KU LEUVEN

Machine to mine

Unearthing Insights in Predictive Process Monitoring

Guest lecture Advanced Process Mining, RWTH Aachen

Dr. Jari Peeperkorn

Research Center for Information Systems Engineering (LIRIS), Faculty of Economics and Business

About me

- *Msc. Astronomy & Astrophysics (2019)*
- PhD in Business Economics (2023)
  - *Novel Conformance Checking Methods and Validation Strategies for Deep Learning in Process Mining*

- Postdoctoral Researcher (2023-…)
  - 2023-2024: Project on process model forecasting
  - 2024-*2027*: FWO grant *“Robust Multi-Modal Prediction in Business Processes”*

- Research expertise
  - Process mining
  - Predictive process monitoring
  - Machine learning

- https://jaripeeperkorn.github.io/

Agenda

- Addressing (unsolved) questions for current researchers to enhance predictive process monitoring adoption (PPM)
  1. How to adapt ML for process data?
  2. How to properly evaluate PPM?
  3. How do ML/DL models learn process model structure?
  4. How to develop explainable, robust, and reliable models?
  5. How to incorporate the inter-case perspective?
  6. How to go from case-level to model-level predictions? (*micro to macro*)

Bias → focus on research done at my research group

Adapting ML for process data

Machine learning on process data

- Why machine/deep learning?
  - Results on other types of data/problems
  - Research shows it to outperform process discovery + simulation on most predictive tasks
  - Easier to add new event or case features
    - No need to explicitly program this
    - Less bias?
  - Speed, efficiency, scalability
  - …

Machine learning on process data

- Often: take technique that has been successful in field x(computer vision, NLP, …)
  → apply on process data

- However…
  - Vocabulary sizes different
    - *Few activity labels in PM vs a lot of words in NLP*
  - Data distributions?
    - Process data often not normally distributed
  - Concurrency/parallelism?
    - When you don't explicitly model concurrent behavior → You need lots of data for it to not seem *random*
  - …

P. Ceravolo, S. B. Junior, E. Damiani and W. Van Der Aalst, "Tuning Machine Learning to Address Process Mining Requirements," in *IEEE Access*, vol. 12, pp. 24583-24595, 2024

Solutions?

- Smarter preprocessing?
  - Process data specific features
    - Trace encoding in process mining: A survey and benchmarking. GM. Tavares, RS Oyamada, S Barbon Jr., P Ceravolo. EAAI, 2023.
    - Empowering Predictive Process Monitoring through time-related Features. RS Oyamada, GM Tavares, S Barbon Jr., P Ceravolo. CAiSE, 2024.
    - …

    → Works show that doing feature creation with simple machine learning is often on par with deep learning!

    → And faster
  - Not a lot of research done → open problem!

- Adapted architectures?
  - Tabular methods or deep learning
  - Not a lot of work done on this yet!

Seq2Seq-LSTMs

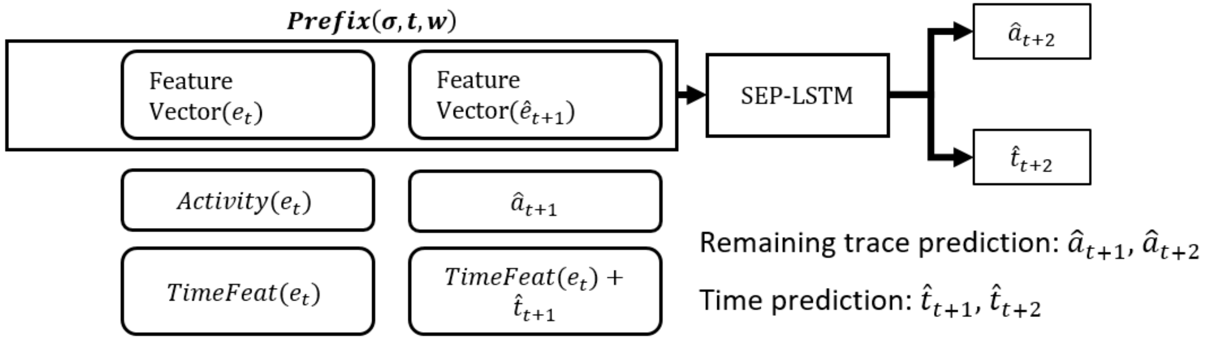
- Motivation: Processing of luggage at Brussels Airport, trajectory and execution times of luggage influenced by:
  - Case features: Arrival/Departing flight (e.g. Flight ID, Airport)
  - Time features: Time information (e.g. hour of day, remaining time until scheduled departure)
  - Event features: Information on concurrently running cases (e.g. load at Check-in/Screening/Sorter)

- Research done at our group:
  - Develop a model architecture that can natively incorporate all relevant information in order to predict the remaining **suffixes and runtimes** of bags processed at the airport

Gunnarsson, Björn & vanden Broucke, Seppe & Weerdt, Jochen. (2023). A Direct Data Aware LSTM Neural Network Architecture for Complete Remaining Trace and Runtime Prediction. IEEE Transactions on Services Computing. PP. 1-13. 10.1109/TSC.2023.3245726.

