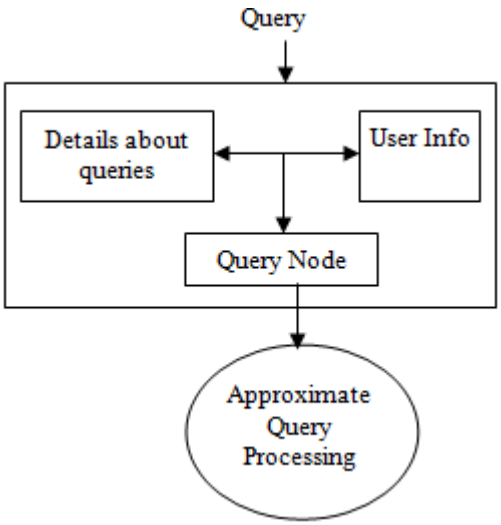

Figure 1: Approximate Query Processing System Architecture

The built system includes connection of the different nodes in a network to form the peer network. As the figure depicts, the aggregation query would be taken from the user at any node for which the results need to be evaluated. The various details regarding the query and the specific user information may also be used for the further query processing. This may be needed as the approximate results of the query are to be further generated. Accordingly, the different peer nodes would be chosen for the task of evaluation of the query. At the beginning, the input is given to the system in the form of a file that covers the information about all the active nodes in the network that can work on the real time databases.

The entire query processing system includes four steps. The first step includes the task of node construction where the peer-to-peer network is actually set up. The second step involves selection of a random node for aggregate queries processing. Different techniques may be used for the selection of a random node. The third step does the selection of records from the database that would be needed for the further processing of the query. Also, the query may be run on multiple peer nodes and the results would accordingly be generated. Then, the fourth and the final step is to do performance evaluation. This involved the comparison of the generated results at peer nodes and then verifying whether the generated results are valid. This is called as approximate processing of aggregate queries. The aim is to increase its efficiency in generating results.

3.3 Experimental Evaluation

For evaluating the system‟s results, firstly the server needs to be started and then the user needs to login. For this purpose SQL Server 2008 is used. Then, for the task of approximate processing of queries in a peer-to-peer network, the no. of nodes may depend upon the application in which the query processing is being performed. The following figure shows the experimental evaluation of the query processing system:

Figure 2: Results of Approximate Query Processing System

In our case, we have made use of four nodes of which one is a server and the other three are clients. The system shows how many nodes connected to it are active. Also, it displays the status of each node whether it is connected in the system or not. Then, when the **Processing Sample** button is clicked, the system asks for the further details from the user regarding query processing. The important step in the query processing work is to select an appropriate aggregate function which is to be used for the task of query processing. Here, separate fields are provided for the selection of table name, attribute field and conditions, if any. Now, a specific table is selected using the „Table‟ field on which the above selected aggregate function is to be performed. Also, the attribute name in the table is to be selected using the **Process Field** on which the aggregate function will be performed. If any condition is to be applied on the attribute field, then the **Condition Field** is used for the same.

After the above procedure is done, then the **Random Walk button** is to be clicked. Now, the process of approximate query processing is done on the table data. The results of the query processing are displayed in another window. The following figure shows the results obtained:

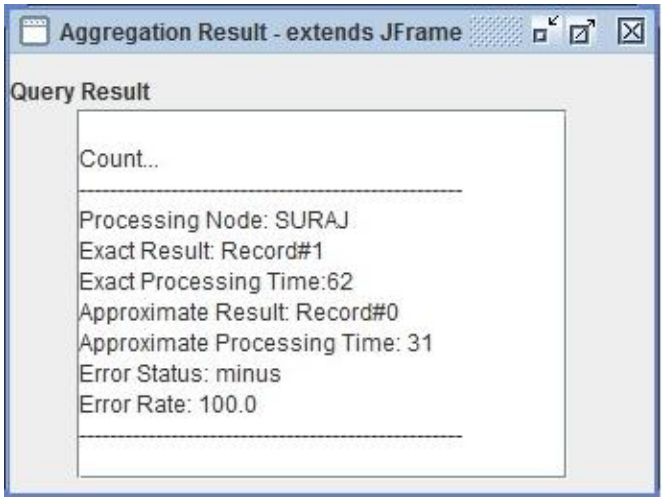

Figure 3: Approximate Query Processing Results

Since the data on which the processing was performed is large, the results could also be shown using a processing chart. The following figure displays the processing chart for the results obtained:

