Building and Exploiting Ad Hoc Concept Hierarchies for Web Log Analysis

Carsten Pohle and Myra Spiliopoulou

Department of E-Business, Leipzig Graduate School of Management
{cpohle,myra}@ebusiness.hhl.de

Abstract. Web usage mining aims at the discovery of interesting usage patterns from Web server log files. "Interestingness" relates to the business goals of the site owner. However, business goals refer to business objects rather than the page hits and script invocations recorded by the site server. Hence, Web usage analysis requires a preparatory mechanism that incorporates the business goals, the concepts reflecting them and the expert's background knowledge on them into the mining process. To this purpose, we present a methodology and a mechanism for the establishment and exploitation of application-oriented concept hierarchies in Web usage analysis. We demonstrate our approach on a real data set and show how it can substantially improve both the search for interesting patterns by the mining algorithm and the interpretation of the mining results by the analyst.

Keywords: Concept hierarchies, taxonomy construction, pre-mining, data preparation, association rules' discovery, pattern matching, data mining

1 Introduction

The success of data mining projects depends heavily on the analyst's ability to translate a domain expert's problem statement into the semantics of the data to be mined. The complexity of this problem depends on the application domain and the data sources.

For example, when mining a retail store's transaction data for items frequently bought together, the store manager's notion of *item* or *product* can usually be directly transferred to the analyzed data, because products are represented by their respective product codes. Market basket analysis would then return rules like `ProdA`→`ProdB`, denoting that buyers of product A are likely to purchase product B, too. The data analyst's task becomes more complicated when the management is not interested in association patterns at the product level but in generic rules on product categories, like `SkimmedMilk`→`OatmealCereals`. If these product categories differ from those depicted in the company's warehouse, as is the case when new business opportunities are sought for, then the new categories must be established during data preparation.

The mapping of business concepts upon raw data is even more acute in web usage mining applications. Web servers record invocations of URLs, scripts, images and frames, while Web site owners are interested in patterns like "Users inspect three to five products before adding something to the shopping cart". As in the market basket analysis example, the analyst must formulate concept hierarchies in order to map object

Y. Kambayashi, W. Winiwarter, M. Arikawa (Eds.): DaWaK 2002, LNCS 2454, pp. 83–93, 2002.

invocations into business concepts. A *concept hierarchy* or *taxonomy* defines a sequence of mappings from detailed concepts into higher level abstractions.

Concept hierarchies are well-known from the research on data warehousing [3]: The corporate data warehouse is associated with multiple hierarchical views over the data, conceptualized as the dimensions of an OLAP cube. However, data mining often requires the exploitation of *ad hoc* taxonomies, which, in contrast to the rather static OLAP taxonomies, are tailored to the specific business problem specification for the data mining process. For example, a new marketing campaign of a large store may require that the products are observed in a different context than the general-purpose product classification available in the company warehouse.

The establishment of concept hierarchies is a human-centric task. Although text mining can be used to extract representative concepts from web pages, the specification of the concepts of interest for the analysis can only be done by the analyst. In this study, we do not attempt to delegate the analyst's work to the software, but rather propose a methodology for the establishment of application-oriented ad hoc concept hierarchies and provide tools that support the steps of this methodology.

We first provide a formal definition of concept hierarchies, introduce our Web usage environment WUM and briefly discuss related work. In section 3, we discuss the notion of *application-oriented* concept hierarchies, describe our methodology to establish them and explain how the data preparation module of WUM supports the exploitation of taxonomies by the miner. Section 4 contains a case study, in which we applied concept hierarchies for the behavioural analysis of the visitors of a Web site. The last section summarizes our results and provides an outlook of future work.

2 Concept Hierarchies

In OLAP applications, each dimension of a data warehouse's cube is statically predefined by a hierarchy of concepts. "Roll-up" operations fold the data into a more abstract level, while "drill-down" operations unfold them into more detailed concepts. Concept hierarchies are subject to formal constraints, intended to prevent the establishment of ill-formed taxonomies. Mutual exclusiveness of the concepts at the same level of the hierarchy is one such constraint; completeness is another.

2.1 Fundamentals of Concept Hierarchies

Our understanding of concepts and concept hierarchies goes back to the formal concept analysis developed by Wille et al. [7]. The authors define a formal context $(\mathcal{O}, \mathcal{A}, \mathcal{R})$ as two sets $\mathcal{O}$ and $\mathcal{A}$ and a relation $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{A}$ between them. The elements of $\mathcal{O}$ are called *objects*, those of $\mathcal{A}$ are *attributes*. For each set of objects $O \subseteq \mathcal{O}$, $O' := \{a \in \mathcal{A} | (o, a) \in \mathcal{R} \forall o \in O\}$ denotes the set of attributes common to the objects in O. Accordingly, the set $A' := \{o \in O | (o, a) \in \mathcal{R} \forall a \in A\}$ is the set of objects sharing the attributes for each $A \subseteq \mathcal{A}$.

A *(formal) concept* of the context $(\mathcal{O}, \mathcal{A}, \mathcal{R})$ is a pair (O, A) with $O \subseteq \mathcal{O}$, $A \subseteq \mathcal{A}$, $O' = A$ and $A' = O$. The set O is called the concept's *extension*, the set A is