Engineering Applications of Artificial Intelligence 30 (2014) 225–234

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

A two-phase approach for mining weighted partial periodic patterns

Kung-Jiuan Yang [a], Tzung-Pei Hong [b,c,*], Guo-Cheng Lan [d], Yuh-Min Chen [e]

[a] Department of Information Management, Fortune Institute of Technology, 831, Taiwan
[b] Department of Computer Science and Information Engineering, National University of Kaohsiung, 811, Taiwan
[c] Department of Computer Science and Engineering, National Sun Yat-Sen University, 804, Taiwan
[d] Department of Mathematics and Computer Sciences, Fuqling Branch of Fujian Normal University, Fuqing 350300, China
[e] Institute of Manufacturing Information and Systems, National Cheng Kung University, 701, Taiwan

ARTICLE INFO

Article history:
Received 19 May 2013
Received in revised form
16 October 2013
Accepted 5 January 2014
Available online 4 February 2014

Keywords:
Data mining
Event
Partial periodic pattern
Projection
Weight-based

ABSTRACT

Partial periodic pattern mining has recently become an important issue in the field of data mining due to its wide applications in many businesses. A partial periodic pattern considers part of but not all the events within a specific period length, repeating with high frequency in an event sequence. Traditional partial periodic pattern mining, however, only considered the frequencies of patterns, but did not consider events that might have different importance. The study thus proposes a weighted partial periodic patterns mining algorithm to resolve this problem. To increase the efficiency, the two-phase upper-bound weighted model based on segmental maximum weights is adopted to prune unimportant candidates in early stage. Then the weighted partial periodic patterns are discovered from the candidate patterns. Finally, the experimental results on synthetic datasets and a real oil dataset show that the weighted partial periodic pattern mining is more practical to assist users for decision making.

1. Introduction

Sequential pattern mining is widely applied to the search for frequent patterns for analyzing biological, computer network, and market basket data. Many studies have proposed mining algorithms that improve the efficiency and effectiveness of sequential pattern mining for mining closed (Tzvetkov et al., 2003; Wang and Han, 2004), incremental (Cheng et al., 2004; Liu et al., 2007), multi-dimensional (Pinto et al., 2001), multiple minimum supports (Chen et al., 2011; Madan Kumar et al., 2012) and approximate (Yun and Ryu, 2011) sequential patterns. However, traditional sequential pattern mining only considers the frequencies of patterns in a sequence database, which is an insufficient indicator to find meaningful or important patterns. To resolve this problem, several weight-based sequential pattern mining algorithms were proposed (Yun and Leggett, 2006; Yun, 2008; Yun and Ryu, 2010). The main task in weight-based sequential pattern mining is maintaining the downward-closure property when applying the weight constraints (Agrawal and Srikant, 1995). Because a pattern is weighted as being infrequent, its super patterns can still be weighted as frequent after other items with higher weights are added. In many fields of sequential pattern mining, periodic pattern mining is an important one for discovering regularity in time series or event sequences (Özden et al., 1998). Since real-world activities may not have full regularity, partial periodic pattern mining, which considers most but not all points in the period contributing to the approximate cyclic behavior of the time series, has been developed. Partial periodic pattern mining is a less restrictive form of mining, making it more practical. The main concept of partial periodic pattern mining was first introduced in the study of Han et al. (1998), which found that partial periodicity could be associated with a subset of the time points of the periodic behavior. In other words, a partial periodic pattern is a mixture of periodic and non-periodic events. For example, "The stock's price goes up on Wednesdays" is partial periodicity because it only considers Wednesdays and says nothing about the price fluctuations throughout the rest of the week. For partial periodic pattern mining, "price goes up" and "a week" can be defined as an event and a period, respectively. Note that the non-periodic event positions are usually denoted by the symbol "*". Based on traditional partial periodic pattern mining, if the minimum confidence is set at 60%, only the partial periodic pattern {**c**} with a confidence of 80% (=4/5) can be found. The partial periodic pattern {b****} is removed because the confidence is 40% (=2/5), which is below the minimum confidence. In this case, pattern {b****} is removed even if it is very important. Therefore, related important patterns associated with event *b* will not be generated. Unimportant patterns with high frequency, such as the pattern {**c**}, continuously generate

* Corresponding author at: Department of Computer Science and Information Engineering, National University of Kaohsiung, 811, Taiwan.
E-mail address: tphong@nuk.edu.tw (T.-P. Hong).

0952-1976/$ - see front matter © 2014 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.engappai.2014.01.004

many less important frequent *x*-patterns, which do not provide valuable information for making decisions.

The present study proposes an efficient *p*rojection-based *w*eighted upper-bound partial periodic pattern mining *a*lgorithm (abbreviated as PWA) that adopts the upper-bound model to prune candidates and applies the efficient projection-based partial periodic pattern mining algorithm proposed by Yang et al. to find the weighted partial periodic patterns with a specific period length in an event sequence. Yun and Ryu, 2010 also applied the maximum weight among items as the maximum weight of each sequence to discover weighted sequential patterns from a sequence dataset. The major difference between Yun et al.'s upper-bound model and proposed PWA is that the maximum weight value of each segment in PWA could be changed after creating the projection database. The PWA is designed to use two-phased variable upper-bounds of maximum weight for each segment and adopts a projection-based algorithm to reduce candidates to quickly recognize weighted partial periodic patterns. Experimental results reveal good performance in reducing the number of redundant patterns to discover meaningful weighted partial periodic patterns.