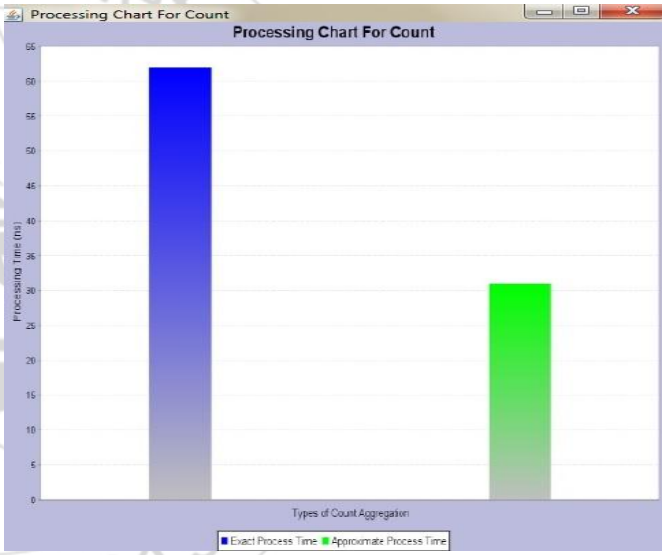

Figure 4: Processing Chart of obtained results

4. Conclusion

The approximate query processing system built makes use of less number of messages to achieve quality service, efficient utilization of resources, increased throughput and less response time. Such a technique might prove to be useful for answering aggregate queries in a P2P network working on real time databases.

References

[1] Amol Bhagat, P. P. Pawade, V. T. Gaikwad, “Efficient Approximate Query Processing in P2P Network”, National Conference on Innovative Paradigms in Engineering & Technology (NCIPET-2012) Proceedings published by International Journal of Computer Applications (IJCA), pp. 25-30, 2012.

[2] Leonidas Galanis, Yuan Wang, Shawn Jeffery, David DeWitt, “Processing Queries in a Large Peer-to-Peer

System", Springer-Verlag Berlin Heidelberg, pp. 273-288, 2003
[3] Benjamin Arai, Gautam Das, Dimitrios Gunopulos, Vana Kalogeraki, "Efficient Approximate Query Processing in Peer-to-Peer Networks", IEEE Transactions on Knowledge and Data Engineering, Vol. 19, No. 7, pp. 919-933, July 2007.
[4] Raddad Al King, Abdelkader Hameurlain, Franck Morvan, "Query Routing and Processing in Peer-to-Peer Data Sharing Systems", International Journal of Database Management Systems (IJDMS), Vol. 2, No. 2, pp. 116-139, May 2010.
[5] Gopal Pandurangan, Prabhakar Raghavan, Eli Upfal, "Building Low-Datameter Peer-to-Peer Networks", IEEE Journal on Selected Areas in Communications, Vol. 21, No. 6, pp. 995-1002, August 2003.
[6] Krishna Kant, Ravi Iyer, Vijay Tewari, "A Framework for Classifying Peer-to-Peer Technologies", Proceedings of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid, May 2002.
[7] Swarup Acharya, Phillip Gibbons, Viswanath Poosala, "Aqua: A Fast Decision Support System Using Approximate Query Answers", Proceedings of the 25th VLDB Conference, Edinburgh, Scotland, 1999.
[8] R. Saravanan, "Processing of Query in Peer to Peer Networks", International Journal of Computer Applications, Vol. 9, No. 6, pp. 12-16, November 2010.
[9] Surajit Chaudhuri, Gautam Das, Vivek Narasayya, "A Robust Optimization-Based Approach for Approximate Answering of Aggregate Queries", ACM SIGMOD 2001, California, USA, 2001.

Author Profile

Suraj N.Arya is a Post Graduation student pursuing M.E in Computer Science and Engineering from Walchand Institute of Technology, Solapur. He received his Bachelor of Technology degree from Shri.Guru Gobind Singhji Institute of Engineering and Technology, Nanded affiliated to Swami Ramanand Teerth Marathwada University in 2010. His research interests lie in the area of Query Processing in peer to peer Networks.

Mr. Rajesh V. Argiddi is an Associate Professor in Computer Science and Engineering Department at Walchand Institute of Technology, Solapur. He received his B.E degree from Shivaji University, Kolhapur and M.E degree from Shivaji University, Kolhapur. He is currently doing his Ph. D from Solapur University, Solapur. His research area lies in Data Mining. Currently he is working for Indian stock market behavior analysis using Data Mining techniques. For this, he has published papers in various renowned journals.