On Root Cause Analysis

A case study with Supervised and Unsupervised Machine Learning

Kenneth EZUKWOKE[1], Samuel GRUFFAZ[2],
Mouhamadou-Lamine NDAO[3], Amine MELAKHSOU[1]
{ifeanyi.ezukwoke, amine.melakhsou}@emse.fr
{samuel.gruffaz@ens-paris-saclay.fr, mlndao@cesi.fr}

[1]Mines Saint-Etienne, [2]ENS Paris-Saclay, [3]CESI

March 31, 2023

Introduction

Root Cause Analysis for Process Mining

- Certain units generate a violation of BPMN (Business Process Model and Notation). It is then interesting to seek to explain the cause of this violation, which takes the form of a binary variable.
- Variables describing the units (supplier, delivery person, type of industrial part, etc.) were used to explain these violations, without much success.

Research objective

- Design and adapt a statistical learning model to predict violations and **explain** the variables causing the violations. The end goal is to have diagnotstic/exploration tools for a software.

Data

Notations

- We are looking to N individuals.
- For any $i \in [N]$, X_i is a sequence of Event with their time of size T_i, ex: [(E4, 0 day), (E2, 1 day),(E2, 1.5 day)], $T_i = 3$.
- For any $i \in [N]$, $Y_i \in \{0,1\}^m$ with $m \in \mathbb{N}^*$ (labels related to the violations).

Formal objective

- There are two datasets, a toy's one and a real one.
- Toys dataset: $N > 10^6$ and the violations are linked to subsequence of events (ex: "15-13" in "16-15-13"), $m = 9$.
- Real dataset: $N \approx 10^5$ and the violations are unknown, $m = 5$.

Summary

From the more to the less complex methods:

- BiLSTM-CRF on the whole sequences.
- Explainable One-class classification with the transition profile.
- Logistic Regression on the three first step of the Process.
- Event clustering with Levenshtein distance.

BiLSTM= Bidirectionnal Long Short Term Memory
CRF= Conditional Random Fields

Training BiLSTM-CRF

- Training a CRF parameterized by a BiLSTM neural network. Forward pass with Vertibi algorithm to learn transitions and backward pass by Negative Log likelihood to compute loss.
- Predictor sequence example "4-7-3-6-2-8-1" with a vector target, "0 0 0 0 0 0". The strings are mapped to corresponding number for modeling.

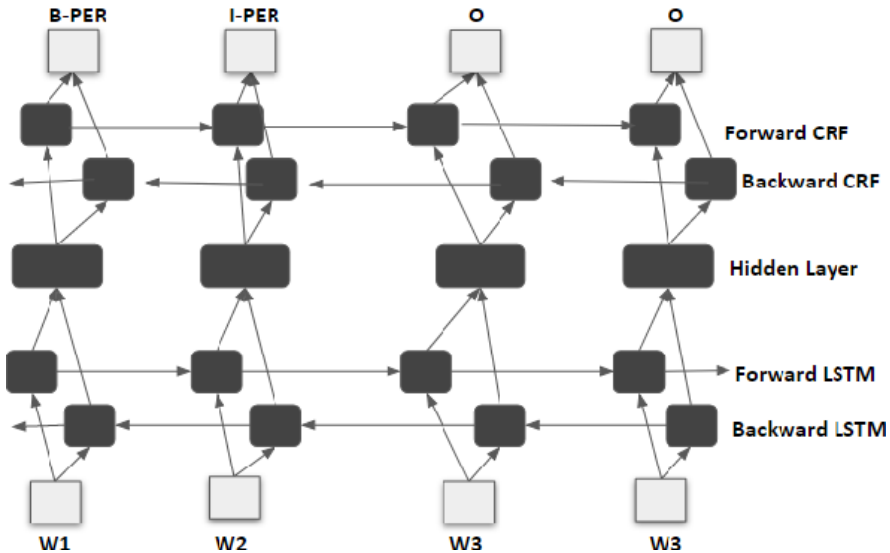


Figure: Illustration of Three Layers of LSTM with CRF head. W1:Event1

	Evaluation metric		
	Precision	Recall	F1-score
Toy violations	0.79	0.78	0.78
Real violations	– 0.99	– 0.99	– 0.99

Table: BiLSTM-CRF. Evaluation metric for Toy and Real datasets after training for 500 training steps on a subsample of the data (1000 data points). The 99% F1-score on the Real dataset is biased towards the majority class with no violations.