Classical SEP-LSTMs (single event prediction)

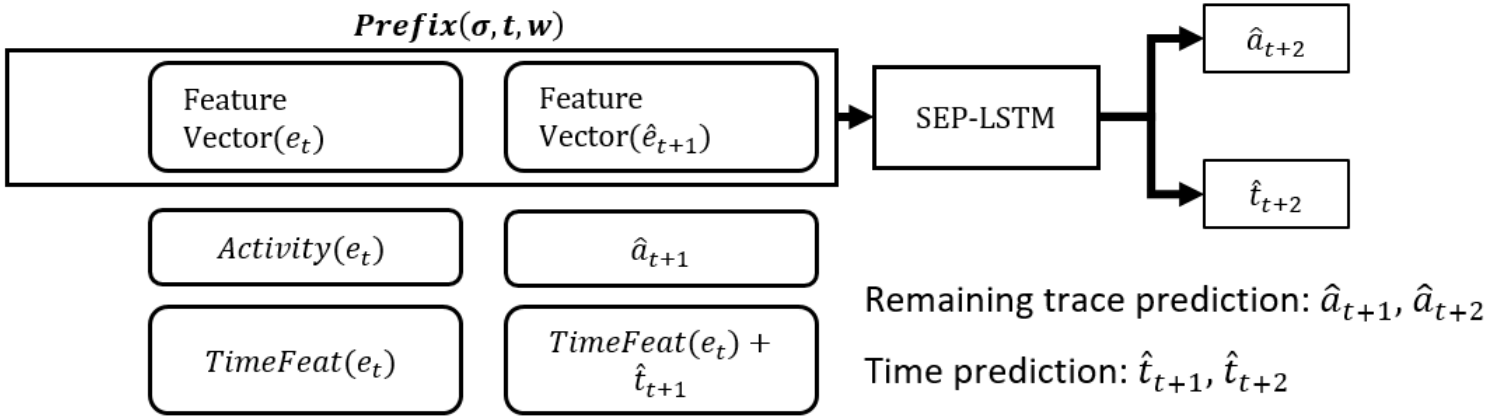
- Main model components
  - Prefix of feature vectors (running case)
  - Propagated into a shared LSTM layer
  - Shared LSTM layer returns a sequence of hidden states



- Hidden states used to obtain separate predictions:
  - Next activity label
  - Time which will elapse until next event is observed

Suffix prediction

- Predicted activity label and elapsed time is utilised to construct a new event
- Predicted event is appended to the original prefix
- Which in turn is fed into the SEP-LSTM model in order to obtain new predictions
  - Makes sense and works well in e.g. NLP
  - **You can not properly use input features if you are not predicting them!**
  - **If you go wrong you stay wrong (hallucination) → less robust**

