Discovering temporal exceptions

Data mining in temporal databases

Vanesa Graíño - PhD Student
CiTIUS, University of Santiago de Compostela, Spain
Doctoral Consortium AIME15 - Pavia, Italy - 17/06/2015


CiTiUS
Centro Singular de Investigación en Tecnoloxías da Información
http://citius.usc.es

Contents

Contents

Exception Mining

Clovers

Sequential Exception Mining

Medical records

Patient #1	
1	Headache
2	Vertigo
3	Nose discharge
6	Shortness of breath
7	Nausea
	...

Patient #2	
1	Headache
2	Vertigo
3	Nose discharge
5	Chest pain
7	Shortness of breath
	...

Patient #3	
1	Headache
2	Nose discharge
6	Chest pain
6	Shortness of breath
7	Nausea
	...

Patient #4	
1	Headache
3	Shortness of breath
9	Chest pain
10	Nose discharge
	...
	...

Sequential Exception Mining

Medical records

Patient #1	
1	Headache
2	Vertigo
3	Nose discharge
6	Shortness of breath
7	Nausea
	...

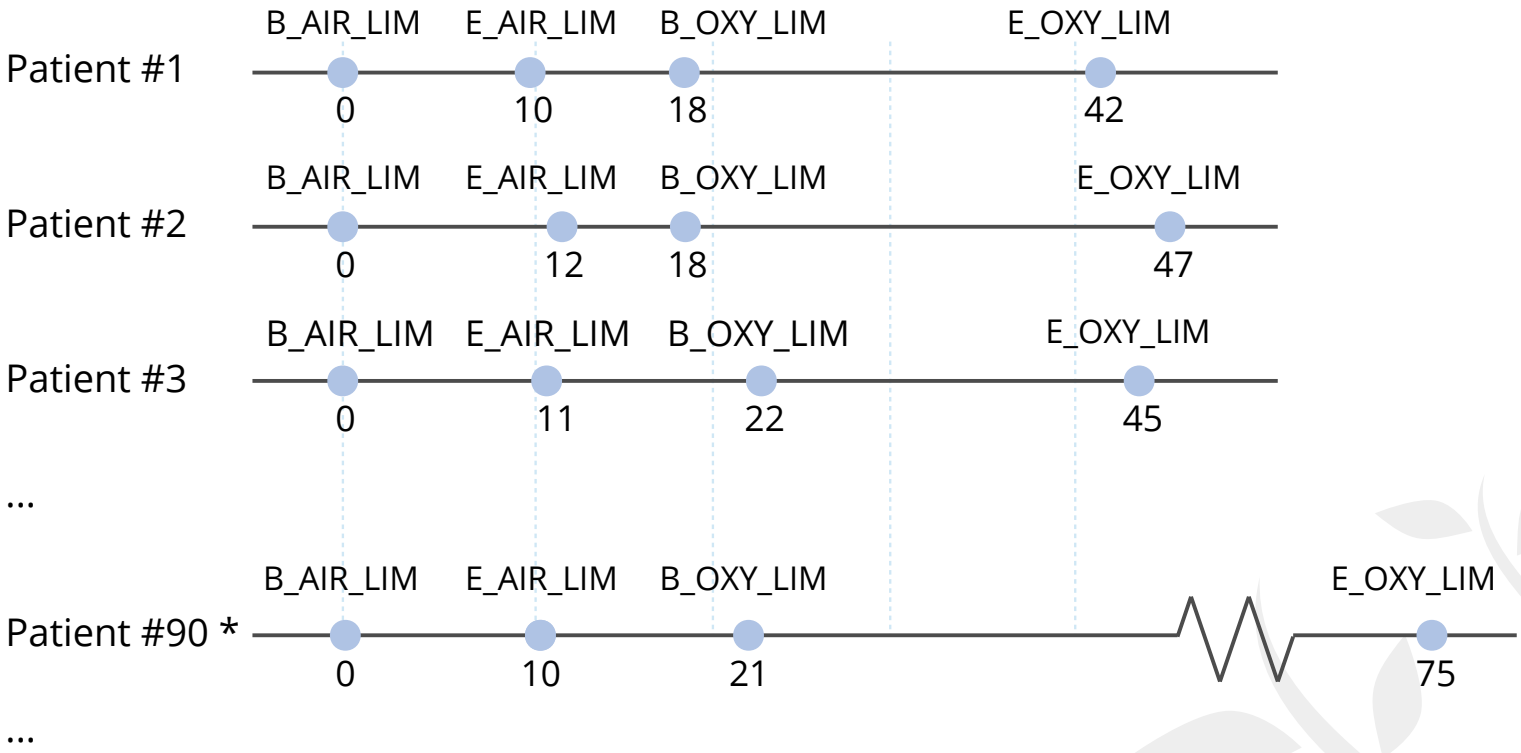
Patient #2	
1	Headache
2	Vertigo
3	Nose discharge
5	Chest pain
7	Shortness of breath
	...