Process/EventID 1: 7-3-6-2-8-1

- The order of above event ID or sequence yields the violations 0-0-0-0-0-1.

Figure: Latent representation of the process variables causing the violation. The score of 69.94 indicates the best score in generating the sequence. The red label values on the sequence indicate the contribution individual subprocesses towards predicting the tag sequence $0 - 0 - 0 - 0 - 0 - 1$.

EventID 1: 7-3-6-2-8-1

- The weights of the sentence in the above representation is parameterized by a BiLSTM neural network and rounded to one decimal place.
- The strong purple color is a strong indicator of a violation. Here, the model indicates event 2, with high negative value, as a possible flag of the violation.

Process/EventID 1: 14-10-11-4-13-12-9-7-1-18-17-15-16

- Violations for this sequence is observed for "1_First_appears_as_Seventh"

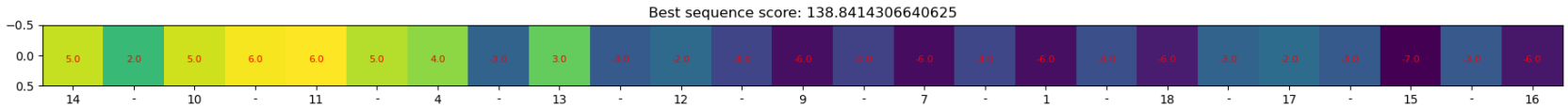


Figure: Latent representation of the process variables causing the violation. The score of 138.84 indicates the best score in generating the sequence. The red label values on the the sequence indicate the contribution or emission weights of the individual subprocesses towards predicting the tag sequence $0-0-0-0-0-0-0-1-0$.

EventID 1: 14-10-11-4-13-12-9-7-1-18-17-15-16

- The weights of the sentence in the above representation is parameterized by a BiLSTM neural network and rounded to one decimal place.
- The strong purple color "maybe" a strong indicator of a a violation. Sequences $9-7-1-18$ with high negative values before a violation is flagged.

Limitation

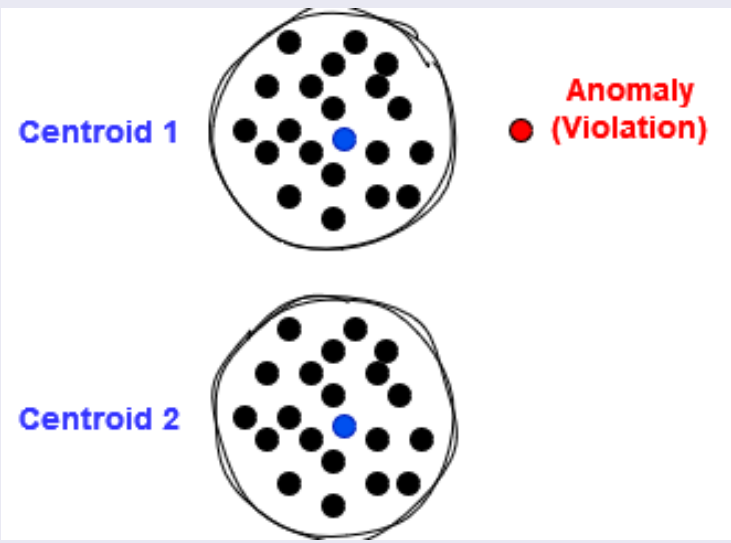
- Training a forward/backward pass of the CRF is time consuming.

One-class classification for detecting and explaining violations

Principle

Violations are rare in nature and their labels are often unavailable. It is then important to develop an explainable anomaly detection model to detect and explain them.
Proposed approach:

- Cluster some violation-free event traces.
- Flag new event traces that are so far from the closest centroid as violations



Issue: Event traces are of different lengths, how to cluster them?

Transition profile

Transition profile is an N × N matrix where N is the number of possible jobs in the process where for each cell i-j we count the number of the occurrence of the sequence (i-j) in the event trace. The matrix is then flattened to form a feature vector.