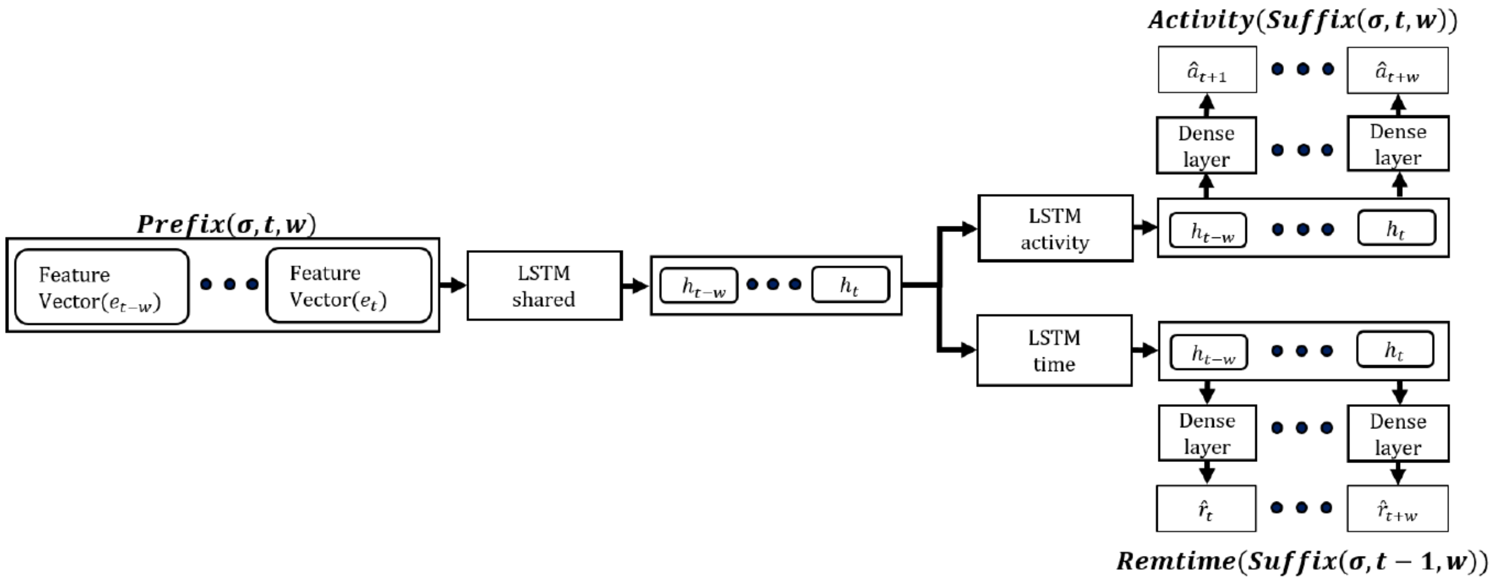


CRTP-LSTM model

- The **CRTP-LSTM model** is trained to directly predict the full remaining trace and a sequence of remaining runtimes

- The pivotal advantage of CRTP-LSTM is that it only relies on previously observed events for prediction without needing to rely on hallucination
  - This makes the architecture more robust in terms of utilizing all available features relating to previously observed events
  - Natively data-aware

- Alternatives
  - Attention? **Transformers**?
  - Graph neural networks?



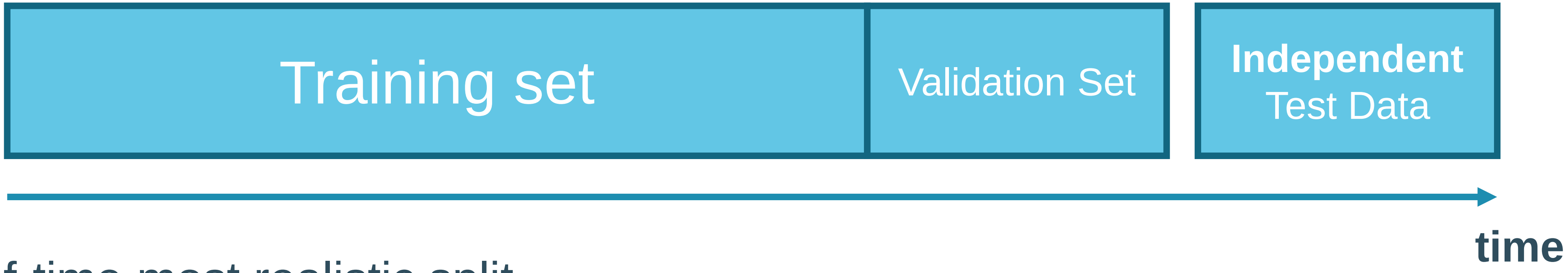
Gunnarsson, Björn & vanden Broucke, Seppe & Weerdt, Jochen. (2023). A Direct Data Aware LSTM Neural Network Architecture for Complete Remaining Trace and Runtime Prediction. IEEE Transactions on Services Computing. PP. 1-13. 10.1109/TSC.2023.3245726.

Predictive process monitoring evaluation

Train – test splits

- Training Set → Train model's weights
  - Validation Set → Determine best hyperparameters

  (can split this however you want, but best to have this as independent as possible)
- Test Set (for research & development) → Determine the model's accuracy/quality
  - Needs to be independent for proper testing



- Out-of-time most realistic split
  - Closest to how it will be done at deployment
  - But how to do this properly with process data?