Patient #3	
1	Headache
2	Nose discharge
6	Chest pain
6	Shortness of breath
7	Nausea
	...

Patient #4	
1	Headache
3	Shortness of breath
9	Chest pain
10	Nose discharge
	...
	...

Temporal Exception Mining

Apnea Syndrome



* 10% of the patients are treated with rheumatoid arthritis medicine

Contents

The problem

Objectives

Design and develop new temporal data mining algorithms able to:

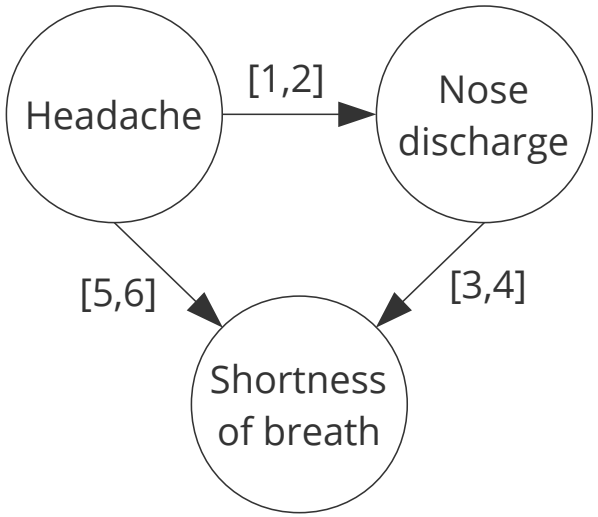
1. Discover **temporal exception patterns**.
2. Consider **events** and **episodes** as temporal entities.
3. Allow a domain expert to introduce **previous knowledge** and **expectations**.
4. Represent **quantitative temporal** information.
5. Provide **expressive, useful, and readable results**.

The solution

Simple Temporal Problem patterns

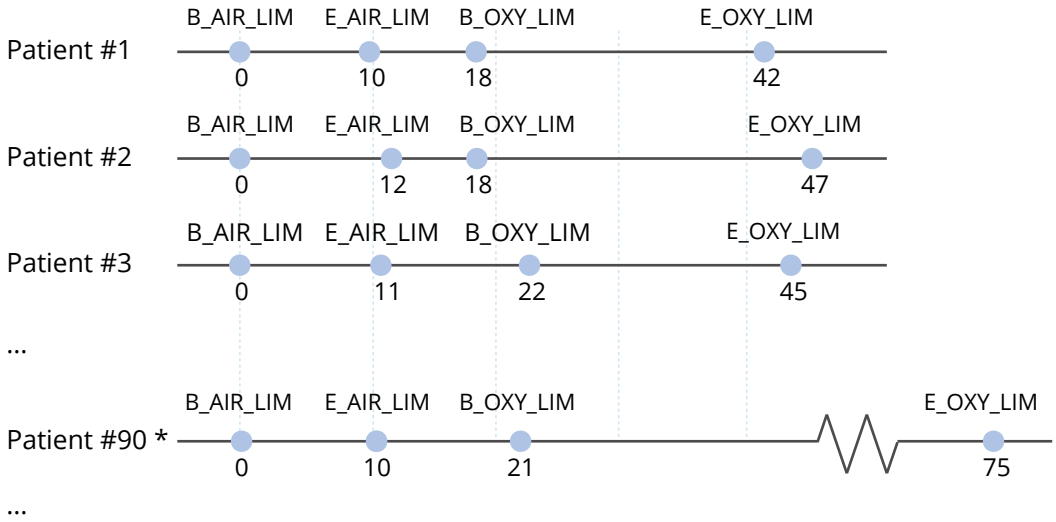
STP Patterns

A **temporal pattern** $P = \langle A, \mathcal{L} \rangle$ of size n consists of a temporal association $A = \{E_1, \ldots, E_n\} \subseteq \mathrm{E}$, and a set of temporal constraints $\mathcal{L} = \{L_{ij};\ 1 \leq i, j \leq n\}$ between the event types in A.

