Hugo De Oliveira [1,2], Vincent Augusto [1], Xiaolan Xie [1], Martin Prodel [2], Ludovic Lamarsalle [2]

[1] Mines Saint-Étienne, France; [2] HEVA, France

Data Science Summer School

June, 24th-28th 2019

Process model-based classification for event log data

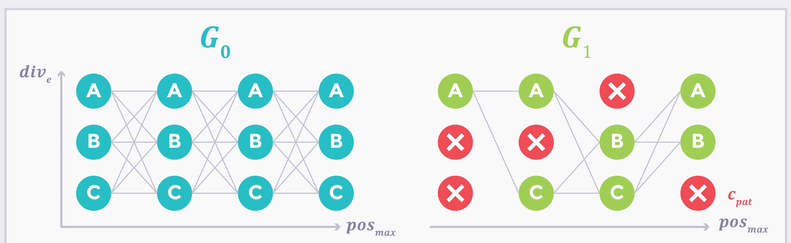
Introduction

Event logs are a widespread type of data structure carrying information of time and ordering of events. As the complexity increases when time-dependent processes are considered, human understanding of predictive models is a lever for acceptability and practical deployment. We present here a new binary classification algorithm, created for event log data. Moreover, the proposed algorithm provides transparency by producing a process model to explain training results and future predictions. Transparency of predictive models is a current challenge, particularly in healthcare where deep learning has become state of the art [1].

Computational experiment

Generation of traces with two graphs G_0 and G_1

After choosing a size configuration ($\boldsymbol{pos}_{max}$ for the length and $\boldsymbol{div}_e$ for the diversity), we create two graphs $\boldsymbol{G}_0$ and $\boldsymbol{G}_1$ with $\boldsymbol{pos}_{max}$ identical layers composed of $\boldsymbol{div}_e$ different nodes. Then, a proportion $\boldsymbol{c}_{pat}$ of shared patterns in $\boldsymbol{G}_1$ is deleted. Traces are then created by randomly crossing the graphs, forming event logs $\boldsymbol{L}_0$ and $\boldsymbol{L}_1$ (10 per parameters combination).



$(\boldsymbol{div}_e; \boldsymbol{pos}_{max}; \boldsymbol{c}_{pat})$

Event logs

80% $\boldsymbol{L}_0^{train}\ \boldsymbol{L}_1^{train}$ — 20% $\boldsymbol{L}_0^{test}\ \boldsymbol{L}_1^{test}$

Training

DT*, RF**, MLP***
with event log flattening
and hyperparameters tuning

RepOpt and ***DiffOpt***

Testing

Qualitative results

Process models

Quantitative results

AUC****

Proposed methodology

Presentation

The problem addressed here is the training of a binary classifier on event log data of two classes $\boldsymbol{L}^{train}=(\boldsymbol{L}_0^{train}, \boldsymbol{L}_1^{train})$ of class 0 and 1.

The replayability $\boldsymbol{R}(\boldsymbol{PsM}, \boldsymbol{\sigma})\in[\boldsymbol{0}, \boldsymbol{1}]$ is a measure which quantifies the ability of a process model $\boldsymbol{PsM}$ to represent a trace [2]. The idea behind the proposed framework is the construction of a process model which well represents traces from event log $\boldsymbol{L}_1^{train}$ while less representing traces from $\boldsymbol{L}_0^{train}$. Thus, it extracts discriminative patterns from $\boldsymbol{L}_1^{train}$.

Two optimization functions ***RepOpt*** and ***DiffOpt*** can be used to construct the model, with a metaheuristic to find optimal process model.