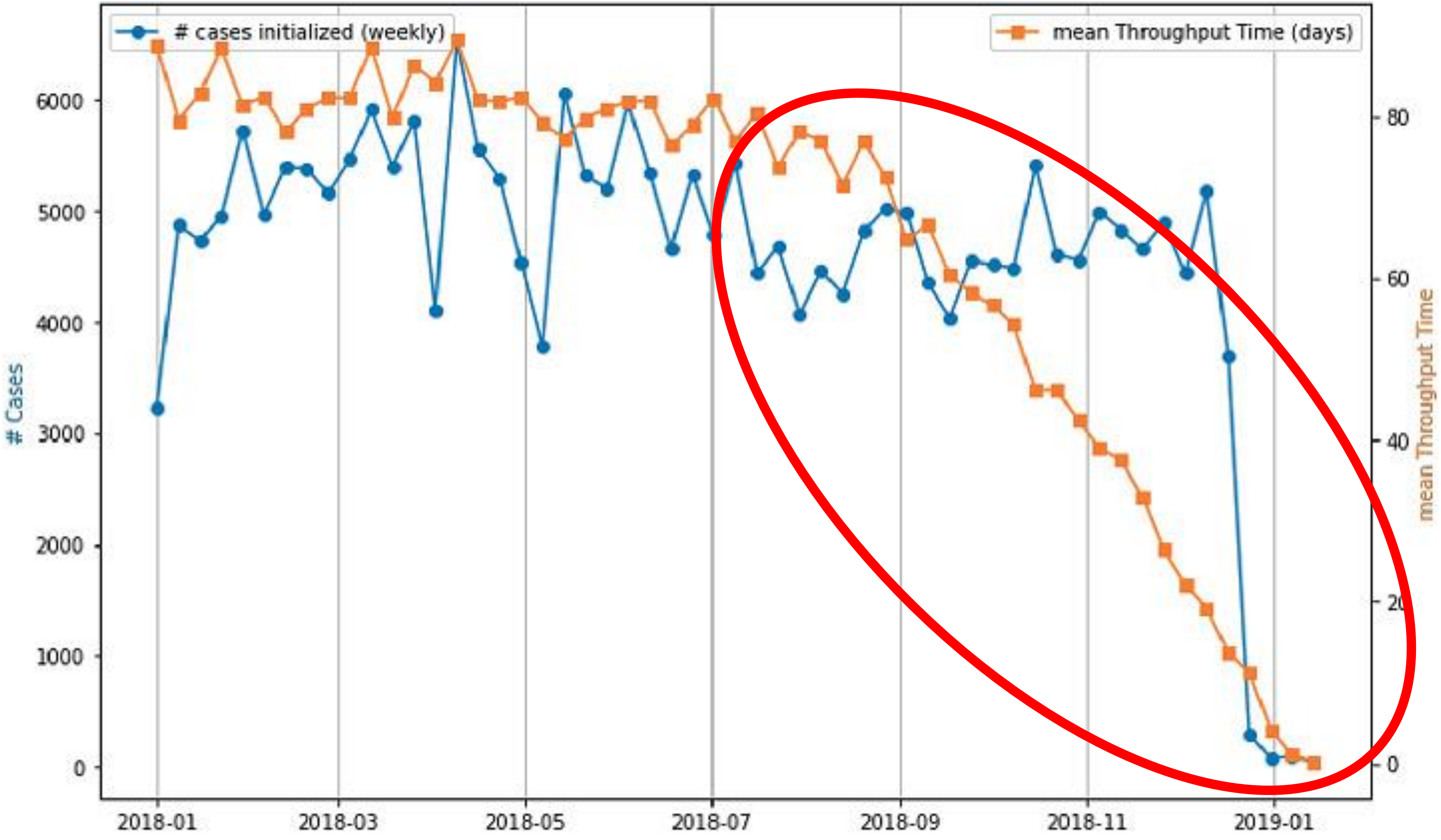
Widespread bad practices in PPM evaluation

- Train 80%, validation 16%, test 4%
- Evaluate the model for different prefix lengths, then average those results
- Compare with previously published results using totally different setups
- Random train-test split for outcome or remaining time prediction
- Random k-fold cross-validation for outcome or remaining time prediction
- Test set overuse: too many models tested on the same test set
- Example leakage* and other data leakage

*Abb, L., Pfeiffer, P., Fettke, P., & Rehse, J. R. (2023). A Discussion on Generalization in Next-Activity Prediction. *arXiv preprint arXiv:2309.09618*.