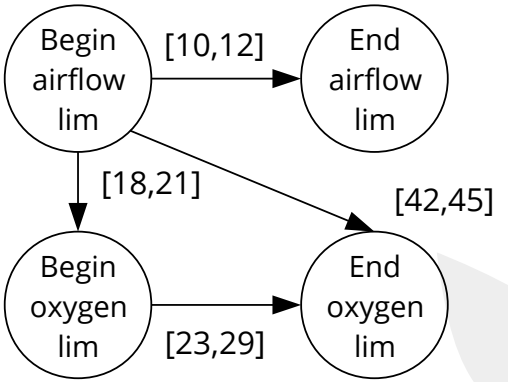


The solution

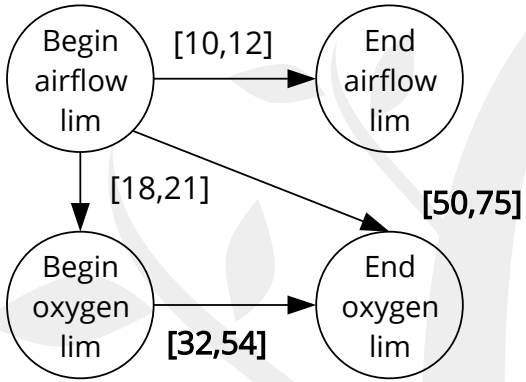
Simple Temporal Problem patterns



* 10% of the patients are treated with rheumatoid arthritis medicine



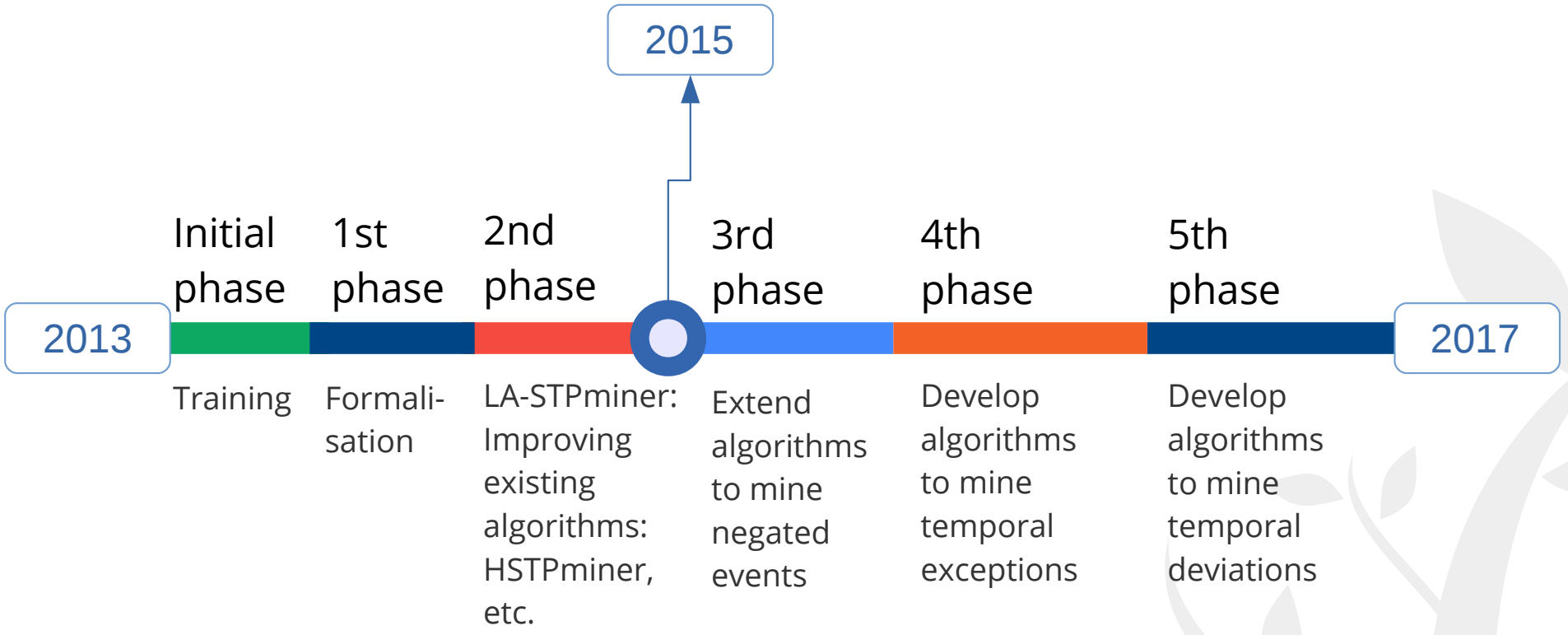
Frequent temporal pattern (expectation)