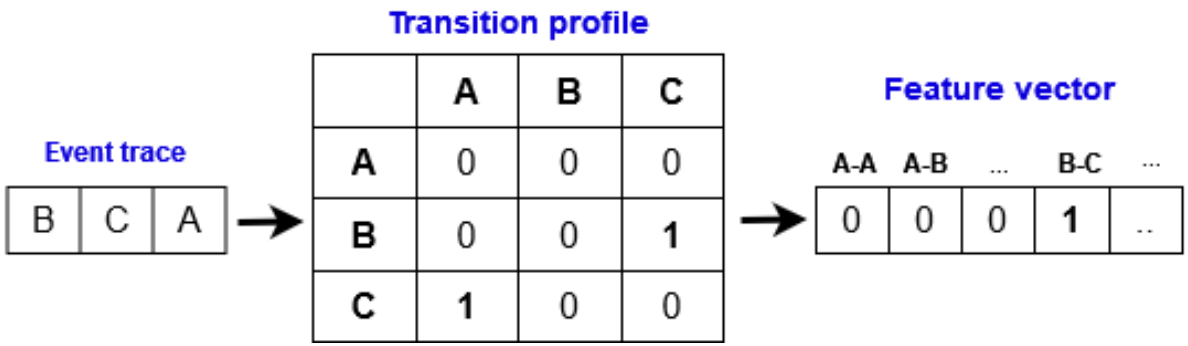


Figure: Transition profile and the feature vector.

The feature vectors are of the same length and ready to use for clustering.

Evaluation

Training

- Training is done by clustering the feature vectors of 100 violation-free event traces.
- At test time, flag new event traces as violation if the distance from the centroid is higher than a threshold.

	Evaluation metric			Prediction time
	Precision	Recall	F1-score	
Real violations data	0.858	0.733	0.766	0.02 seconds

Table: Obtained results.

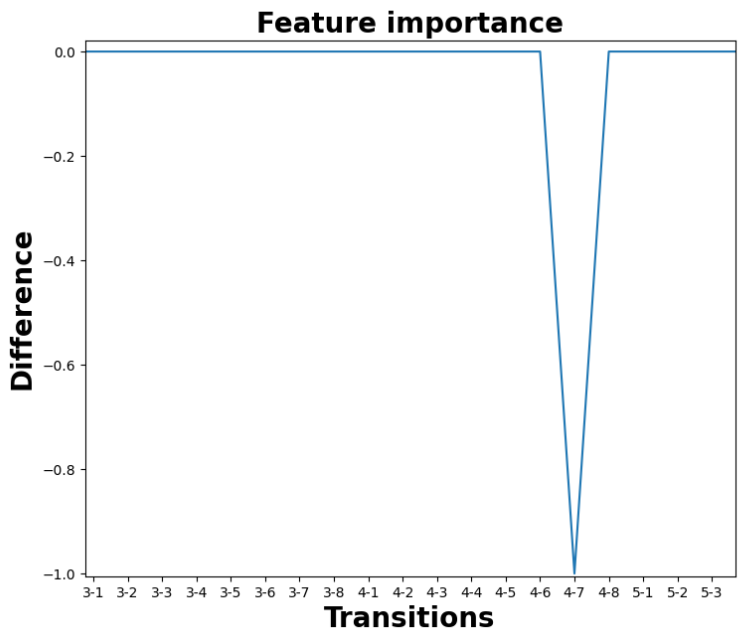
Explainability

Explainability by feature importance

Feature importance is estimated here as the point-wise difference between the transition feature of the abnormal event and the centroid.

Example: for *violation_6*, we get -1 for the transition '4-7'. Interpretation: The sequence '4-7' is missing, which explains the violation.

- Normal event
  trace: [4, 7, 3, 6, 2, 8, 1]
- *violation_6*
  event: [7, 3, 6, 2, 8, 1]



Explainability

Following the proposed approach, here are the explanation for each violation type: **Normal sequence [4-7-3-6-2-8-1]**

Violation number	Explanation	Example
V_1	Normal	[4-7-3-6-2-8-1]
V_2	Normal	[4-7-3-6-2-8-1]
V_3	3-6 missing	[4-6-2-8-1]
V_4	8-2 instead of 2-8	[4-7-3-6-8-2-1]
V_5	6-3 instead of 3-6	[4-7-6-3-2-8-1]
V_6	4-6 missing	[7-3-6-2-8-1]

Table: Obtained explanations for the violations

Violation 1 and Violation 2 are not caused by an abnormal event trace. Other features, such as the mean time between jobs, must be added to the feature vector to detect and explain these violations.

