Chapter III
Using Mined Patterns for XML Query Answering

Elena Baralis
Dip. di Automatica e Informatica, Politecnico di Torino, Italy

Paolo Garza
Dip. di Automatica e Informatica, Politecnico di Torino, Italy

Elisa Quintarelli
Dip. di Electronic e Informazione, Politecnico di Milano, Italy

Letizia Tanca
Dip. di Electronic e Informazione, Politecnico di Milano, Italy

ABSTRACT

XML is a rather verbose representation of semistructured data, which may require huge amounts of storage space. Several summarized representations of XML data have been proposed, which can both provide succinct information and be directly queried. In this chapter, we focus on compact representations based on the extraction of association rules from XML datasets. In particular, we show how patterns can be exploited to (possibly partially) answer queries, either when fast (and approximate) answers are required, or when the actual dataset is not available; for example, it is currently unreachable. We focus on (a) schema patterns, representing exact or approximate dataset constraints, (b) instance patterns, which represent actual data summaries, and their use for answering queries.

INTRODUCTION

The extensible markup language (XML) (World Wide Web Consortium, 1998) was initially proposed as a standard way to represent, exchange, and publish information on the Web, but its usage has recently spread to many other application fields. To name but a few, XML is currently used for publishing legacy data, for storing data that cannot be represented with traditional data models, and for ensuring interoperability among software systems.

However, XML is a rather verbose representation of data, which may require huge amounts of storage space. We propose several summarized representations of XML data, which can both provide succinct information and be directly queried. In particular, we propose *patterns* as abstract representations of the (exact or approximate) constraints that hold the data, and their use for (possibly partially) answering queries, either when fast, though approximate, answers are required, or when the actual dataset is not available; for example, it is currently unreachable. In this last case, the service of a "semantic" proxy, which caches patterns instead of actual data pages, can be provided.

In this chapter, we focus on (a) *schema patterns*, representing exactly or approximately the dataset constraints, and (b) *instance patterns*, which represent, again exactly or approximately, actual data summaries. Our summarized representations are based on the extraction of association rules from XML datasets, and queried by means of the GSL graphical query language (Damiani, Oliboni, Quintarelli, & Tanca, 2003).

Patterns can be exploited to provide intensional query answering. An intensional answer to a query substitutes the actual data answering the query (the extensional answer) with a set of properties characterizing them (Motro, 1989). Thus, our intensional answers are in general more synthetic than the extensional ones, but usually approximate. Applications of intensional query answering become more and more useful as the technology offers improved means for information handling; query optimization in large datasets, decision support, and context based data summarization are only the most important.

Approximate intensional answers may replace the extensional ones whenever a short response time is required, even to the cost of a controlled lack of precision. Furthermore, decision support may take advantage of the inherently synthetic nature of intensional answers. Consider, for example, the query: *What are the papers written by John Doe*? While an extensional answer, listing all the papers, is in order in case further transactional processing is required, an answer like *80% of John Doe's papers are about Data Mining* may be more interesting if a subsequent decision has to be taken, as, for instance, the choice of John Doe as a conference PC member.

Another interesting application domain is the storage and query of patterns instead of data in context-based data personalization for mobile users. Here, data summarization and tailoring are needed because of two main reasons: (a) the need to keep information manageable, in order for the user not to be confused by too much noise, and (b) the frequent case that the mobile device be a small one, like a palm computer or a cellular phone, in which condition only a summary of the information may be kept on board. In this case, patterns are kept on the mobile device instead of the actual data, and context-awareness can be enforced by keeping on board only the patterns which are relevant to the current situation. Finally, extracted patterns may also be used to provide an integrated representation of information mined from different XML documents, in order to answer queries by using all the available information gathered from different (heterogeneous) data sources.

The chapter is organized as follows. In the next section, the background is discussed. The Mined Patterns: A New Approach to XML Intensional Query Answering section introduces the type of patterns we propose and describes how we represent them in our graph-based language. In the Using Patterns to Answer Queries subsection, we propose an approach to provide intensional answers to user queries. The Representing and Querying Instance Patterns section discusses how patterns are physically represented and queries actually performed in this representation. The