A Brief Tutorial on Database Queries, Data Mining, and OLAP

Lutz Hamel, University of Rhode Island, USA

INTRODUCTION

Modern, commercially available relational database systems now routinely include a cadre of data retrieval and analysis tools. Here we shed some light on the interrelationships between the most common tools and components included in today's database systems: query language engines, data mining components, and on-line analytical processing (OLAP) tools. We do so by pair-wise juxtaposition which will underscore their differences and highlight their complementary value.

BACKGROUND

Today's commercially available relational database systems now routinely include tools such as SQL database query engines, data mining components, and OLAP (Craig, Vivona, & Bercovitch, 1999; Oracle, 2001; Scalzo, 2003; Seidman, 2001). These tools allow developers to construct high powered business intelligence (BI) applications which are not only able to retrieve records efficiently but also support sophisticated analyses such as customer classification and market segmentation. However, with powerful tools so tightly integrated with the database technology understanding the differences between these tools and their comparative advantages and disadvantages becomes critical for

effective application development. From the practitioner's point of view questions like the following often arise:

- Is running database queries against large tables considered data mining?
- Can data mining and OLAP be considered synonymous?
- Is OLAP simply a way to speed up certain SQL queries?

The issue is being complicated even further by the fact that data analysis tools are often implemented in terms of data retrieval functionality. Consider the data mining models in the Microsoft SQL server which are implemented through extensions to the SQL database query language (e.g. predict join) (Seidman, 2001) or the proposed SQL extensions to enable decision tree classifiers (Sattler & Dunemann, 2001). OLAP cube definition is routinely accomplished via the data definition language (DDL) facilities of SQL by specifying either a star or snowflake schema (Kimball, 1996).

MAIN THRUST OF THE CHAPTER

The following sections contain the pair wise comparisons between the tools and components considered in this chapter.

Database Queries vs. Data Mining

Virtually all modern, commercial database systems are based on the relational model formalized by Codd in the 60s and 70s (Codd, 1970) and the SQL language (Date, 2000) which allows the user to efficiently and effectively manipulate a database. In this model a database table is a representation of a mathematical relation, that is, a set of items that share certain characteristics or attributes. Here, each table column represents an

This query returns a list of all instances in the table where the value of the attribute *Total Spent* is larger than $100. As this example highlights, queries act as filters that allow the user to select instances from a table based on certain attribute values. It does not matter how large or small the database table is, a query will simply return all the instances from a table that satisfy the attribute value constraints given in the query. This straightforward approach to retrieving data from a database has also a drawback. Assume for a moment that our example store is a large store with tens of thousands of customers (perhaps an online store). Firing the above query against the customer table in the database will most likely produce a result set containing a very large number of customers and not much can be learned from this query except for the fact that a large number of customers spent more than $100 at the store. Our innate analytical capabilities are quickly overwhelmed by large volumes of data.

This is where differences between querying a database and mining a database surface. In contrast to a query which simply returns the data that fulfills certain constraints, data mining constructs models of the data in question. The models can be viewed as high level summaries of the underlying data and are in most cases more useful than the raw data, since in a business sense they usually represent understandable and actionable items (Berry & Linoff, 2004). Depending on the questions of interest, data mining models can take on very different forms. They include decision trees and decision rules for classification tasks, association rules for market basket analysis, as well as clustering for market segmentation among many other possible models. Good overviews of current data mining techniques and models can be found in (Berry & Linoff, 2004;

Han & Kamber, 2001; Hand, Mannila, & Smyth, 2001; Hastie, Tibshirani, & Friedman, 2001).

To continue our store example, in contrast to a query, a data mining algorithm that constructs decision rules might return the following set of rules for customers that spent more than $100 from the store database:

```
IF AGE > 35 AND CAR = MINIVAN THEN TOTAL SPENT > $100

                        OR

IF SEX = M AND ZIP = 05566 THEN TOTAL SPENT > $100
```

These rules are understandable because they summarize hundreds, possibly thousands, of records in the customer database and it would be difficult to glean this information off the query result. The rules are also actionable. Consider that the first rule tells the store owner that adults over the age of 35 that own a mini van are likely to spend more than $100. Having access to this information allows the store owner to adjust the inventory to cater to this segment of the population, assuming that this represents a desirable cross-section of the customer base. Similar with the second rule, male customers that reside in a certain ZIP code are likely to spend more than $100. Looking at census information for this particular ZIP code the store owner could again adjust the store inventory to also cater to this population segment presumably increasing the attractiveness of the store and thereby increasing sales.

As we have shown, the fundamental difference between database queries and data mining is the fact that in contrast to queries data mining does not return raw data that satisfies certain constraints, but returns models of the data in question. These models are

From a user perspective it might be interesting to ask some of the following questions:

- How much did sales unit A earn in January?
- How much did sales unit B earn in February?
- What was their combined sales amount for the first quarter?

Even though it is possible to extract this information with standard SQL queries from our database, the normalized nature of the database makes the formulation of the appropriate SQL queries very difficult. Furthermore, the query process is likely to be slow due to the fact that it must perform complex joins and multiple scans of entire database tables in order to compute the desired aggregates.

By rearranging the database tables in a slightly different manner and using a process called pre-aggregation or *computing cubes* the above questions can be answered with much less computational power enabling a real time analysis of aggregate attribute values – OLAP (Craig et al., 1999; Kimball, 1996; Scalzo, 2003). In order to enable OLAP, the database tables are usually arranged into a star schema where the inner-most table is called the fact table and the outer tables are called dimension tables. Figure 3 shows a star schema representation of our store organized along the main dimensions of the store business: customers, sales units, products, and time.

real time using specialized OLAP operations. In a larger context we can view OLAP as a methodology for the organization of databases along the dimensions of a business making the database more comprehensible to the end user.

Data Mining vs. OLAP

Is OLAP data mining? As we have seen, OLAP is enabled by a change to the data definition of a relational database in such a way that it allows for the pre-computation of certain query results. OLAP itself is a way to look at these pre-aggregated query results in real time. However, OLAP itself is still simply a way to evaluate queries which is different from building models of the data as in data mining. Therefore, from a technical point of view we cannot consider OLAP to be data mining. Where data mining tools model data and return actionable rules, OLAP allows users to compare and contrast measures along business dimensions in real time.

It is interesting to note, that recently a tight integration of data mining and OLAP has occurred. For example, Microsoft SQL Server 2000 not only allows OLAP tools to access the data cubes but also enables its data mining tools to mine data cubes (Seidman, 2001).

FUTURE TRENDS

Perhaps the most important trend in the area of data mining and relational databases is the liberation of data mining tools from the “single table requirement”. This new breed of data mining algorithms is able to take advantage of the full relational structure of a relational database obviating the need of constructing a single table that