Setup of the Logistic Regression on the real dataset

$X = (X_i[:3])_{i \in [N]}$, we choose the 3 first event with their reltime as input (events are onehotencoded). Y is the label will be the label of each violation.

	Evaluation metric
	F1-score
Y1	0.75
Y2	0.99
Y3	0.99
Y4	0.03
Y5	0.99

Table: F1 score of the Logistic Regression for the different violations. Remark that Y1 without time have F1 score of 0.2

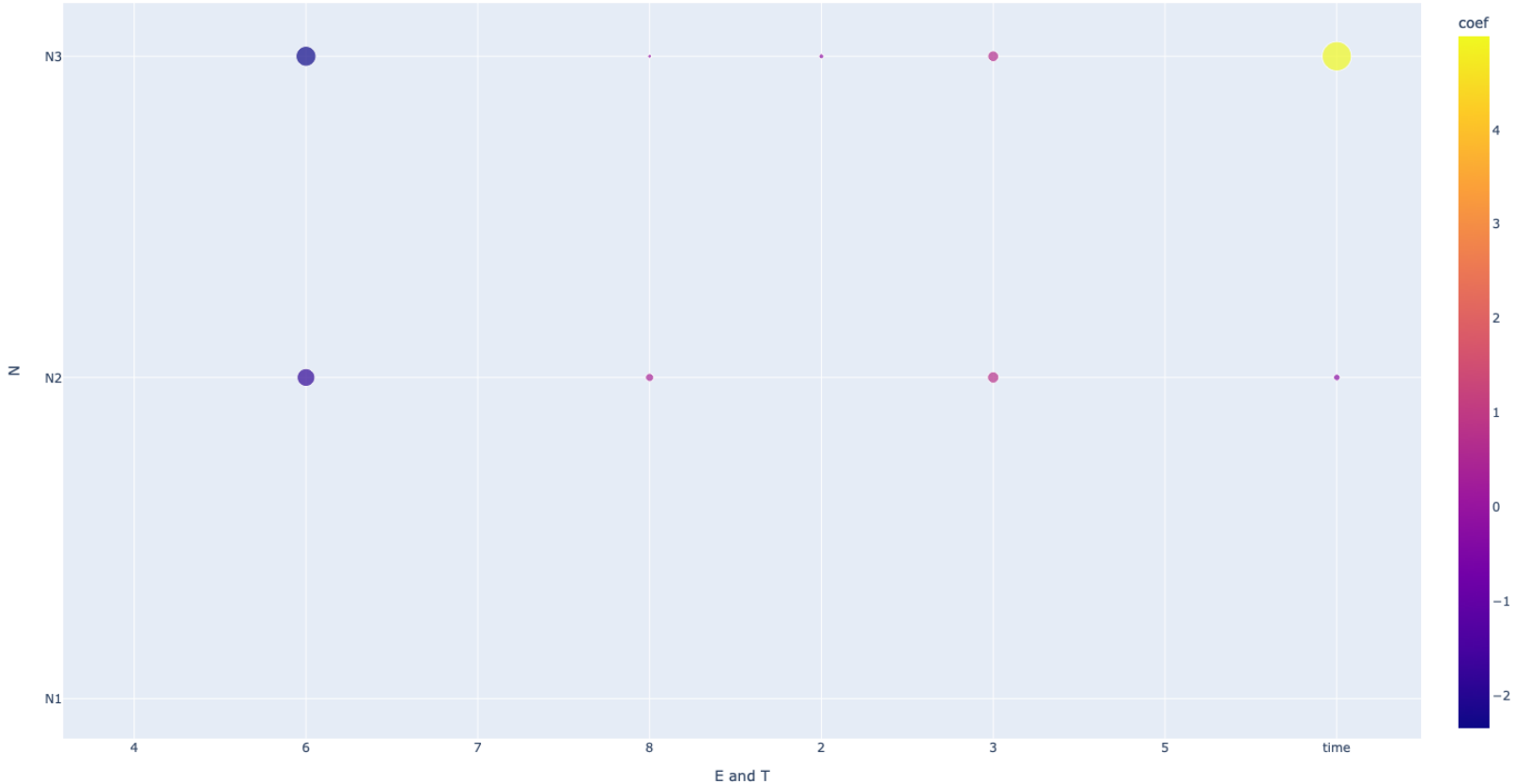


Figure: We choose a L1 penalty to keep only relevant coefficients. The points are bigger according to the norm of the coefficients. The points are dark when their related variables decrease the probability of having $Y = 1$, the logic is reverse for light points.

