Chapter IX

Classification on Top of Data Cube

Lixin Fu
University of North Carolina, Greensboro, USA

ABSTRACT
Currently, data classification is either performed on data stored in relational databases or performed on data stored in flat files. The problem with these approaches is that for large data sets, they often need multiple scans of the original data and thus are often infeasible in many applications. In this chapter we propose to deploy classification on top of OLAP (online analytical processing) and data cube systems. First, we compute the statistics in various combinations of the attributes known as data cubes. The statistics are then used to derive classification models. In this way, we only scan the original data once, which improves the performance of classification significantly. Furthermore, our new classifier will provide “free” classification by eliminating the dominating I/O overhead of scanning the massive original data. An architecture that integrates database, data cube, and data mining is given and three new cube-based classifiers are presented and evaluated.

INTRODUCTION
Data classification is the process of building a model from available data called the training data set and classifying objects according to their attributes. Classification is a well-studied important problem (Han & Kamber, 2001). It has many applications. For example, it has been used in online classification of articles from registered newsgroups on the Internet into predefined subject categories, in the insurance industry, for tax and
credit card fraud detection, for medical diagnosis, and so forth. Such data classification is one of the important topics in Web mining and Web services.

Currently, data classification is either performed on data stored in relational database management systems or performed on data stored in small flat files. The problem of existing classifiers built on these approaches is that for large data sets, they often need multiple scans of the original data and thus are infeasible in many real applications.

In this chapter we propose to deploy classification on top of OLAP (online analytical processing) and data cube systems. First, a multidimensional analysis is conducted on these large data sets. The output of this analysis is summarized data; for example, aggregates in various combinations of the attributes also known as data cubes (Gray et al., 1997). The aggregates are then used to derive classification models. In this way, we only scan the original data once, which improves the performance of classification significantly. Furthermore, since in the decision support systems data cubes are usually already precomputed (in terms of materialized views for example) for answering OLAP queries, our new classifier will provide “free” classification functions by eliminating the dominating I/O overhead of scanning the original data.

Our objectives in this chapter are:
- designing new classifiers built on data cubes and,
- proposing an architecture that takes the advantages of above new algorithms and integrates DBMS, OLAP systems, and data mining systems seamlessly.

The remainder of the chapter is organized as follows. First we give a brief summary of the related work. In the next two sections, statistics tree structures and related data cube computation algorithms are described as the foundation of later sections, and an architecture that integrates DBMS, OLAP, and data mining functions is proposed. After that, we present three new classifiers based on the data cube: pattern detection, cube-based naïve Bayesian classification, and cube-based decision tree classification. Lastly, we give the evaluation of these three algorithms, summarize the chapter, and discuss our future work directions related to classification.

**BACKGROUND**

Many popular classification algorithms are based on decision tree induction. Algorithm ID-3 generates a simple tree in a top-down fashion (Quilan, 1986). It chooses the attribute with the highest information gain as the split attribute. Data are partitioned into subsets recursively until the partitions contain samples of the same classes. Algorithm C4.5 extends the domain of classification in ID-3 from categorical domain to numerical domain (Quilan, 1993). It gives approaches to transform decision trees into rules. For continuous attribute A, the values are sorted and the midpoint \( \nu \) between two values is considered as a possible split. The split form is \( A \leq \nu \). There are \( V-1 \) such splits if \( A \) has \( V \) values in its domain. For a categorical attribute, if its cardinality is small, all subsets of its domain can be a candidate split; otherwise, we can use a greedy strategy to create candidate splits.

Recent decision-tree classifiers focus on scalability issues for large data sets. SLIQ (Supervised Learning In Quest) uses Gini index as the classification function (Mehta, Agrawal & Rissanen, 1996). Presorting (for numerical attributes) and breadth-first
Related Content

Evaluating Social Interaction and Support Methods over Time
www.igi-global.com/chapter/evaluating-social-interaction-and-support-methods-over-time/137398?camid=4v1a

Image Mining: A Case for Clustering Shoe Prints
www.igi-global.com/chapter/image-mining-case-clustering-shoe/40422?camid=4v1a

A New Approach of Web Systems Modularity Increase Using Combination of Event-Driven Software Architecture and Relationship Mechanism Based on Message Passing: Case Study
www.igi-global.com/article/a-new-approach-of-web-systems-modularity-increase-using-combination-of-event-driven-software-architecture-and-relationship-mechanism-based-on-message-passing/123181?camid=4v1a
Information Acquisition and Recall in Location-Aware and Search Engine Retrieval Systems
www.igi-global.com/article/information-acquisition-recall-location-aware/44921?camid=4v1a