Another effect…

DyLoPro



Over weeks

(lots of cases that run for multiple weeks)

#cases

mean throughput time

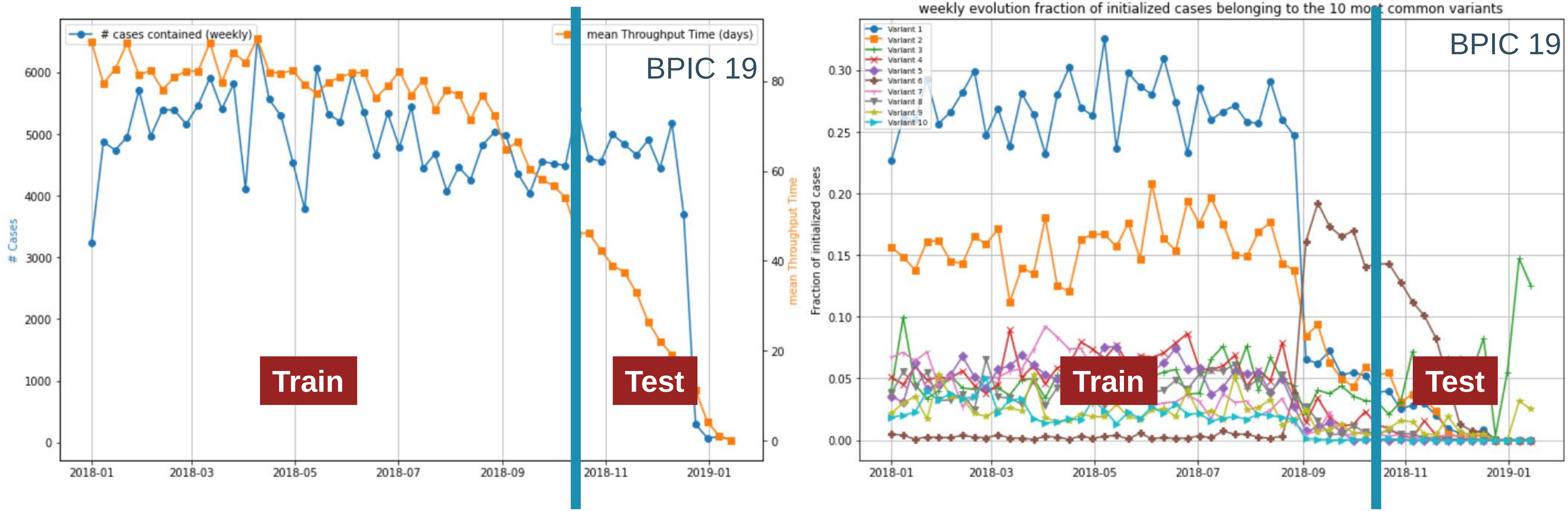
"Extraction bias"

Sort of virtual concept drift

Failing to factor in the dynamics of event logs

DyLoPro

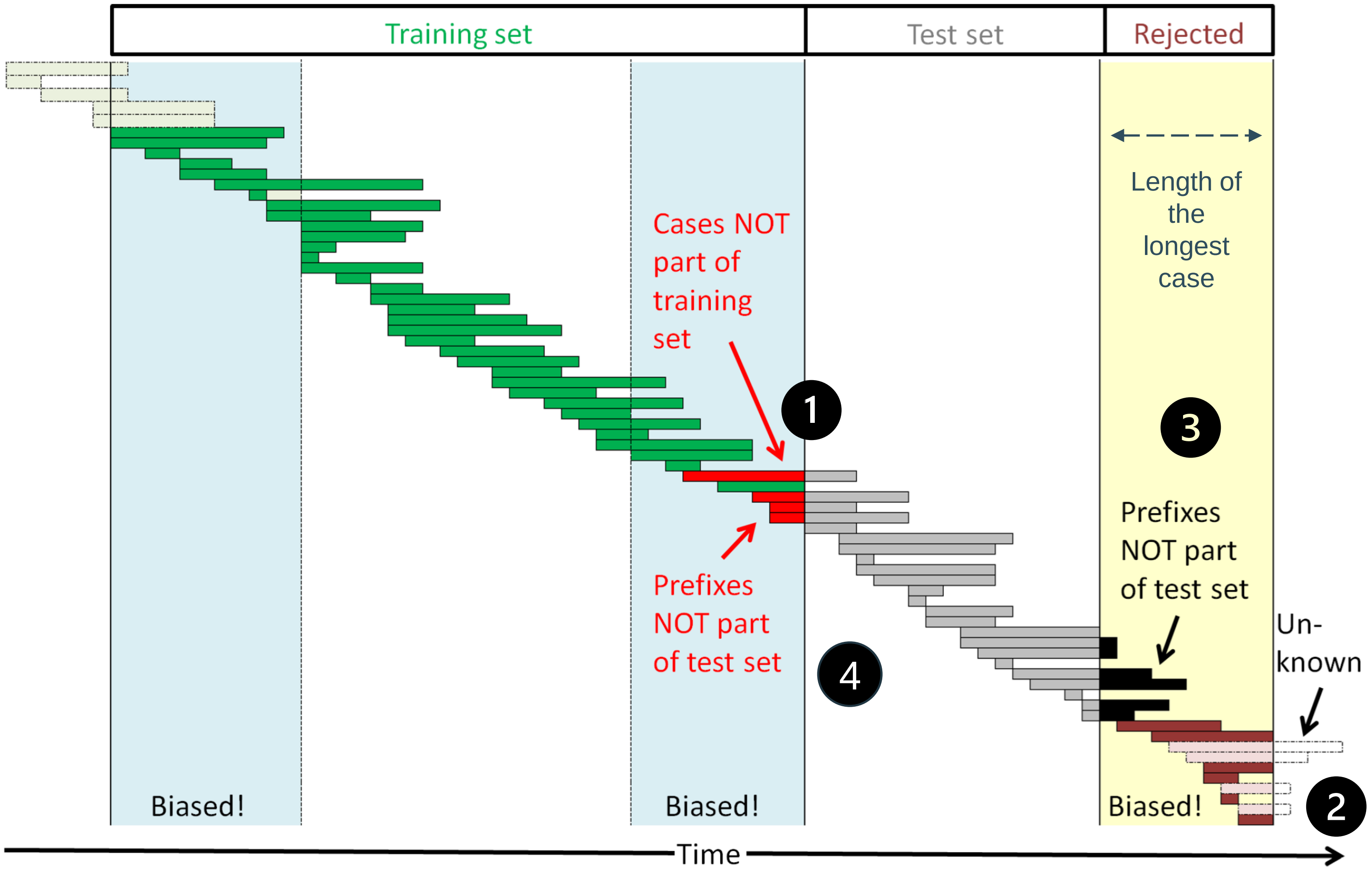
→ Test set is not a good representation of that process!