Clustering sequential data

- The objective is to use a non-supervised approach to identify clusters of events that contain violations.

Methodology

- *K*-**mode** (Mastrogiannis et al., 2009): an approach used to group categorical datasets into homogeneous clusters.
- With similarity measured by the **Levenshtein distance**(Yujian and Bo, 2007).
- Comparison between the partition obtained by k-mode and the types of violations observed by an **adjusted rand index** (Steinley, 2004).

Results

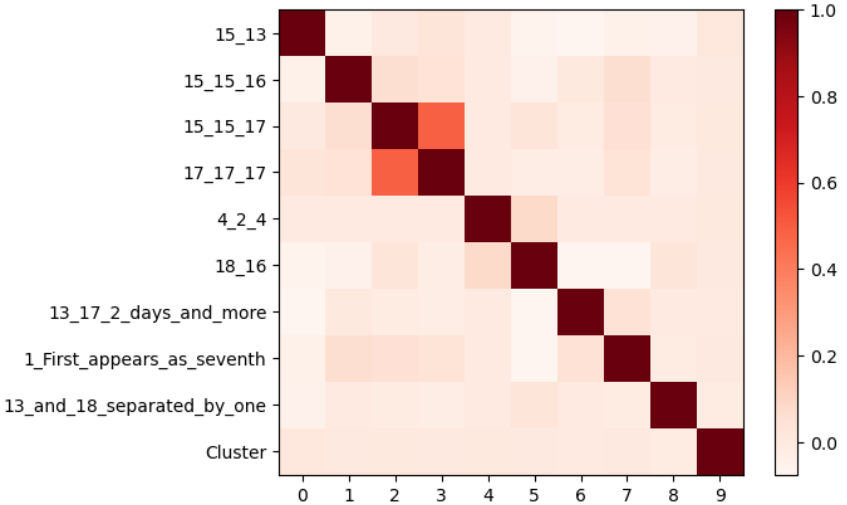


Figure: Similarity between clusters with simulation data.

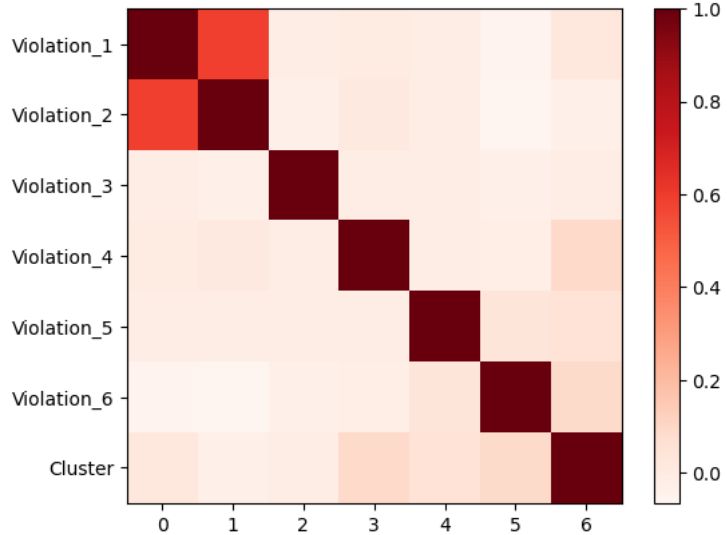


Figure: Similarity between clusters with real data.

Conclusion

Conclusion

- There are severals methods to tackle the problem and we should select it according to what we are looking for and the kind of violation.
- It seems that there are signals both in the ordering of events and their relative times, but expert knowledge are needed to draw robust conclusions.

Thank you

Thank you

Thank you for your attention...