The remainder of this paper is organized as follows. In Section 2, concepts related to weight-based sequential patterns and the partial periodicity are introduced. In Section 3, the problem statement and definitions are given. Section 4 presents the proposed algorithm for mining weighted partial periodic patterns, and an example of the application of the proposed algorithm is given. Section 5 describes the experimental evaluation using synthetic datasets and eleven years of Brent oil data. Finally, conclusions are given in Section 6.

2. Review of related works

In this section, studies related to weighted partial periodic pattern mining, partial periodic pattern mining and weight-based sequential pattern mining are briefly reviewed.

2.1. The PrefixSpan algorithm

Mining sequential patterns is to discover frequently occurred subsequences in a sequence database. In the past, most of the traditional sequential mining techniques were based on the apriori-like related algorithms, such as *AprioriAll*, *AprioriSome*, and *DynamicSome* (Agrawal and Srikant, 1995). These algorithms, however, had to spend a lot of expensive time cost for dealing with candidate subsequence generation. To resolve the problem, an algorithm, namely *PrefixSpan*, was proposed by Pei et al., which examined only prefix subsequences and projected only corresponding postfix subsequences into a set of smaller projected databases. The algorithm could effectively reduce the effort of candidate subsequence generation while mining a complete set of patterns and substantially reduced the sizes of projected databases (Pei et al., 2004). For example, for a sequence database which includes four sequences, $<a(abc)(ac)d(cf)>$, $<(ad)c(bc)(ae)>$, $<(ef)(ab)(df)cb>$ and $<eg(af)cbc>$, assume $min\_sup=2$. Take the first sequence $<a(abc)(ac)d(cf)>$ as an example. $<a>$, $<aa>$, $<a(ab)>$ and $<a(abc)>$ are prefixes of the sequence. If $<a>$ is chosen as the prefix, the suffix will be $<(abc)(ac)d(cf)>$. If the prefix is $<aa>$, the suffix will be $<(\_bc)(ac)d(cf)>$. The prefix-projection method includes the following steps. (1) Find length-1 sequential patterns. In the above example, the patterns found include $<a>:4$, $<b>:4$, $<c>:4$, $<d>:3$, $<e>:3$ and $<f>:3$ in which the associate counts are greater than or equal to *min_sup*. (2) Project the database into smaller projected databases according to the length-1 patterns. For the above example, after the first projection, there are six projected databases, and the sequences $<(abc)(ac)d(cf)>$, $<(\_d)c(bc)(ae)>$, $<(\_b)(df)cb>$ and $<(\_f)cbc>$ are in the $<a>$-*projected database*. (3) Find subsets of sequential patterns by scanning a *projected database*, so that the set of frequent items can be assembled to the last element of a prefix item for forming a sequential pattern; or each frequent item can be appended to a prefix item to form a sequential pattern by constructing corresponding projected databases and to mine recursively. For example, to find the sequential patterns with the prefix $<a>$, the approach needs to scan the $<a>$-*projected* database once, get the six length-2 sequential patterns with counts which have the prefix $<a>$, such as $<aa>:2$, $<ab>:4$, and so on. Then recursively, all sequential patterns having prefix $<a>$ can be further partitioned into several length-2 subsets – those with prefix $<aa>$-, $<ab>$-, and so on. Projected databases are then constructed according to the subsets and the same procedure is repeated to mine useful patterns.

Afterward, there were several studies which applied the *Prefixspan* algorithm to solving various practical applications. Qiao et al. (2010) proposed the Plute algorithm with prefix projection to decompose a search space and intensively reduced the candidate trajectory sequences. Vijayalakshmi et al. (2009) introduced an EXT-*Prefixspan* algorithm in web-usage mining to extract multi-dimensional frequent sequential patterns. Zhao et al. (2012) proposed the U-PrefixSpan algorithm to probabilistically mine frequent sequential patterns in uncertain databases. Their experiments showed that the approaches could effectively avoid the problem of "possible world explosion" in RFID applications. All these papers applied the prefix projection to reduce candidate subsequence generation, substantially reduced the size of projected database, and led to efficient mining.

2.2. Partial periodic pattern mining

Periodic pattern mining can be divided into full periodic pattern mining and partial periodic pattern mining. For full periodic pattern mining, each position in a pattern has to be definite and nonempty. The constraint in full periodic pattern mining is strict since all the symbols in a pattern must appear in order and at the same time in segments with a high ratio. However, the counts of some periodic patterns in an event sequence might be a little less than the minimum support threshold. And such patterns will not be found by full periodic pattern mining techniques. Hence, partial periodic pattern mining was proposed to deal with this problem. To find partial periodic patterns in an event sequence, Han et al. (1999) proposed an effective tree-based mining algorithm called the max-subpattern hit set method. Two scans of an event sequence are required to build the max-subpattern tree. In the first scan of the event sequence, all frequent 1-patterns, in each of which there is only one definite symbol (the other symbols are all '*'), are found in the sequence, and then a candidate max-pattern (C_{max}) is generated from the set of frequent 1-patterns as the root of the tree. Next, the second scan of the event sequence is conducted to intersect each period segment recursively with C_{max} to expand a child node as a subpattern of C_{max} with one non-* letter replaced with the symbol '*'. If a segment can match a subpattern of C_{max}, the existing node for the subpattern is increased by 1.

The max-subpattern hit set method, however, generates a large subpattern tree when there are a large number of events in one position or a max-subpattern is long. For example, assume that "$a\{bcdef\}d*\{ab\}$" is a max-subpattern with the length of 5. Since the max-subpattern includes many events {*bcdef*} in the second position, a large subpattern tree is created and a lot of time is required to traverse the tree to find possible partial periodic patterns. Recently, Yang et al. adopted the projection method (Pei et al., 2004) and the encoding method used in Han et al.'s (1998) study to re-encode events into a new event representation using the positions of the