Chapter V
Modeling and Managing Heterogeneous Patterns: The PSYCHO Experience

Anna Maddalena
University of Genoa, Italy

Barbara Catania
University of Genoa, Italy

ABSTRACT

Patterns can be defined as concise, but rich in semantics, representations of data. Due to pattern characteristics, ad-hoc systems are required for pattern management, in order to deal with them in an efficient and effective way. Several approaches have been proposed, both by scientific and industrial communities, to cope with pattern management problems. Unfortunately, most of them deal with few types of patterns and mainly concern extraction issues. Little effort has been posed in defining an overall framework dedicated to the management of different types of patterns, possibly user-defined, in a homogeneous way. In this chapter, we present PSYCHO (Pattern based SYstem arCHitecture prOtotype), a system prototype providing an integrated environment for generating, representing, and manipulating heterogeneous patterns, possibly user-defined. After presenting the PSYCHO logical model and architecture, we will focus on several examples of its usage concerning common market basket analysis patterns, that is, association rules and clusters.

INTRODUCTION

The large volume of heterogeneous raw data collected from various sources in real-world application environments usually does not constitute knowledge by itself for the end users. Indeed, little information can be deduced by simply observing such a huge quantity of data, and advanced knowl-

edge management techniques are required to extract from them concise and relevant information that can help human users to drive and specialize business decision processes. Of course, since raw data may be very heterogeneous, different kinds of knowledge artifacts, representing knowledge hidden into raw data, can be extracted.

We use the generic term *patterns* to denote in a concise and general way such compact but rich in semantics knowledge artifacts. Patterns reduce the number and size of data, to make them manageable from humans while preserving as much as possible their hidden information or discovering new interesting correlations.

Pattern management is an important issue in many different contexts and domains. However, without doubt, the most relevant context in which pattern management is required is data mining. Clusters, frequent itemsets, and association rules are some examples of common data mining patterns. The trajectory of a moving object in a localizer control system or the keyword frequency in a text document represent other examples of patterns.

Since patterns can be generated from different application contexts, their structure can be highly heterogeneous. Moreover, patterns can be extracted from raw data by applying some data mining tools (*a-posteriori patterns*) but also known by the users and used, for example, to check how well some data source is represented by them (*a-priori patterns*). In addition, it is important to determine whether existing patterns, after a certain time, still represent the data source they are associated with, possibly being able to change pattern information when the quality of the representation changes. Finally, independently from their type, all patterns should be manipulated (e.g., extracted, synchronized, deleted) and queried through ad hoc languages. Those specific characteristics make traditional database management systems (DBMSs) unsuitable for pattern representation and management. Therefore, the need arises for the design of ad hoc *pattern management systems (PBMSs)*, that is, systems for handling (storing/processing/retrieving) patterns defined over raw data (PANDA, 2001).

Many efforts have been devoted towards this issue. Scientific community efforts are mainly devoted to develop frameworks providing a full support for heterogeneous pattern management. The 3W Model (Johnson, Lakshmanan, & Ng, 2000) and the PANDA framework (PANDA, 2001) are examples of such approaches, in which raw data are stored and managed in a traditional way by using, for example, a DBMS whereas patterns are stored and managed by a dedicated PBMS. On the other hand, under the inductive databases approach, mainly investigated in the context of the CINQ project (CINQ, 2001), raw data and patterns are stored by using the same data model and managed in the same way by the same system. Industrial proposals mainly deal with standard representation purposes for patterns resulting from data mining, in order to support their exchange between different platforms. Examples of such approaches are the predictive model markup language (PMML, 2003), an XML-based language for common data mining representation, and the Java data mining aPI (JDM, 2003), a Java API for pattern management. In both cases, no user-defined patterns can be specified, and manipulation functionalities are quite limited. Finally, in the commercial world, the most important DBMSs address the pattern management problem by offering features for representing and managing typical data mining patterns.

In general, existing proposals do not provide a unified framework dealing with heterogeneous patterns in a homogeneous way. Indeed, usually they cope with some predefined pattern types, and they do not provide advanced capabilities for pattern extraction, querying, and management.

Starting from the limitations of existing proposals and taking into account the results presented in the context of the PANDA project (Bertino, Catania, & Maddalena, 2004; Catania, Maddalena, Mazza, Bertino, & Rizzi, 2004; Rizzi,