Chapter 16
Exploring Calendar-Based Pattern Mining in Data Streams

Rodrigo Salvador Monteiro
COPPE / UFRJ – Brazil

Geraldo Zimbrão
COPPE / UFRJ – Brazil

Holger Schwarz
IPVS - University of Stuttgart – Germany

Bernhard Mitschang
IPVS - University of Stuttgart – Germany

Jano Moreira de Souza
COPPE / UFRJ – Brazil

ABSTRACT

Calendar-based pattern mining aims at identifying patterns on specific calendar partitions. Potential calendar partitions are for example: every Monday, every first working day of each month, every holiday. Providing flexible mining capabilities for calendar-based partitions is especially challenging in a data stream scenario. The calendar partitions of interest are not known a priori and at each point in time only a subset of the detailed data is available. The authors show how a data warehouse approach can be applied to this problem. The data warehouse that keeps track of frequent itemsets holding on different partitions of the original stream has low storage requirements. Nevertheless, it allows to derive sets of patterns that are complete and precise. Furthermore, the authors demonstrate the effectiveness of their approach by a series of experiments.

INTRODUCTION

Calendar-based schemas (Li, Y. et al., 2001) (Ramaswamy, S. et al., 1998) were proposed as a semantically rich representation of time intervals and used to mine temporal association rules. An example of a calendar schema is (year, month, day, day_period), which defines a set of calendar patterns, such as every morning of January of 1999 (1999, January, *, morning) or every 16th day of January...
of every year (*, January, 16, *). In the research field of data mining, frequent itemsets derived from transactional data represent a particularly important pattern domain due to their large applicability (Boulicaut, J., 2004). Association rule mining is the most recognized application of frequent itemsets (Agrawal, R. et al., 1993). Other examples are generalized rule mining (Mannila, H., & Toivonen, H., 1996) and associative classification (Liu, B. et al., 1998). The combination of the rich semantics of calendar-based schemas with frequent itemset mining, namely calendar-based frequent itemset mining, corresponds to the first step of various calendar-based pattern mining tasks, e.g., calendar-based association rules. An example of calendar-based association rules provided in Li, Y. et al. (2001) is that eggs and coffee are frequently sold together in morning hours. Considering the transactions at the all-day granule would probably not reveal such a rule and its implicit knowledge.

Recent applications, such as network traffic analysis, web click stream mining, power consumption measurement, sensor network data analysis, and dynamic tracing of stock fluctuation are some examples where a new kind of data arises, the so called data stream. A data stream is continuous and potentially infinite. Mining calendar-based patterns in data streams is a difficult task described in the following statement:

Problem Statement: Let D be a transactional dataset provided by a data stream. Let X be a set of ad-hoc calendar-based constraints and T the subset of transactions from D satisfying X. The frequency of an itemset I over T is the number of transactions in T in which I occurs. The support of I is the frequency divided by the total number of transactions in T. Given a minimum support \( \sigma \), the set of calendar-based frequent itemsets is defined by the itemsets with support \( \geq \sigma \) over the set of transactions T.

Some examples of calendar-based constraints are: weekday in {Monday, Friday}; day_period = “Morning”; holiday = “yes”; etc. The calendar partitions that will reveal interesting temporal patterns are not known a priori and at each point in time only a subset of the detailed data is available in a window based on the most recent data.

Existing approaches cannot solve the above problem because either they require all transactions to be available during the calendar-based mining task or they do not provide enough flexibility to consider a calendar-based-subset of the data stream transactions. In order to flexibly derive patterns based on calendar-based-subset of the data stream transactions, we need some kind of summary for previous time windows. As the calendar partitions that will be interesting for analysis are not known in advance, it is not obvious how to build and store such a summary.

CONTRIBUTIONS

The focus of DWFIST (Monteiro, R. S. et al., 2005) is on presenting the model of a data warehouse that keeps track of frequent itemsets holding on disjoint sets of the original transactions. We call each of these disjoint sets a partition. For each partition, the data warehouse stores all itemsets that are frequent in this partition.

In this chapter, we discuss how to apply DWFIST in a data stream scenario leveraging calendar-based pattern mining capabilities. One partition represents a period of one hour for example. The partitions may be freely combined in order to retrieve the frequent itemsets holding on any set of partitions. A temporal dimension on the data warehouse represents the calendar features, such as year, month, holiday, weekday, and many others. The main contributions of this chapter are as follows:

- Explain the structure of a data warehouse that supports calendar-based frequent itemset mining.
- Show that we can guarantee the completeness of the retrieved frequent itemsets and