Association Rule Mining of Relational Data

Anne Denton
North Dakota State University, USA

Christopher Besemann
North Dakota State University, USA

INTRODUCTION

Most data of practical relevance are structured in more complex ways than is assumed in traditional data mining algorithms, which are based on a single table. The concept of relations allows for discussing many data structures such as trees and graphs. Relational data have much generality and are of significant importance, as demonstrated by the ubiquity of relational database management systems. It is, therefore, not surprising that popular data mining techniques, such as association rule mining, have been generalized to relational data. An important aspect of the generalization process is the identification of challenges that are new to the generalized setting.

BACKGROUND

Several areas of databases and data mining contribute to advances in association rule mining of relational data.

- **Relational data model:** Underlies most commercial database technology and also provides a strong mathematical framework for the manipulation of complex data. Relational algebra provides a natural starting point for generalizations of data mining techniques to complex data types.
- **Inductive Logic Programming, ILP (Džeroski & Lavrač, 2001, pp. 48-73):** Treats multiple tables and patterns as logic programs. Hypothesis for generalizing data to unseen examples are solved using first-order logic. Background knowledge is incorporated directly as a program.
- **Association Rule Mining, ARM (Agrawal & Srikant, 1994):** Identifies associations and correlations in large databases. The result of an ARM algorithm is a set of association rules in the form $A \rightarrow C$. There are efficient algorithms such as Apriori that limit the output to sets of items that occur more frequently than a given threshold.
- **Graph Theory:** Addresses networks that consist of nodes that are connected by edges. Traditional graph theoretic problems typically assume no more than one property per node or edge. Solutions to graph-based problems take into account graph and subgraph isomorphism. For example, a subgraph should only count once per isomorphic instance. Data associated with nodes and edges can be modeled within the relational algebra framework.
- **Link-based Mining (Getoor & Diehl, 2005):** Addresses data containing sets of linked objects. The links are exploited in tasks such as object ranking, classification, and link prediction. This work considers multiple relations in order to represent links.

Association rule mining of relational data incorporates important aspects of these areas to form an innovative data mining area of important practical relevance.

MAIN THRUST OF THE CHAPTER

Association rule mining of relational data is a topic that borders on many distinct topics, each with its own opportunities and limitations. Traditional association rule mining allows extracting rules from large data sets without specification of a consequent. Traditional predictive modeling techniques lack this generality and only address a single class label. Association rule mining techniques can be efficient because of the pruning opportunity provided by the downward closure property of support, and through the simple structure of the resulting rules (Agrawal & Srikant, 1994).

When applying association rule mining to relational data, these concepts cannot easily be transferred. This

can be seen particularly easily for data with an underlying graph structure. Graph theory has been developed for the special case of relational data that represent connectivity between nodes or objects with no more than one label. A commonly studied pattern mining problem in graph theory is frequent subgraph discovery (Kuramochi & Karypis, 2004). Challenges in gaining efficiency differ substantially in frequent subgraph discovery compared with data mining of single tables: While downward closure is easy to achieve in single-table data, it requires advanced edge disjoint mining techniques in graph data. On the other hand, while the subgraph isomorphism problem has simple solutions in a graph setting, it cannot easily be discussed in the context of relational joined tables.

This chapter attempts to view the problem of relational association rule mining from the perspective of these and other data mining areas, and highlights challenges and solutions in each case.

General Concept

Two main challenges have to be addressed when applying association rule mining to relational data. Combined mining of multiple tables leads to a search space that is typically large even for moderately sized tables. Performance is, thereby, commonly an important issue in relational data mining algorithms. A less obvious problem lies in the skewing of results (Jensen & Neville, 2007, Getoor & Diehl, 2005). Unlike single-table data, relational data records cannot be assumed to be independent.

One approach to relational data mining is to convert the data from a multiple table format to a single table format using methods such as relational joins and aggregation queries. The relational join operation combines each record from one table with each occurrence of the corresponding record in a second table. That means that the information in one record is represented multiple times in the joined table. Data mining algorithms that operate either explicitly or implicitly on joined tables, thereby, use the same information multiple times. This also applies to algorithms in which tables are joined on-the-fly by identifying corresponding records as they are needed. The relational learning task of transforming multiple relations into propositional or single-table format is also called propositionalization (Kramer et al., 2001). We illustrate specific issues related to reflexive relationships in the next section on relations that represent a graph.

A variety of techniques have been developed for data mining of relational data (Džeroski & Lavrač, 2001). A typical approach is called inductive logic programming, ILP. In this approach relational structure is represented in the form of Prolog queries, leaving maximum flexibility to the user. ILP notation differs from the relational algebra notation; however, all relational operators can be represented in ILP. The approach thereby does not limit the types of problems that can be addressed. It should, however, also be noted that relational database management systems are developed with performance in mind and Prolog-based environments may present limitations in speed.

Application of ARM within the ILP setting corresponds to a search for frequent Prolog (Datalog) queries as a generalization of traditional association rules (Dehaspe & Toivonen, 1999). An example of association rule mining of relational data using ILP (Dehaspe & Toivonen, 2001) could be shopping behavior of customers where relationships between customers are included in the reasoning as in the rule:

{customer(X), parent(X,Y)}→{buys(Y, cola)},

which states that if *X* is a parent then their child *Y* will buy a *cola*. This rule covers tables for the parent, buys, and customer relationships. When a pattern or rule is defined over multiple tables, a relational key is defined as the unit to which queries must be rolled up (usually using the Boolean existential function). In the customer relationships example a key could be "customer", so support is based on the number of customers that support the rule. Summarizations such as this are also needed in link-based classification tasks since individuals are often considered the unknown input examples (Getoor & Diehl, 2005). Propositionalization methods construct features by traversing the relational link structure. Typically, the algorithm specifies how to place the constructed attribute into a single table through the use of aggregation or "roll-up" functions (Kramer et al., 2001). In general, any relationship of a many-to-many type will require the use of aggregation when considering individual objects since an example of a pattern can extend to arbitrarily many examples of a larger pattern. While ILP does not use a relational joining step as such, it does also associate individual objects with multiple occurrences of corresponding