Wuyts, B., Weytjens, H., vanden Broucke, S., and De Weerdt, J. (2023) DyLoPro: Profiling the Dynamics of Event Logs. BPM 2023, 146–162.

Removing test set bias

(extraction bias)



1. (Different) prefixes but obtained from the same traces should not be part of both training and testing set – “strict temporal splitting”

2. Cases for which we don’t observe the outcome (unknown), should not be in the test set (and are often not part of the dataset in general)
→ In public data set often done already

3. + 4. This causes two types bias: the number of running cases and their average length no longer reflect the underlying reality (e.g. inter-case variables)
→ Remove the black prefixes from test set
→ Grey prefixes of the red-gray cases should be included in the test set

(restore balance in #cases running and case lengths)

Weytjens, H., De Weerdt, J. (2022). Creating Unbiased Public Benchmark Datasets with Data Leakage Prevention for Predictive Process Monitoring. In: A. Marrella, B. Weber (Eds.), *Business Process Management Workshops, BPM 2021*: *vol. 436*, (18-29). doi: 10.1007/978-3-030-94343-1_2

Out-of-time cross-validation might become difficult

Weytjens, H., De Weerdt, J. (2022). Creating Unbiased Public Benchmark Datasets with Data Leakage Prevention for Predictive Process Monitoring. In: A. Marrella, B. Weber (Eds.), *Business Process Management Workshops, BPM 2021*: *vol. 436*, (18-29). doi: 10.1007/978-3-030-94343-1_2