Temporal exception pattern

Contents

Work status


2015
Initial phase
1st phase
2nd phase
3rd phase
4th phase
5th phase
2013
2017
Training
Formali-sation
LA-STPminer: Improving existing algorithms: HSTPminer, etc.
Extend algorithms to mine negated events
Develop algorithms to mine temporal exceptions
Develop algorithms to mine temporal deviations

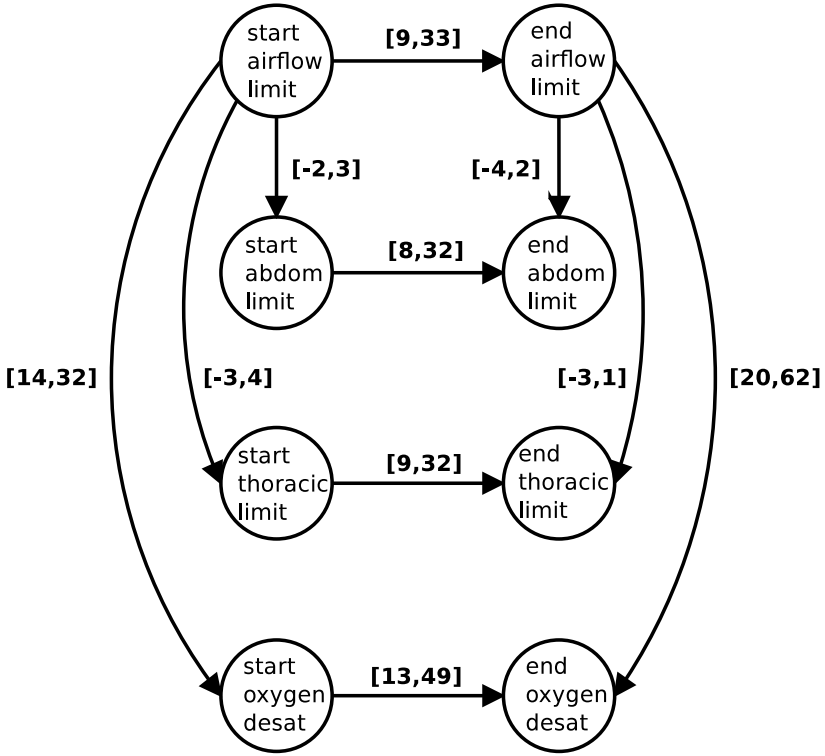
Modelling temporal exception patterns

- Given a temporal pattern representing an **expectation** $P_E = \langle D_E, \mathcal{L}_E \rangle$, an **exception** is defined as a new temporal pattern $P_X = \langle D_X, \mathcal{L}_X \rangle$ such that $P_X = P_{E'} \cup \neg(P_E - P_{E'}) \cup P_A$, where
  - $P_{E'}$ is a subpattern of P_E,
  - $\neg(P_E - P_{E'})$ is the result of negating the rest of the pattern P_E, and
  - P_A is the result of adding new event types and constraints to that of P_E.
- The negation of rest of the pattern can be accomplished either **negating its events**, or **negating its temporal constraints**, or a combination of both.

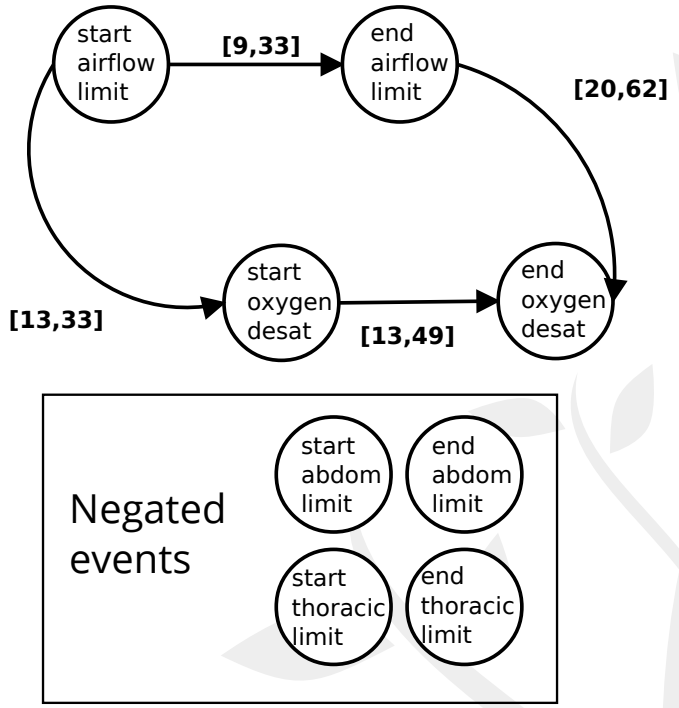
Temporal exception example: SAHS

- Annotated database of 50 SAHS patients, consisting of 120,000 events from 8 event types and 280 hours of sleep.