RepOpt	$\max_{PsM} mean(R_1^{train})$
DiffOpt	$\max_{PsM} mean(R_1^{train}) - mean(R_0^{train})$

Event log and process model definitions

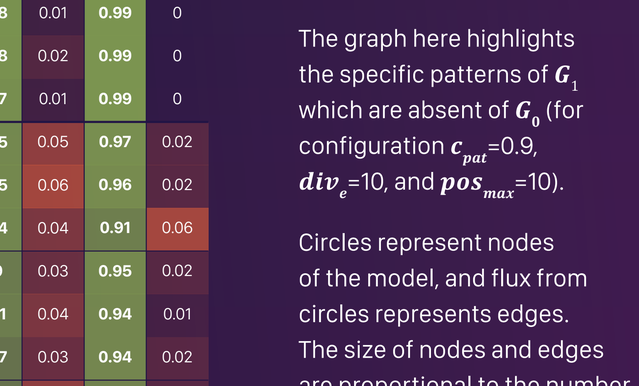
An **event log** $\boldsymbol{L}=\{\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2..., \boldsymbol{\sigma}_n\}$ regroups data in traces, each **trace** $\boldsymbol{\sigma}_l = \boldsymbol{e}_1\ \boldsymbol{e}_2...\boldsymbol{e}_m$ being an ordered list of events, each **event** $\boldsymbol{e}_k=(\boldsymbol{a}, \boldsymbol{t})$ having a **label** $\boldsymbol{a}\in\boldsymbol{A}$ and a **time-stamp t**.

A **process model** $\boldsymbol{PsM}=(\boldsymbol{N}, \boldsymbol{E}, \boldsymbol{L}, \boldsymbol{P})$ is defined as a four-tuple with a set of **nodes** $\boldsymbol{N}$ and **edges** $\boldsymbol{E}$, a **label function** $\boldsymbol{L}$ and a **position function** $\boldsymbol{P}$. $\boldsymbol{L}$ and $\boldsymbol{P}$ map each node $\boldsymbol{n}\in\boldsymbol{N}$ respectively to a label $\boldsymbol{a}\in\boldsymbol{A}$ and a position $\boldsymbol{p}\in\mathbb{N}^*$. Each edge links a node to another one of **strictly higher position.**

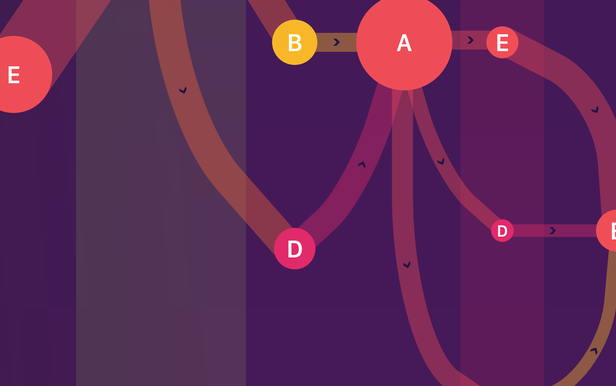
The procedure of class prediction of a trace σ

1 Train *PsM* using *RepOpt* or *DiffOpt*
on $\boldsymbol{L}^{train}=(\boldsymbol{L}_0^{train}, \boldsymbol{L}_1^{train})$ to obtain replayability distributions $\boldsymbol{R}_0^{train}=\boldsymbol{R}(\boldsymbol{PsM},\boldsymbol{L}_0^{train})$ and $\boldsymbol{R}_1^{train}=\boldsymbol{R}(\boldsymbol{PsM},\boldsymbol{L}_1^{train})$

Overlapped replayability distributions



Distinct after optimization



2 Set a threshold of separation
$\boldsymbol{\theta}=\boldsymbol{\theta}(\boldsymbol{R}_0^{train}, \boldsymbol{R}_1^{train})$, e.g. using Gini impurity

3 Compute the replayability
$\boldsymbol{r}_\sigma= \boldsymbol{R}(\boldsymbol{PsM}, \boldsymbol{\sigma})$

4 Predict the class of the trace σ
by comparison between $\boldsymbol{r}_\sigma$ and $\boldsymbol{\theta}$

Quantitative results

Results show that the proposed method using the optimization equation ***DiffOpt*** generally outperforms ***RepOpt*** and other methods.

Moreover, the AUC**** variability on the 10 replications per parameter combinations is lower for ***DiffOpt***.