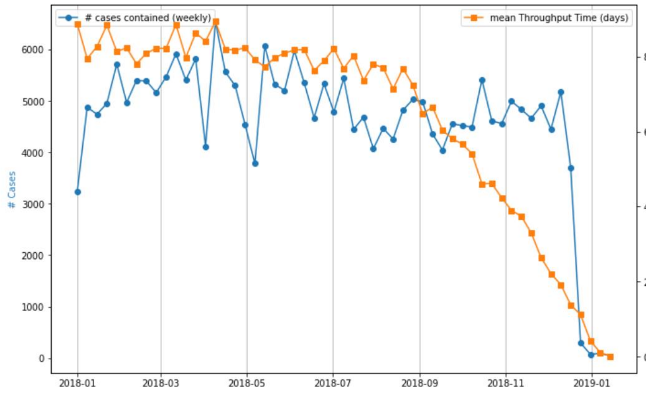
KU LEUVEN

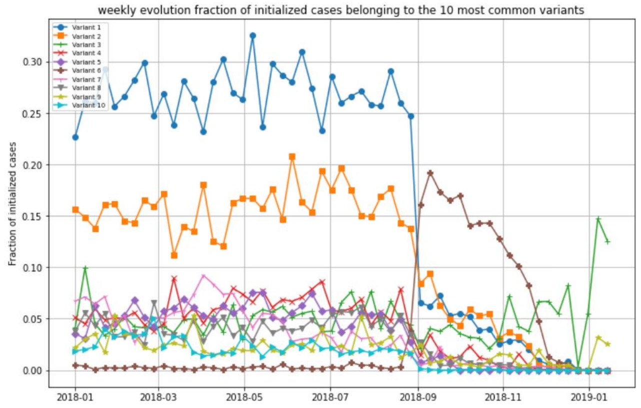
PPM evaluation: Key takeaways

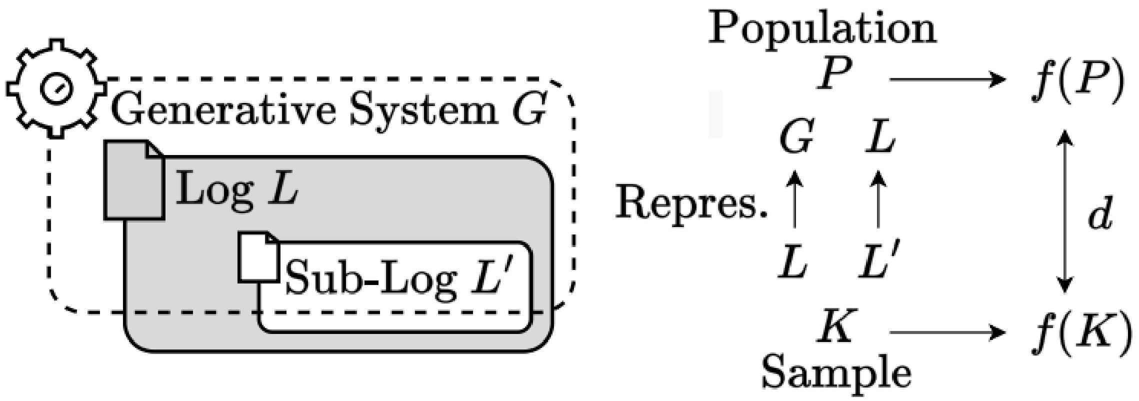
- With training data, you can (intrinsically) do whatever you like
- Test data should be created rigorously
  - Constructing benchmark datasets with fixed test sets, and/or apply best practices
  - Public data sets should become available with masked test data, as done in other ML domains
  - We need better scientific recognition of making PPM datasets available
- Assuming steady-state is naïve at best
  - Consider the deployment setting: what will your models do in a real-life environment?
  - Creating “hard” test data is what we should aim for
- Of course, you cannot assume that models can “learn” concept drift, however, dealing with concept drift is also a main task for model monitoring during deployment (MLOps)
  - Measuring model stability?

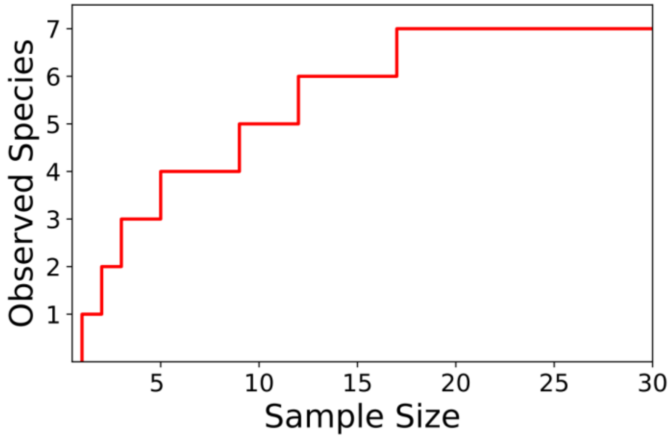
Not always straightforward…

- Concept drift? Extraction bias?
- Model stability? Generalization?
- Log completeness?