Central Apnea Pattern



Obstructive Apnea Pattern



Negated-events in temporal mining

Formalisation

- An **extended temporal pattern** of size n is a pair $P = \langle A, \mathcal{L} \rangle$ where
  - A is an extended temporal association $A = \{\hat{E}_i, \ldots, \hat{E}_n\} \subseteq \mathcal{E}, E_1 < \cdots < E_n$ $A = \{A^+ \cup A^-\}$ where
    - A^+ are the positive event types $A^+ = \{\hat{E}_i \in A | \hat{E}_i \in \mathcal{E}\}$ and
    - A^- are the negated event types $A^- = \{\hat{E}_i \in A | \neg\hat{E}_i \in \mathcal{E}\}$
  - $\mathcal{L}$ is a set of temporal constraints $\mathcal{L} = \{L_{ij}; 1 \leq i, j \leq n\}$ between the event types in A^+.
- An **occurrence** of an extended temporal pattern is a temporal window $W = \{e_1, \ldots, e_m\}$ of width ω where every $E_i \in A^+$ has an event occurrence $e_j \in W$ satisfying all the temporal constraints in $\mathcal{L}$ and none of $\neg E_i \in A^-$ has an event occurrence.

Improving algorithms: LA-STPminer

Look-Ahead STP Miner Algorithm

Purpose

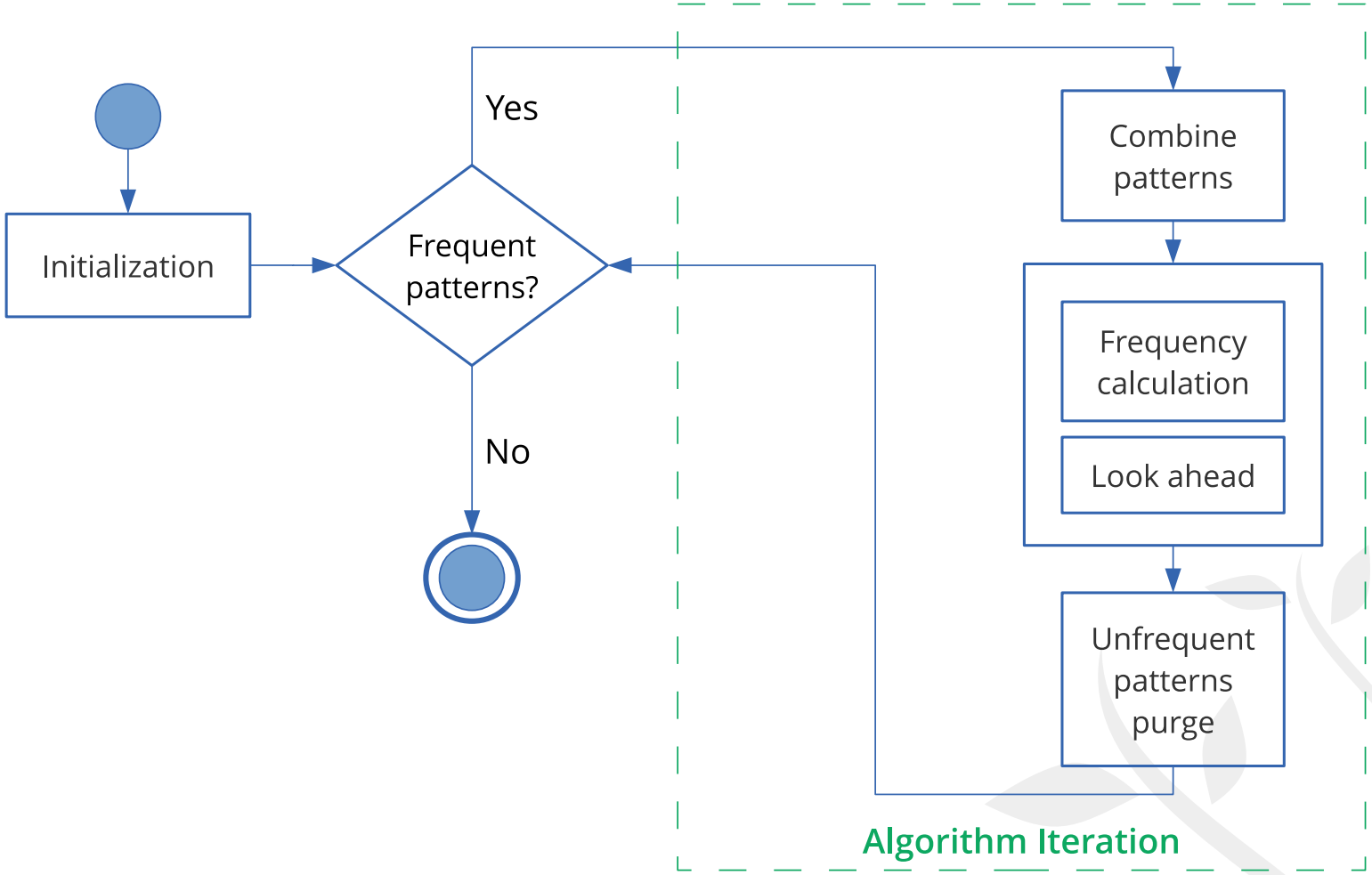
- To extract a set of frequent temporal patterns from a collection of event sequences.
- Each pattern represents a set of temporal arrangements between events considered sufficiently **similar** and **frequent**.

