Efficient Mining of Partial Periodic Patterns in Time Series Database

J. Han, G.Dong, Y. Yin

695 KDD

Paper Presentation

University of Alberta

Department of Computing Science

Anita Petrinjak

Outline

- Introduction
- Problem definition
- Methods for mining partial periodic patterns
- Implementation issues
- Performance
- Conclusion
- Future work

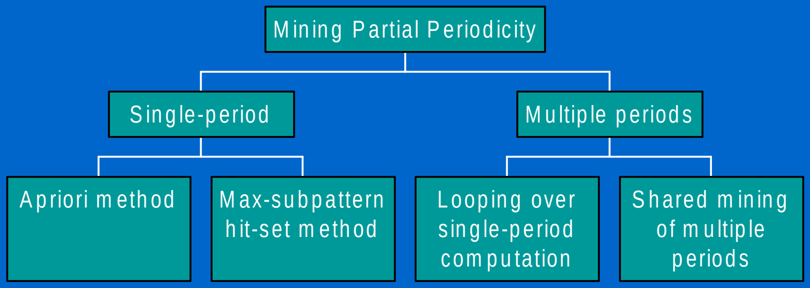
Introduction

- Previous work:
  - mining full periodicity -> not applicable
  - time series data mining -> not periodicity
- Importance
  - real life patterns - mostly partially periodical
- Goal:
  - discovering all frequent patterns of the time series for one period or a range of periods

Problem definition

- Partial periodic pattern:
  - abc***d*abc***d*abc***d*
  - s = abc***d* :
    - length of period=8
    - L-length=4
  - frequency count, m - maximum number of periods of length |s|
  - confidence(s)=frequency count/m
  - confidence threshold

Methods


Mining Partial Periodicity
Single-period
Multiple periods
Apriori method
Max-subpattern hit-set method
Looping over single-period computation
Shared mining of multiple periods

Single-period methods

- Apriori on periodicity
  - each subpattern of a frequent pattern of period p is itself a frequent pattern of period p.
  - Apriori-like algorithm: p database scans
  - PROBLEM:
    - the number of frequent i-patterns shrinks slowly as i increases (unlike in mining association rules)

Single-period methods

- Single-period max-subpattern hit set method
  - improved mining of partial periodicity:
  - 1) max-pattern, Cmax
    - combination of all 1-patterns
    - Example: {a***, *b**, **c*} -> Cmax=abc*
  - 2) subpattern is hit of a period segment if it is maximal subpattern of Cmax and is true in that period segment

Single-period methods

- Single-period max-subpattern hit set algorithm:
  - 1) Scan db to find set of frequent 1-patterns, generate max-pattern Cmax
  - 2) Scan db to find max-subpattern for each period segment in time-series data
  - 3) Derive frequent patterns from the max-subpattern tree
- 2 database scans

Multiple periods methods

- Looping over single period computation
  - find all partial periodic patterns for a set of periods p1, …, pk
- Algorithm:
  - 1) for each period pj in the range of interest (p1, …, pk) apply max-subpattern hit set algorithm
- 2xk database scans

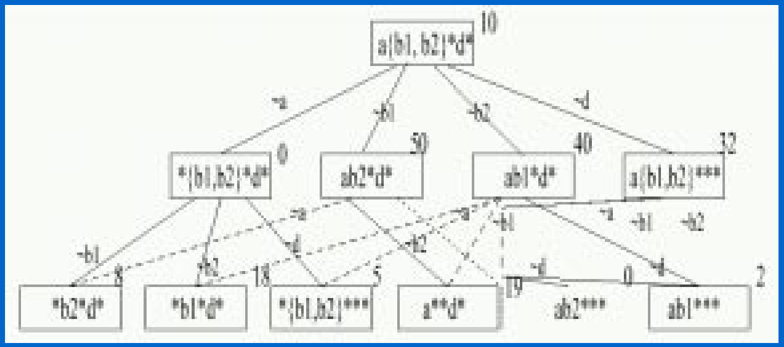
Multiple periods methods

- Shared mining of multiple periods
  - given set of periods p1, …, pk
- Algorithm:
  - 1) find sets of 1-patterns and Cmax-s for each period pj
  - 2) scan db, find max-subpatterns for every period segment for each period pj
- 2 database scans
- requires more space than multiple scan

Implementation issues

- Max-subpattern hit set algorithm - efficient implementation:
  - good data structure:
    - storing the set of max-subpattern hits
    - deriving the set of all frequent patterns
  - new data structure: max-subpattern tree

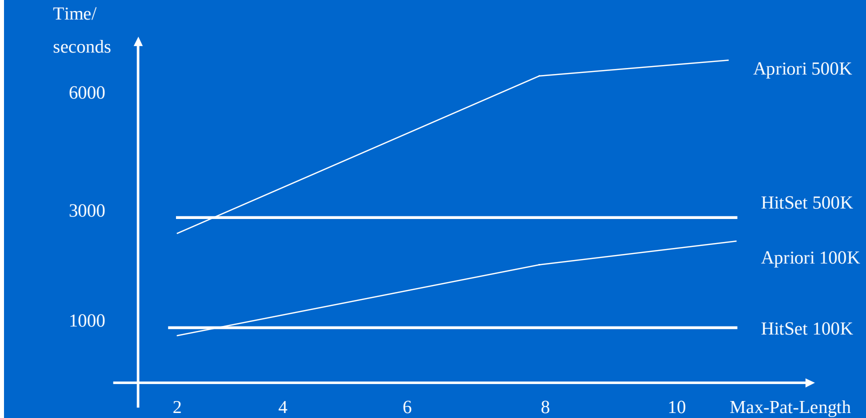
Max-subpattern tree



Max-subpattern tree

- Insertion of a node in the max-subpattern tree:
  - starting from the root, find the corresponding node by checking the missing non-* letter
  - if node exists, increase count, otherwise create it
- Derivation of frequent patterns from max-subpattern tree
  - Apriori-like technique

Performance



Conclusion/Summary

- Mining partial periodicity in time series database
- Properties:
  - Apriori-like property
  - max-subpattern hit set property
- Best performance: Max-subpattern hit set algorithm
- Multiple periods mining: lots of scans or lot of memory

Future work

- Further exploration of shared mining for periodicity with multiple periods
- Mining periodic association rules based on partial periodicity
- Query- and constraint- based mining of partial periodicity