Mining Maximal Sequential Patterns without Candidate Maintenance

Artificial Intelligence
Seminar
10 March 2016

Syméon Malengreau
Mhd Ayad Aldayeh

Table of contents

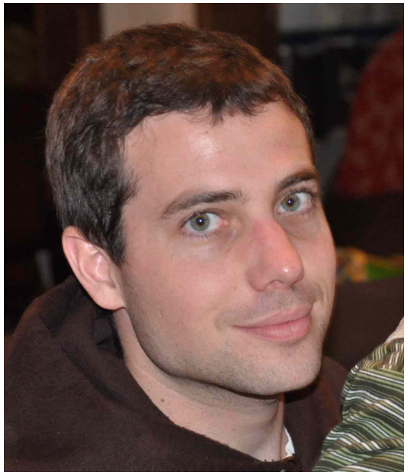
Reference : Pierre Schauss

Let's take a comparison for the article's authors

Pierre Schauss is a UCL professor, you might know him

H-index : 10

Citations : 283



Authors

Vincent S. Tseng

Philippe Fournier-Viger

Cheng-Wei Wu

Authors : Vincent S. Tseng

Ph.D. in Computer Science

Professor, Dept. Computer Science, National Chiao Tung University, Taiwan

H-index : 32

3,2x more

Citations : 3308

11,7x more

Authors : Philippe Fournier-Viger

Ph.D. in Computer Science

Associate Professor, Harbin Institute of Technology, Shenzhen Graduate School

He created the SPMF library

H-index : 17

- 1,7x more

Citations : 712

- 2,5x more

Authors : Cheng-Wei Wu

H-index : 12

1,2x more

Citations : 518

1,83x more

First Concepts

Sequence Database

Sequential pattern

Closed sequential pattern

Maximal sequential pattern

Sequence Database

A sequence database consist of :

- A set of items
  - {1, 2, 3, 4, …, N}
- Itemset (set of item, distinct and unordered)
  - {1, 2, 3, 5} or {4, 5} or {3, 7} or ...
- Sequence (set of itemsets)
  - <{1,2}, {3}, {5}> or <{4}, {6}> or ...

The sequence database is a set of sequences

What do theses concepts represents ?

Sequence Database : Illustration

Let's take as an example a book

Set of item → The words

{He, nice, the, is, a, guy, sun, shine, ...}

Itemset → A sentence (where words are distinct and unordered)

{He, a, nice, guy, is}

{The, sun, shine, in, the, sky}

Sequence → A chapter of the book

Sequence Database → The book

Sequential pattern

Synonyms are *sub-sequence* or *frequent sequence*

It is a sequence of item that appears a certain number of time, that number is the *minimum support threshold* (or *minsup*)

Sequence database

- <{1,2},{3},{4},{6}>
- <{2},{5},{6}>
- <{1,3},{5},{6}>

With minsup = 2, some examples of sequential pattern

- {5},{6}
- {1}
- {3},{6}
- …

Closed sequential pattern

A closed sequential pattern is a sequential pattern not included in another closed pattern having the same frequency.

<{1},{1 2 3},{1 3},{4},{3 6}>

<{1 4},{3},{2 3},{1 5}>

<{5 6},{1 2},{4 6},{3},{2}>

<{5},{7},{1 6},{3},{2},{3}>

With support 2 (or 2/4 entry → 50 %), here are some closed sequential pattern

{1}, {3} 100 % (4/4)

{1}, {3}, {2} 75 % (3/4)

{5}, {1}, {3}, {2} 50% (2/4)

{5} 75 % (3/4)

And this one is NOT

{1} 100 % (4/4)

Maximal sequential pattern

The same as the closed sequential pattern, but if one sequence is in another one, it is not maximal.

Interesting property :

- You can derive every closed sequential patterns from the maximal sequential patterns

MaxSP Algorithm

Find the maximal sequential pattern

It is build uppon the PrefixSpan Algorithm

Why the need for a new algorithm ?