STP Patterns

- Quantitative temporal constraint networks.
- Candidate patterns constructed from previous frequent patterns.
- Size two patterns obtained using a clustering procedure.

LA-STPminer

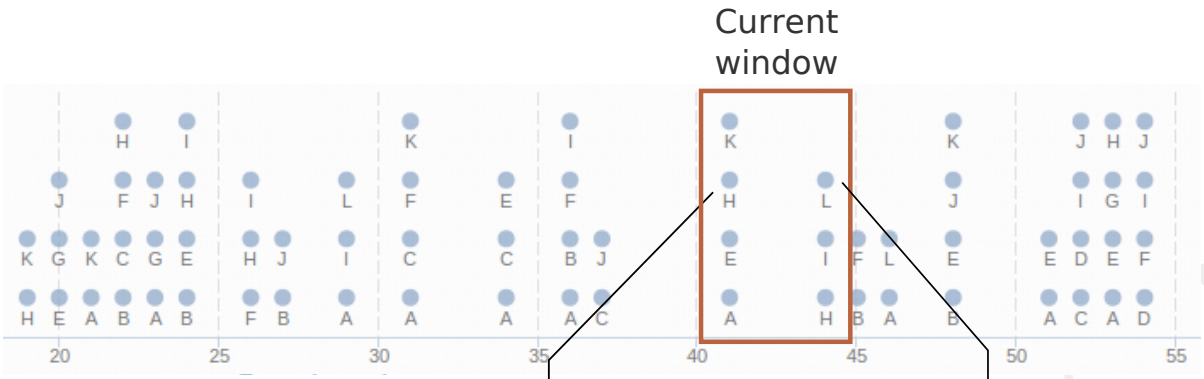
Flow chart



LA-STPminer

Look ahead strategy

- The patterns found in one iteration are **annotated** in the last event of their occurrences, but only if **they can be combined into a bigger association**.
- These annotations are **extended** in the frequency calculation of the next iteration.



Iteration #3

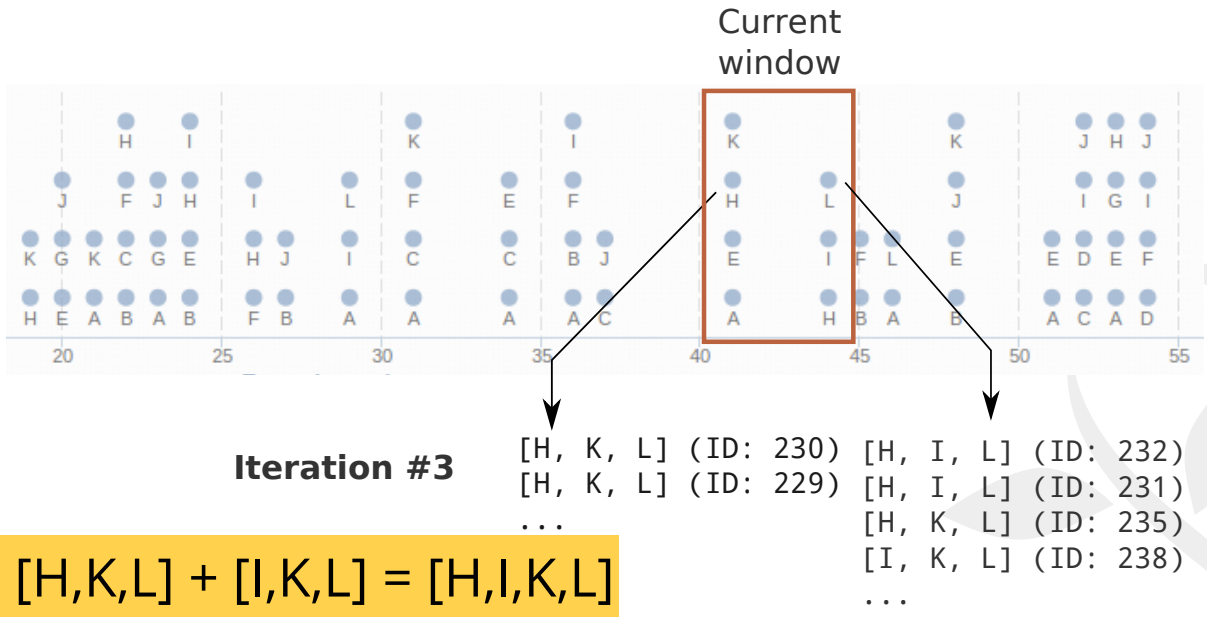
```
[H, K, L] (ID: 230)
[H, K, L] (ID: 229)
...
```

```
[H, I, L] (ID: 232)
[H, I, L] (ID: 231)
[H, K, L] (ID: 235)
[I, K, L] (ID: 238)
...
```

