Mining Generalized Association Rules in an Evolving Environment

Wen-Yang Lin
National University of Kaohsiung, Taiwan

Ming-Cheng Tseng
Institute of Information Engineering, Taiwan

INTRODUCTION

The mining of Generalized Association Rules (GARs) from a large transactional database in the presence of item taxonomy has been recognized as an important model for data mining. Most previous studies on mining generalized association rules, however, were conducted on the assumption of a static environment, i.e., static data source and static item taxonomy, disregarding the fact that the taxonomy might be updated as new transactions are added into the database over time, and as such, the analysts may have to continuously change the support and confidence constraints, or to adjust the taxonomies from different viewpoints to discover more informative rules. In this chapter, we consider the problem of mining generalized association rules in such a dynamic environment. We survey different strategies incorporating state-of-the-art techniques for dealing with this problem and investigate how to efficiently update the discovered association rules when there are transaction updates to the database along with item taxonomy evolution and refinement of support constraint.

BACKGROUND

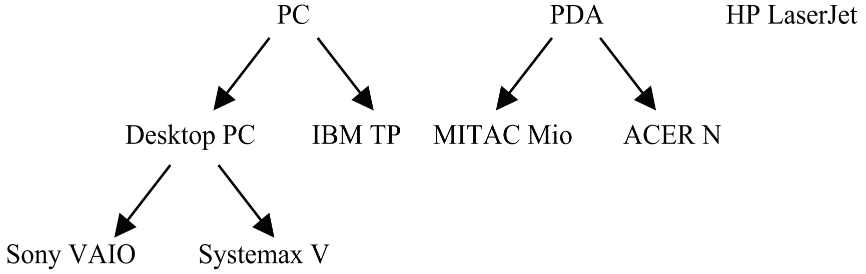
An association rule is an expression of the form $X \Rightarrow Y$, where X and Y are sets of items. Such a rule reveals that transactions in the database containing items in X tend to also contain items in Y, and the probability, measured as the fraction of transactions containing X that also contain Y, is called the *confidence* of the rule. The *support* of the rule is the fraction of the transactions that contain all items in both X and Y. The problem of mining association rules is to discover all association rules that satisfy support and confidence constraints.

In many applications, there are explicit or implicit taxonomies over the items, so it may be more useful to find associations at different taxonomic levels than only at the primitive concept level. For example, consider the taxonomy of items in Figure 1. It is likely that the association rule,

Systemax V $\Rightarrow$ HP LaserJet ($sup \underset{\cdot}{=} 20\%$, $conf$=100%) does not hold when the minimum support is set to 25%, but the following association rule may be valid,

Desktop PC $\Rightarrow$ HP LaserJet

Figure 1. Example of taxonomy



This kind of association rule with taxonomy is also called the *generalized association rule* (Srikant & Agrawal, 1995) or the *multi-level association rule* (Han & Fu, 1995). The work in Srikant and Agrawal (1995) aimed at finding associations among items at any level of the taxonomy, whereas the objective in Han and Fu (1995) was to discover associations level-by-level in a fixed hierarchy, i.e., only associations among items on the same level were examined progressively from the top level to the bottom.

Mining GARs with Transaction Update

In the real world, however, data mining practitioners are usually confronted with a dynamic environment. The first challenge comes from the fact that the source database is not static. New transactions are continually added into the database over time, outdated transactions are occasionally or periodically purged from the repository, and some transactions might be modified, thus leading to the essence of updating discovered association rules when transaction updates occur in the database over time. Cheung et al. (1996a) first addressed this problem and proposed an algorithm called FUP (Fast UPdate). They further extended the model to incorporate the situations of deletion and modification (Cheung et al., 1997). Subsequently, a number of techniques have been proposed to improve the efficiency of incremental mining algorithms, such as *negative border* (Sarda & Srinivas, 1998; Thomas et al., 1997), *dynamic counting* (Ng & Lam, 2000), *pre-large itemsets* (Hong et al., 2001), *sliding window filter* (Lee et al., 2005), *support prediction* (Guirguis et al., 2006), and *FP-like tree structure* (Ezeife & Su, 2002; Leung et al., 2007), although all of these were confined to mining associations among primitive items.

The maintenance issue for generalized association rules was also first studied by Cheung et al. (1996b), who proposed an extension of their FUP algorithm, called MLUp, to accomplish the task. Hong et al. (2004) then extended Han and Fu's approach (Han & Fu, 1995) by introducing the concept of pre-large itemsets (Hong et al., 2000) to postpone the original database rescanning until a number of records have been modified. In (Tseng & Lin, 2004), Tseng and Lin extended the problem to incorporate non-uniform minimum support.

Mining GARs with Interestingness Refinement

The second challenge arises from users' perception of information needs. Faced with an unknown and large volume of data collected under a highly competitive environment, analysts generally lack of knowledge about the application domains and so have to change their viewpoints continuously, in an interactive way to find informative rules. In the context of association mining, the user's viewpoint, in its simplest form, is specified through the rule's thresholds, e.g., minimum support and minimum confidence. In the past decade, various strategies have been proposed to realize the interactive (or online) association mining, including *precomputation* (Han, 1998; Aggarwal & Yu, 2001; Czejdo et al., 2002; Duan et al., 2006; Liu et al., 2007), *caching* (Nag et al., 1999), and *incremental update* (Hidber, 1999; Liu & Yin, 2001; Ma et al., 2002; Deng et al., 2005).

The general idea for precomputation strategy is to precompute all frequent itemsets relative to a presetting support threshold, and once the specified support threshold is larger than the presetting value, the qualified association rules can be immediately generated without the burden of an expensive phase for itemset generation. With a similar philosophy, the caching strategy tries to eliminate the cost spent on frequent itemsets computation by temporarily storing previously discovered frequent itemsets (may be accompanied with some infrequent itemsets) that are beneficial to subsequent association queries. The effectiveness of this strategy relies primarily on a successful design of the cache replacement algorithm. The third strategy can be regarded as a compensation for the first two strategies. During the course of a sequence of remining trials with varied support thresholds, the incremental update strategy endeavors to utilize the discovered frequent itemsets in previous trial to reduce the cost for subsequent re-execution.

Mining GARs with Taxonomy Evolution

The third challenge comes to the evolution of item taxonomy. As a representation of the classification relationship imposed on items, a taxonomy must evolve to reflect what has occurred to the domain applications. For example, items corresponding to new products must be added into the taxonomy, and their insertion