Sequence Prediction using Partially-Ordered Episode Rules

Yangming Chen¹, Philippe Fournier-Viger¹,², Farid Nouioua², Youxi Wu³

1 Harbin Institute of Technology (Shenzhen), China
2 Shenzhen University, China
3 University of Bordj Bou Arreridj, Algeria
4 Hebei University of Technology, Tianjin, China

UDML 2021 @ ICDM 2021
Introduction

• **Sequence prediction**: predicting the next event/symbol of a sequence

![Sequence Prediction Diagram]

• **Episode rule mining**: discovering rules to help understand the data or do predictions.

• We compare **three types of rules** for prediction:
  • **Episode rules** (Toivonen et al., 1996)
  • **Precise positioning episode rules** (Ao et al., 2017)
  • **Partially-ordered episode rules** (Fournier-Viger et al., 2021)
Three Types of Episode Rules

1) **Standard episode rule (SER):** rule where events in the antecedent and in the consequent are ordered (found by MINEPI+ algorithm).

Examples:
The rule $\langle\{c\}, \{a\}\rangle \rightarrow \langle\{d\}\rangle$ has an occurrence in time interval $[t_1, t_3]$.

The rule $\langle\{a\}, \{c\}\rangle \rightarrow \langle\{d\}\rangle$ has an occurrence in time interval $[t_5, t_8]$.

**Support:** How many times a rule appears

**Confidence:** The conditional probability that the antecedent is followed by the consequent
3) Precise-Positioning Episode rule (PER): rule where the elapsed time between the antecedent and the consequent is fixed (found by PRU algorithm).

Examples:
The rule $\langle \{b\}, \{d\}\rangle \rightarrow \langle \{a\}\rangle$ has an occurrence in time interval $[t_2, t_5]$

The rule $\langle \{b\}, \{d\}\rangle \rightarrow \langle \{a\}\rangle$ has an occurrence in time interval $[t_7, t_{11}]$
Three Types of Episode Rules

3) Partially-Ordered Episode Rules (POER): rule where events in the antecedent and in the consequent are unordered (found by POERM algorithm).

Example:
The rule <\{a,b,c\}> → <\{d\}> has occurrences in [t1, t3] and [t5, t8].

```
minsupt = 3, minconf = 0.6, XSpan = 3, winlen = 4, YSpan = 1
```
Sequence Prediction

**Goal:** predict an event that will appear in the *suffix* of a sequence given the events observed in the *prefix*.

**Possible outcomes:**
- Good prediction
- Wrong prediction
- Unable to make a prediction (no match)
3. The EpisodePredictor Framework

Phase 1) Training

Training sequence → Building sequence prediction model → Prediction Model (episode rules)

Phase 2) Prediction

Prediction Model → Prediction algorithm → Prediction e.g. d

A sequence e.g. a, b, c → Prediction Model
### Dataset and Default Values

<table>
<thead>
<tr>
<th>Dataset</th>
<th># Timestamps</th>
<th># Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bible</td>
<td>649,024</td>
<td>13,905</td>
</tr>
<tr>
<td>OnlineRetail</td>
<td>2,364,798</td>
<td>2,603</td>
</tr>
<tr>
<td>FIFA</td>
<td>710,435</td>
<td>2,990</td>
</tr>
<tr>
<td>Leviathan</td>
<td>153,682</td>
<td>9025</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Default Values</th>
<th>minsup</th>
<th>minconf</th>
<th>prefixSize</th>
<th>suffixSize</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bible</td>
<td>120</td>
<td>0.3</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>OnlineRetail</td>
<td>650</td>
<td>0.3</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>FIFA</td>
<td>2800</td>
<td>0.3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Leviathan</td>
<td>80</td>
<td>0.3</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>
4. Experimental Evaluation (Influence of $\text{minsup}$)

 Generally, $\uparrow \text{minsup} \Rightarrow \uparrow \text{rule count}, \uparrow \text{matching rate}, \uparrow \text{accuracy}$

Partially-ordered episode rules have the best results.
4. Experimental Evaluation (Influence of $\text{minconf}$)

Generally, $\uparrow \text{minconf} \Rightarrow \downarrow \text{rule count}, \downarrow \text{matching rate}, \downarrow \text{accuracy}$
4. Experimental Evaluation (Influence of *prefix size*)

Generally, $\uparrow$ *prefix size* $\Rightarrow$ $\uparrow$ rule count, $\uparrow$ matching rate, $\downarrow\downarrow$ accuracy
4. Experimental Evaluation (Influence of suffix size)

Generally, \( \uparrow \text{suffix size} \) \( \Rightarrow \uparrow \text{rule count} \), \( \uparrow \text{matching rate} \), \( \uparrow \text{accuracy} \)
5. Conclusion

• Contributions:
  ➢ utilize a new type of episode rules, named **partially-ordered episode rules to improve** sequence prediction
  ➢ Better results are obtained in terms of **accuracy and matching** rate compared to two popular types of episode rules

• Future work:
  ➢ Episode rule-based sequence prediction models for more complex event sequences.
  ➢ Design an incremental episode rule mining algorithm to decrease runtime performance in dynamic environment

➢ Open-source code, datasets:
  ➢ [SPMF data mining library](https://www.spmf.fr/sdm/index.html) (over 200 algorithms)
Thanks for listening!
Frequent Episode Mining

**Input:** a sequence of events with timestamps

![Events and Timestamps Diagram]

**Output:** All frequent episode
(minimal occurrence ≥ minSup)

**For example:**
episode \(<\{a,b\},\{c\}>\)

**Episode types:**
parallel episodes, serial episodes, complex episodes

**Algorithms:**
WINEPI, MINEPI, EMMA, MINEPI+, ...