LA-STPminer

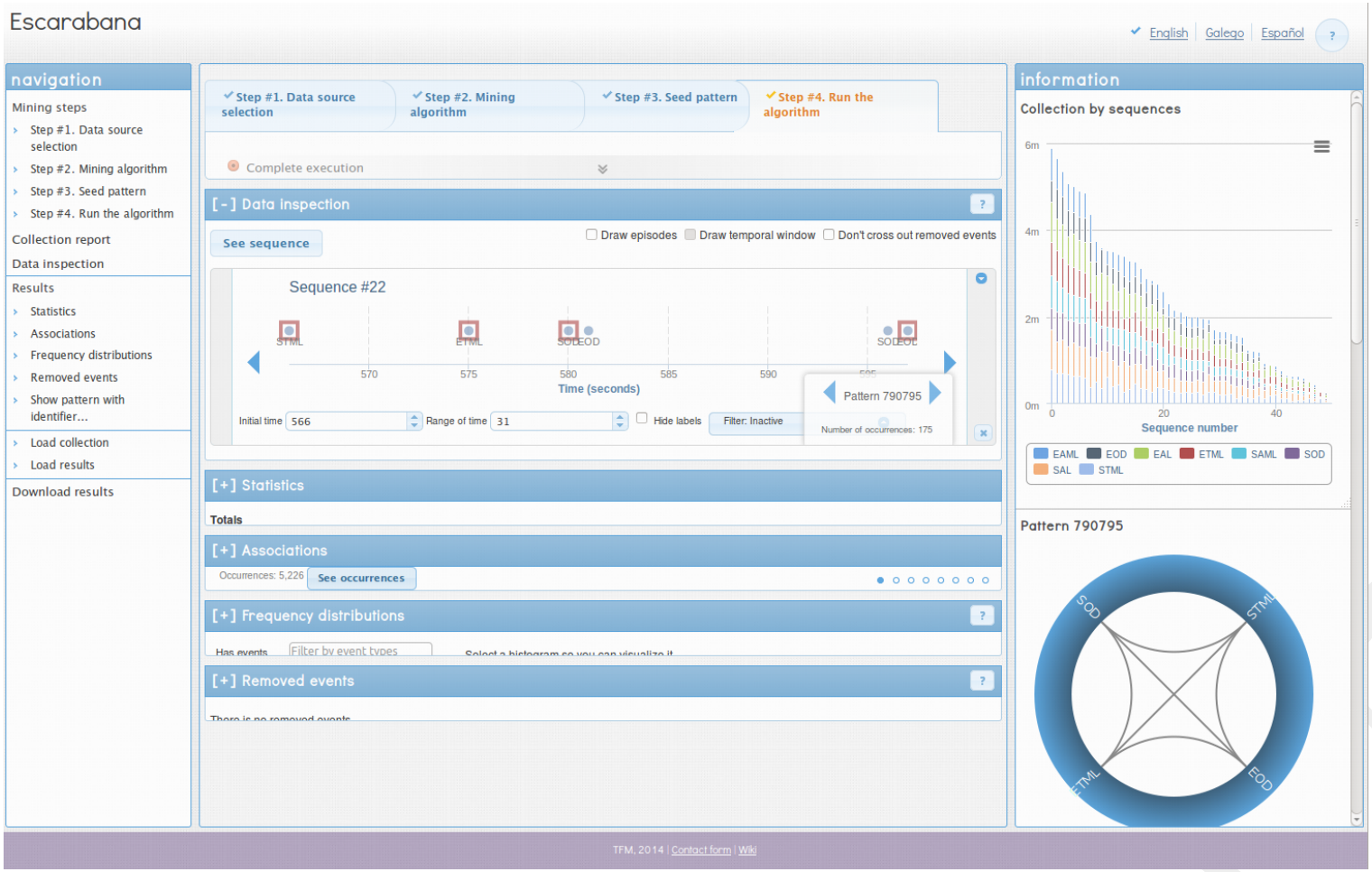
Look ahead strategy

- The patterns found in one iteration are **annotated** in the last event of their occurrences, but only if **they can be combined into a bigger association**.
- These annotations are **extended** in the frequency calculation of the next iteration.