→ Less memory usage

→ Faster to find sequential pattern

PrefixSpan : Start

First let's explain the PrefixSpan Algorithm

We start with a sequence database

<{1},{1 2 3},{1 3},{4},{3 6}>

<{1 4},{3},{2 3},{1 5}>

<{5 6},{1 2},{4 6},{3},{2}>

<{5},{7},{1 6},{3},{2},{3}>

PrefixSpan : Pattern-growth

It works by pattern-growth, which does not generate any candidates (saving memory)

1. Scan : ***Calculate support for each item***

2.

PrefixSpan : Scan

MinSup 75 % (3)

<{1},{1 2 3},{1 3},{4},{3 6}>

<{1 4},{3},{2 3},{1 5}>

<{5 6},{1 2},{4 6},{3},{2}>

<{5},{7},{1 6},{3},{2},{3}>

Item	Support
1	
2	
3	
4	
5	
6	
7	

PrefixSpan : Scan

MinSup 75 % (3)

- <{1},{1 2 3},{1 3},{4},{3 6}>
- <{1 4},{3},{2 3},{1 5}>
- <{5 6},{1 2},{4 6},{3},{2}>
- <{5},{7},{1 6},{3},{2},{3}>

Item	Support
1	100 % (4)
2	
3	
4	
5	
6	
7	

PrefixSpan : Scan

MinSup 75 % (3)

- <{ },{ 2 3},{ 3},{4},{3 6}>
- <{ 4},{3},{2 3},{ 5}>
- <{5 6},{ 2},{4 6},{3},{2}>
- <{5},{7},{ 6},{3},{2},{3}>

Item	Support
1	100 % (4)
2	100 % (4)
3	
4	
5	
6	
7	

PrefixSpan : Scan

MinSup 75 % (3)

<{ },{ 3},{ 3},{4},{3 6}>

<{ 4},{3},{ 3},{ 5}>

<{5 6},{ },{4 6},{3},{ }>

<{5},{7},{ 6},{3},{ },{3}>

Item	Support
1	100 % (4)
2	100 % (4)
3	100 % (4)
4	
5	
6	
7	

PrefixSpan : Scan

MinSup 75 % (3)

- <{ },{ },{ },{4},{ 6}>
- <{ 4},{ },{ },{ 5}>
- <{5 6},{ },{4 6},{ },{ }>
- <{5},{7},{ 6},{ },{ },{ }>

Item	Support
1	100 % (4)
2	100 % (4)
3	100 % (4)
4	75 % (3)
5	
6	
7	

PrefixSpan : Scan

MinSup 75 % (3)

<{ },{ },{ },{ },{ 6}>

<{ },{ },{ },{ 5}>

<{5 6},{ },{ 6},{ },{ }>

<{5},{7},{ 6},{ },{ },{ }>

Item	Support
1	100 % (4)
2	100 % (4)
3	100 % (4)
4	75 % (3)
5	75 % (3)
6	
7	

PrefixSpan : Scan

MinSup 75 % (3)

<{ },{ },{ },{ },{ 6}>

<{ },{ },{ },{ }>

<{ 6},{ },{ 6},{ },{ }>

<{ },{7},{ 6},{ },{ },{ }>

Item	Support
1	100 % (4)
2	100 % (4)
3	100 % (4)
4	75 % (3)
5	75 % (3)
6	75 % (3)
7	

PrefixSpan : Scan

MinSup 75 % (3)

<{ },{ },{ },{ },{ }>

<{ },{ },{ },{ }>

<{ },{ },{ },{ },{ }>

<{ },{7},{ },{ },{ },{ }>

Item	Support
1	100 % (4)
2	100 % (4)
3	100 % (4)
4	75 % (3)
5	75 % (3)
6	75 % (3)
7	25 % (1)

PrefixSpan : Scan

MinSup 75 % (3)

- <{1},{1 2 3},{1 3},{4},{3 6}>
- <{1 4},{3},{2 3},{1 5}>
- <{5 6},{1 2},{4 6},{3},{2}>
- <{5},{7},{1 6},{3},{2},{3}>

Item	Support
1	100 % (4)
2	100 % (4)
3	100 % (4)
4	75 % (3)
5	75 % (3)
6	75 % (3)
7	25 % (1)