Process/EventID 2: 14-10-9-12-17-15-15-13-16

- Violations for this sequence is observed for "15_15_17" and "1_First_appears_as_Seventh"

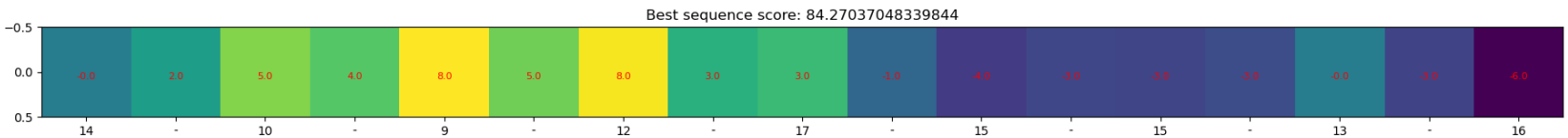


Figure: Latent representation of the process variables causing the violation. The red label values on the sequence indicate the contribution or emission weights of the individual subprocesses towards predicting the tag sequence
$0 - 1 - 0 - 0 - 0 - 0 - 0 - 1 - 0$.

EventID 2: 14-10-9-12-17-15-15-13-16

- The weights on the representation indicates a violation from subsequence "15-15" but especially on the first "15".

Process/EventID 3: 14-10-9-12-17-15-15-13-16

- Violations for this sequence is observed for "15_13"

Figure: Latent representation of the process variables causing the violation. The red label values on the the sequence indicate the contribution or emission weights of the individual subprocesses towards predicting the tag sequence
$1 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0$.

EventID 3: 14-10-9-12-17-15-15-13-16

- The weights on the representation does not visibly indicate a violation before "15_13".
- Since, no early signal is observed for this violation, we can imagine the violation is caused by an unknown or hidden variable.

Process/EventID 4: 14-10-11-4-13-12-9-7-1-18-17-15-16

- Violations for this sequence is observed for "1_First_appears_as_Seventh"

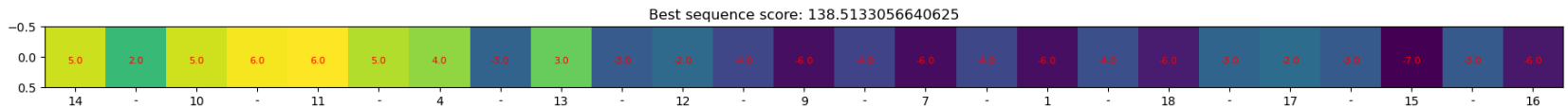


Figure: Latent representation of the process variables causing the violation. The red label values on the the sequence indicate the contribution or emission weights of the individual subprocesses towards predicting the tag sequence
$0 - 0 - 0 - 0 - 0 - 0 - 0 - 1 - 0$.

EventID 4: 14-10-11-4-13-12-9-7-1-18-17-15-16

- The weights on the representation strongly indicate a violation for the subsequence 9-7-1-18. We observe an early signal before the violation is flagged.

Process/EventID 5: 14-10-11-13-12-9-17-15-15-16

- Violations for this sequence is observed for "15_15_16"

Figure: Latent representation of the process variables causing the violation. The red label values on the the sequence indicate the contribution or emission weights of the individual subprocesses towards predicting the tag sequence
$0 - 1 - 0 - 0 - 0 - 0 - 0 - 0 - 0$.

EventID 5: 14-10-11-13-12-9-17-15-15-16

- The weights indicated in the representation does not signal an early anomaly in the subsequences until "15_15".

Process/EventID 6: 14-10-11-12-9-17-15-13-16

- Violations for this sequence is observed for "15_13"

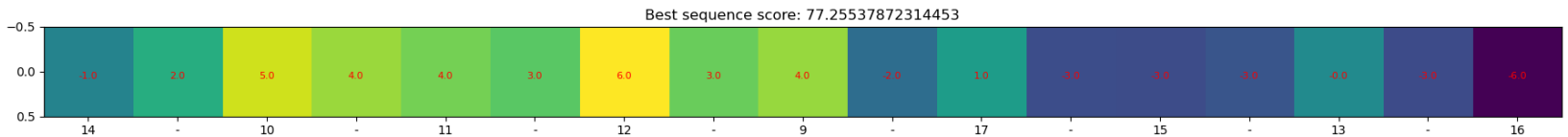


Figure: Latent representation of the process variables causing the violation. The red label values on the the sequence indicate the contribution or emission weights of the individual subprocesses towards predicting the tag sequence
$1 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0$.

EventID 6: 14-10-11-12-9-17-15-13-16

- The weights indicated in the representation does not signal an early anomaly in the subsequences until "15_13".
- However, subsequence "17" is correlated with "13" going by the weights, perhaps it is an anomaly subsequence.