The performance gaps between ***DiffOpt*** and other methods, as well as performance variabilities increase with event logs complexity (from top to bottom in the table).

c_{pat}	div_e	p_{max}	DT* AVG	DT* STD	RF** AVG	RF** STD	MLP*** AVG	MLP*** STD	*RepOpt* AVG	*RepOpt* STD	*DiffOpt* AVG	*DiffOpt* STD
0.9	10	10	**0.96**	0.01	**0.96**	0.01	**0.99**	0.01	**0.99**	0.01	**1**	0
		25	**0.95**	0.01	**0.95**	0.01	**1**	0	**0.99**	0.02	**0.99**	0.02
		50	**0.96**	0.01	**0.96**	0.01	**1**	0	**0.99**	0.02	**0.98**	0.01
	50	10	**0.95**	0.02	**0.95**	0.02	**0.97**	0.02	**0.98**	0.01	**1**	0
		25	**0.95**	0.02	**0.95**	0.02	**0.97**	0.01	**0.97**	0.01	**0.99**	0
		50	**0.95**	0.03	**0.95**	0.03	**0.98**	0.02	**0.97**	0.01	**0.99**	0
	100	10	**0.95**	0.01	**0.95**	0.01	**0.96**	0.01	**0.98**	0.01	**0.99**	0
		25	**0.92**	0.05	**0.92**	0.05	**0.97**	0.01	**0.98**	0.02	**0.99**	0
		50	**0.9**	0.07	**0.9**	0.07	**0.97**	0.02	**0.97**	0.01	**0.99**	0
0.75	10	10	**0.88**	0.05	**0.88**	0.05	**0.94**	0.03	**0.95**	0.05	**0.97**	0.02
		25	**0.89**	0.04	**0.9**	0.04	**0.95**	0.04	**0.95**	0.06	**0.96**	0.02
		50	**0.85**	0.06	**0.86**	0.06	**0.93**	0.04	**0.94**	0.04	**0.91**	0.06
	50	10	**0.88**	0.03	**0.88**	0.03	**0.86**	0.06	**0.9**	0.03	**0.95**	0.02
		25	**0.87**	0.04	**0.85**	0.06	**0.87**	0.05	**0.91**	0.04	**0.94**	0.01
		50	**0.88**	0.03	**0.86**	0.06	**0.85**	0.08	**0.87**	0.03	**0.94**	0.02
	100	10	**0.77**	0.1	**0.78**	0.06	**0.86**	0.03	**0.87**	0.05	**0.93**	0.02
		25	**0.65**	0.06	**0.64**	0.07	**0.8**	0.11	**0.81**	0.04	**0.92**	0.02
		50	**0.64**	0.07	**0.64**	0.06	**0.85**	0.05	**0.72**	0.05	**0.86**	0.07

Qualitative results

A process model is used to discriminate the two classes using replayability. As a result, discriminative patterns extracted during the training procedure can be visualized.

The graph here highlights the specific patterns of $\boldsymbol{G}_1$ which are absent of $\boldsymbol{G}_0$ (for configuration $\boldsymbol{c}_{pat}$=0.9, $\boldsymbol{div}_e$=10, and $\boldsymbol{pos}_{max}$=10).

Circles represent nodes of the model, and flux from circles represents edges. The size of nodes and edges are proportional to the number of traces represented.

Conclusion

Across this study, we proposed a new binary classification algorithm for event log data, based on process model optimization. Quantitative and qualitative results show the competitiveness and the transparency of the method. Future research will focus on the integration of more information into the model to increase prediction performance for other types of discriminative patterns.

HEVA — MINES Saint-Étienne — DS3 DATA SCIENCE SUMMER SCHOOL

* DT: Decision Tree; **RF: Random Forest; ***MLP: Multi Layer Perceptron; ****AUC: Area Under the Curve;

References

[1] C. Xiao, E. Choi, and J. Sun. Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review. Journal of the American Medical Informatics Association, 06 2018.

[2] M. Prodel, V. Augusto, B. Jouaneton, L. Lamarsalle, and X. Xie. Optimal process mining for large and complex event logs. IEEE Transactions on Automation Science and Engineering, 2018.