[H,I,K] + [H,I,L] + [H,K,L] + [I,K,L] = [H,I,K,L]

Escarabana


Escarabana
English | Galego | Español
navigation
Mining steps
Step #1. Data source selection
Step #2. Mining algorithm
Step #3. Seed pattern
Step #4. Run the algorithm
Collection report
Data inspection
Results
Statistics
Associations
Frequency distributions
Removed events
Show pattern with identifier...
Load collection
Load results
Download results
Step #1. Data source selection
Step #2. Mining algorithm
Step #3. Seed pattern
Step #4. Run the algorithm
Complete execution
[-] Data inspection
See sequence
Draw episodes
Draw temporal window
Don't cross out removed events
Sequence #22
Time (seconds)
Pattern 790795
Number of occurrences: 175
Initial time
Range of time
Hide labels
Filter: Inactive
[+] Statistics
Totals
[+] Associations
Occurrences: 5,226
See occurrences
[+] Frequency distributions
[+] Removed events
information
Collection by sequences
Sequence number
EAML EOD EAL ETML SAML SOD SAL STML
Pattern 790795
TFM, 2014 | Contact form | Wiki

Open problems

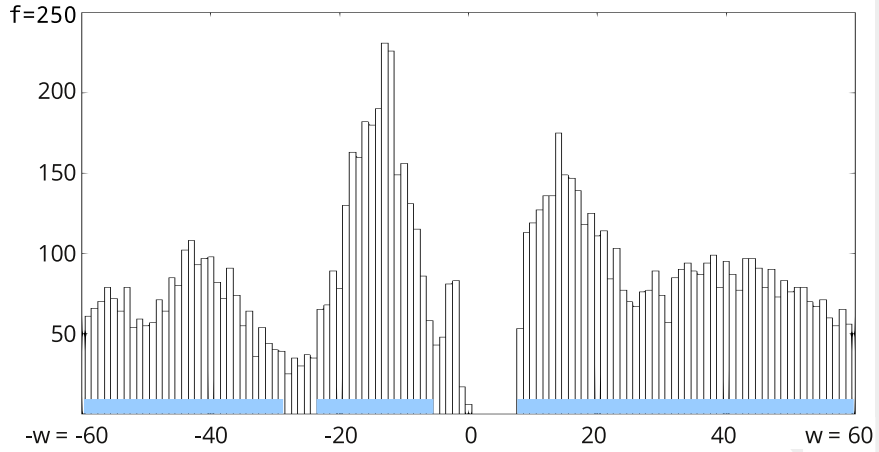
Temporal exception patterns

- Measures to:
  - Distinguish an interesting exception pattern from a non-interesting one.
  - Distinguish noise from an exception pattern.

Open problems

Temporal deviation patterns

- **Temporal deviation patterns** are temporal anomalies inside a dataset. They represent instances that are a relatively frequent **distorsion** from what is frequent **inside the dataset** when new data is added. They are not an opposition to an expectation.
- **Measures** of how different is a pattern from another in order to consider it a deviation.
- Comparing their temporal distances distributions between pair of event types:
  - Statistical tests: $\tilde{\chi}^2$ test which tells if two histograms come from the same population.
  - Distance function with a threshold:
    - Manhattan, or City Block Distance
    - Cosine distance
    - ...



Temporal distance distribution example

Contents

Summary

- Conclusions
  - Improving of existing algorithms: parallelization and new strategies.
  - Modelling of the problem
  - Visualization tool
- Mid- and long-term problems:
  - Design of measure that quantifies the **interestingness** of temporal exception and deviation patterns.
  - Design of a **deviation measure** between different temporal arrangements.
  - Generation of synthetic databases with exceptions and deviations.
- Short-term problem: A need for **temporal databases** where there may be deviations and/or exceptions.

Thank You! Questions?

Contact information:

email vanesa.graino@usc.es
twitter @nessia.gp
www es.linkedin.com/in/nessiagp/es
github github.com/Nessia