Materialized View Selection using Improvement based Bee Colony Optimization

Biri Arun, School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India

T.V. Vijay Kumar, School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India

ABSTRACT

In the present information age, data and information are vital not just for the survival of any corporate entity, but also to provide it with an edge over its competitors. Data warehouses have become the foundational databases of almost every corporation. However, extracting new information from these data warehouses takes hours, and even days, which is practically unacceptable. Materialized views have been popularly used to facilitate fast information extraction. However, the selection of appropriate views, which significantly accelerate information synthesis is an NP-Complete problem. The aim of this paper is to select near optimal sets of views for materialization using the improvement bee colony optimization algorithm. The experimental results indicate that the improvement bee colony optimization algorithm performs better than the constructive bee colony optimization algorithm and the fundamental view selection algorithm HRUA. The views thus selected would significantly minimize the response time of analytical queries, when materialized, resulting in efficient strategic decision making.

Keywords: Bee Colony Optimization, Data Warehouse, Decision Support System, Materialized View Selection, Swarm Intelligence

1. INTRODUCTION

In the present information age, data and information are vital not only for the survival of any corporate entity, but also for providing it with an edge over its competitors (Boisot and Canals, 2004; Rowley, 2007). In the context of today’s global business and commerce, and innovative management paradigms, data and information have become the key currencies of corporations and governments worldwide. Secure and efficient data management, and fast information synthesis, have become all the more challenging as volumes of digital data are rapidly growing with time (Inmon, 2003). Database management systems (DBMS) were developed to efficiently manage and store the transactional data of various business processes of a department. Different departments of a corporation autonomously employed DBMSs that were purchased from dif-

DOI: 10.4018/IJSSCI.2015100103
fferent independent vendors. As a consequence, the data managed and stored in the departments were consistent and reliable for information synthesis only at the departmental level and not at the corporate level. This was so because, when data of one department was shared with other departments, the former seldom propagated its subsequent updates to those with whom it had shared its data. On the other hand, on account of incompatible DBMSs employed by different departments, it was never easy to efficiently propagate the updates. Such practice greatly compromised the consistency and reliability of data at the corporate level (Inman, 2003). Other factors like incompatible data formats, problems of homonyms and synonyms, heterogeneous operating systems and heterogeneous computer network protocols at departmental level, further aggravated the problem of data integrity at the corporate level. Information synthesis at corporate level from departmental databases became prohibitively expensive in terms of time, money and quality of corporate information; this resulted in an information crisis (Inmon, 2003; Psomas et al., 2002). To address such an information crisis, the data warehouse was developed whereupon data from various departments is collected, and then organized, based on business subjects. Such data is then transformed, cleaned and integrated to achieve consistent and reliable data. They are then loaded into the data warehouse. Every update of the data is time stamped and stored for historical analysis. Reliable corporate information can then be synthesized by processing the appropriate data from the data warehouse (Inmon, 2003). Thus, a data warehouse is the foundational database that provides reliable and consistent data at the corporate level to meet all the informational requirements of a corporation. In order to extract the necessary information from the data warehouse, it has to be analyzed from multiple dimensions. For this reason, a multidimensional model is used in data warehouse to store data (Gray et al., 1996; Shukla et al., 1998; Wremble and Koncilia, 2007). In a relational DBMS, data is stored using the star schema. It consists of many dimension tables connected to a single fact table. Generally, fact tables are massive tables containing millions of rows (Shukla et al., 1998; Shukla et al., 2000; Wremble and Koncilia, 2007). To support timely and effective decision making, data in the data warehouse is analyzed regularly to extract the necessary information; such data analysis, which requires response times of a few seconds, is called On-line Analytical Processing (OLAP) (Inmon, 2003). OLAP queries are not predefined; they are posed in an ad hoc manner by data analysts in response to the information acquired during analysis. To extract new information, analysts pose OLAP queries in an exploratory manner. With the rapid growth in the volume of data in the data warehouse, the response times of OLAP queries increase substantially due to the numerous expensive join operations between massive base tables of the data warehouse; in fact it takes hours and days to process such OLAP queries (Gupta et al., 1997). Such delayed response times of OLAP queries, not only render the data analysis process unsatisfactory, but also makes the obtained information stale and obsolete and, often, not useful for making any real time strategic business decision (Inmon, 2003; Psomas et al., 2002). With the view to improve the OLAP query response time, many techniques like indexing, query optimizer, query evaluation and materialized views have been proposed in literature (Harinarayan et al., 1996). The aim of the paper is to use materialized views to minimize the OLAP query response time.

A subgroup of a single or multiple base tables of a data warehouse forms a view. It is specified by a user query with the purpose of obtaining specific information as required by the data analyst. Usually, it contains millions of rows of data and, therefore, computing views is a very expensive operation. The RDBMS of the data warehouse stores only the OLAP query corresponding to the view and not to the contents of the view. Therefore, the contents of the view have to be recomputed whenever its defining OLAP query is posed. As a consequence, computing views becomes an exorbitantly costly process in terms of CPU time resulting in the response time of OLAP queries suffering much delay (Chirkova and Yang, 2011; Chan et al., 1999). However,
Semantic Matching, Propagation and Transformation for Composition in Component-Based Systems
[www.igi-global.com/article/semantic-matching-propagation-transformation-composition/2784?camid=4v1a](www.igi-global.com/article/semantic-matching-propagation-transformation-composition/2784?camid=4v1a)

Evaluation Model of Cognitive Distraction State Based on Eye Tracking Data Using Neural Networks

Improving the Efficiency of Large-Scale Agent-Based Models Using Compression Techniques
[www.igi-global.com/chapter/improving-efficiency-large-scale-agent/67299?camid=4v1a](www.igi-global.com/chapter/improving-efficiency-large-scale-agent/67299?camid=4v1a)