International Journal of Data Warehousing and Mining
Volume 16 • Issue 1 • January-March 2020

Mining Integrated Sequential Patterns From Multiple Databases

Christie I. Ezeife, University of Windsor, Ontario, Canada

Vignesh Aravindan, Royal Bank of Canada, Canada

Ritu Chaturvedi, School of Computer Science, University of Guelph, Ontario, Canada

ABSTRACT

Existing work on multiple databases (MDBs) sequential pattern mining cannot mine frequent sequences to answer exact and historical queries from MDBs having different table structures. This article proposes the transaction id frequent sequence pattern (TidFSeq) algorithm to handle the difficult problem of mining frequent sequences from diverse MDBs. The TidFSeq algorithm transforms candidate 1-sequences to get transaction subsequences where candidate 1-sequences occurred as (1-sequence, itssubsequenceidlist) tuple or (1-sequence, position id list). Subsequent frequent i-sequences are computed using the counts of the sequence ids in each candidate i-sequence position id list tuples. An extended version of the general sequential pattern (GSP)-like candidate generates and a frequency count approach is used for computing supports of itemset (I-step) and separate (S-step) sequences without repeated database scans but with transaction ids. Generated patterns answer complex queries from MDBs. The TidFSeq algorithm has a faster processing time than existing algorithms.

KEYWORDS

Candidate Generation, Complex Queries, Foreign key, Frequent Itemsets, Frequent Patterns, Frequent Sequences, Multiple Databases, Sequence Database, Transaction Ids

INTRODUCTION

Existing works are mostly for mining frequent itemsets/sequences from single databases (Han, Kamber & Pei, 2012; Nanopoulos & Manolopoulos, 2000). Work does not exist for a sequential pattern algorithm that mines exact frequent sequences from multiple tables or databases that are related through foreign key attributes. For more useful interpretation and application of frequent patterns to real life cases where patterns from different tables or databases related through foreign key attributes need to be integrated to answer relevant queries, algorithms for mining frequent sequences from multiple data sources that carry foreign key tags (e.g., transaction id) are important. Existing work on mining frequent itemsets from transaction tables can be classified into Apriori- and nonApriori-based algorithms, including the Fp-tree algorithm (Agrawal & Srikant, 1994; Srikant & Agrawal, 1995; Han, Pei, Yin & Mao, 2004). Some prominent Apriori-based sequence pattern mining (SPM) algorithms on single databases include GSP (Srikant & Agrawal, 1996). Frequent sequence mining algorithms that are non-Apriori based include SPAM and Prefix-span (Ayres, Flannick, Gehrke, & Yiu, 2002; Pei, Han, Mortazavi-asl, & Zhu, 2000). Algorithms specifically for mining Web sequential patterns include WAP-tree and PLWAP-tree algorithms (Pei, Han, Mortazavi-asl, & Zhu, 2000; Ezeife, Lu, & Liu, 2005).

DOI: 10.4018/IJDWM.2020010101

However, these single sequence/itemset database mining algorithms cannot mine frequent patterns from MDBs or tables like a database with two tables, example drug/side effects sequence table for recording drugs and their side effects with the schema DrugSE(Drugid, Sequences of side effects). The second table is patient/drug sequence table for recording sequences of drugs taken by patients with the schema PatientDr(Patientid, Sequences of Drugids). The DrugSE and PatientDr tables are related through the Drugid foreign key attribute. Regular SPM algorithms, including GSP, can be run on each of these tables. It finds the table drug/side effects with frequent sequences of side effects, as well as the table patient/drug with frequent sequences of drugs (Srikant & Agrawal, 1996). Multiple table scans provide little or no information on finding the patterns. A complex pattern query requiring associating patterns from these two tables, for example "find frequent sequences of side effects suffered from patients p_1 and p_2" cannot be directly or easily answered with these algorithms without additional post-processing database scans. Some reasons for the need to mine frequent patterns from MDBs and example queries for each category include:

1. **Comparative Analysis:** in applications like e-commerce websites where product information (e.g., product name, price) and products sold by online stores (e.g., Best Buy, Walmart) are stored in MDBs and updated frequently. An example historical query is "Find the e-commerce website that sells the cheapest Samsung television products".
2. **Frequent Local and Global Product Pattern Analysis:** There is a need to find frequent local and global patterns of products purchased from customer transaction databases with the same table structure in several local branches.
3. **Mining Frequent Patterns from Multiple Tables with Different Table or Attribute Structures:** There is a need to mine frequent itemsets/sequences from related databases with structures related through foreign/primary key attributes (i.e., patient/drugs and drugs/side effects). For example, "Find patients who are affected by frequent sequences of side effect patterns involving side effect s_1".
4. **Mining Alternate Types of Information:** Patterns for discovering regular product or customer behavior for targeted marketing, such as stable patterns or identifying important customers.

Existing techniques for mining frequent patterns from MDBs include algorithms mining global frequent patterns from multiple tables with the same structures for local databases. Example algorithms are the ApproxMAP algorithm (Kum, Chang, & Wang, 2006), IndividualMine (Peng & Liao, 2009), the hierarchical gray clustering algorithm (HGCA) (Lin, Hu, Li, & Wu, 2013), and clustering local frequency items in MDBs (Adhikari, 2013). An example algorithm that can mine frequent itemsets (not sequences) from MDBs with different structures is the TidFP algorithm (Ezeife & Zhang, 2009).

The main purpose of this article is to propose an algorithm for mining exact frequent sequences from MDBs with different table structures. These database structures are related through foreign key attributes, which would allow answering informative queries involving shared patterns.

Contributions and Problem Definition

Single database sequence mining algorithms cannot mine frequent sequential patterns from multiple related sequences. In addition, they cannot integrate the results to answer queries related to MDBs. This article contributes the following features to the problem of SPM through its newly proposed algorithm (TidFSeq) and work from an unpublished thesis (Aravindan, 2016) for mining exact frequent sequential patterns from general sequences (both multiset and uniset sequences) in MDBs (with different or similar structures) using transaction ids:

1. Answers complex sequence database queries involving related data from more than one table or database.

Hybrid Partitioning-Density Algorithm for K-Means Clustering of Distributed Data Utilizing OPTICS

Mikoaj Markiewicz and Jakub Koperwas (2019). *International Journal of Data Warehousing and Mining (pp. 1-20).*

www.igi-global.com/article/hybrid-partitioning-density-algorithm-for-k-means-clustering-of-distributed-data-utilizing-optics/237135?camid=4v1a