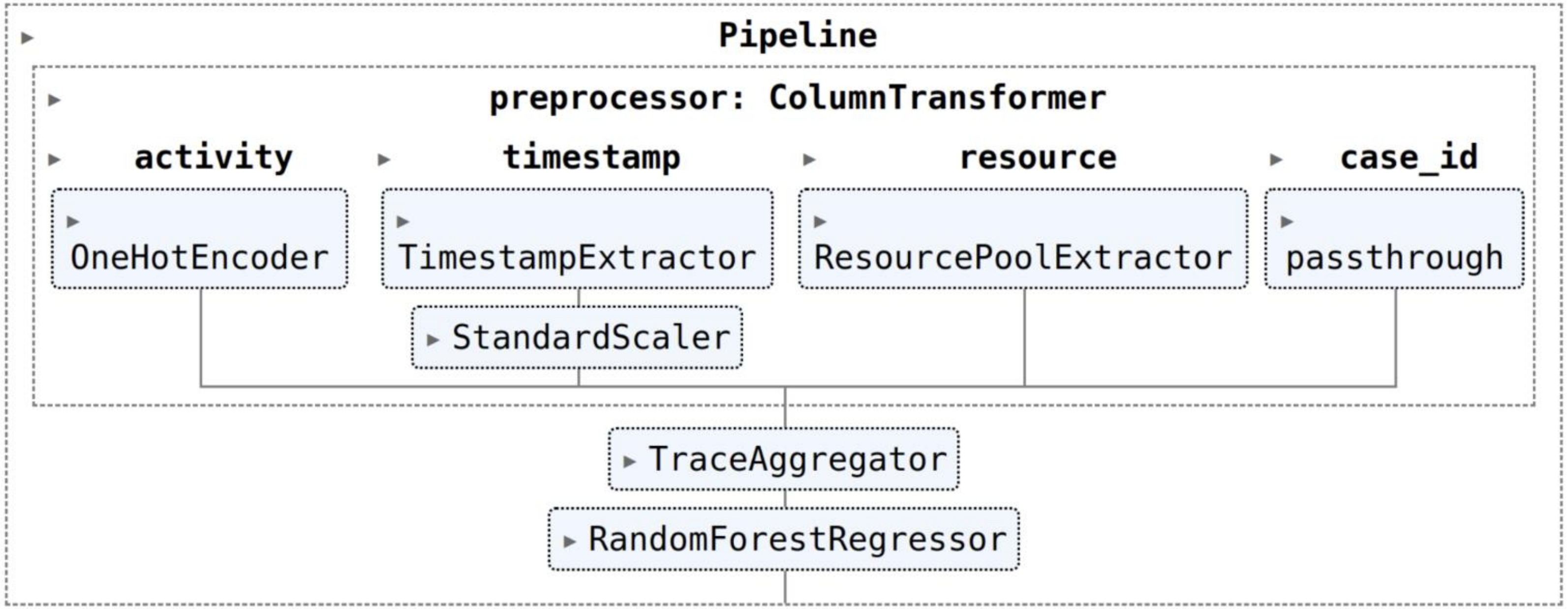




M. Kabierski, M. Richter and M. Weidlich, "Addressing the Log Representativeness Problem using Species Discovery," 2023 5th International Conference on Process Mining (ICPM), Rome, Italy, 2023

SKPM: an extension of the popular scikit-learn library

- Reproducibility in PM is a big issue
- Researchers are, first of all, researchers!



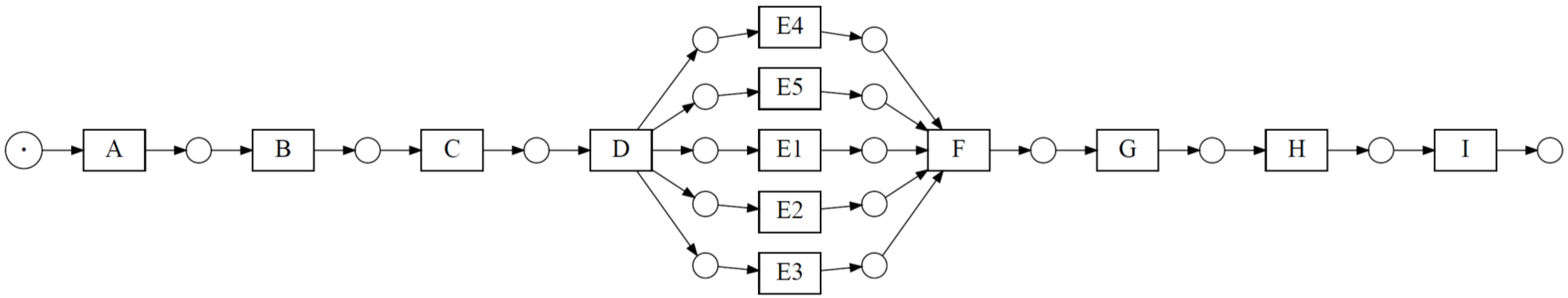
A Scikit-learn Extension Dedicated to Process Mining Purposes (DEMO Paper). RS Oyamada, GM Tavares, S Barbon Jr., P Ceravolo. CoopIS, 2023.

How do ML/DL models learn process model structure?

Generalization

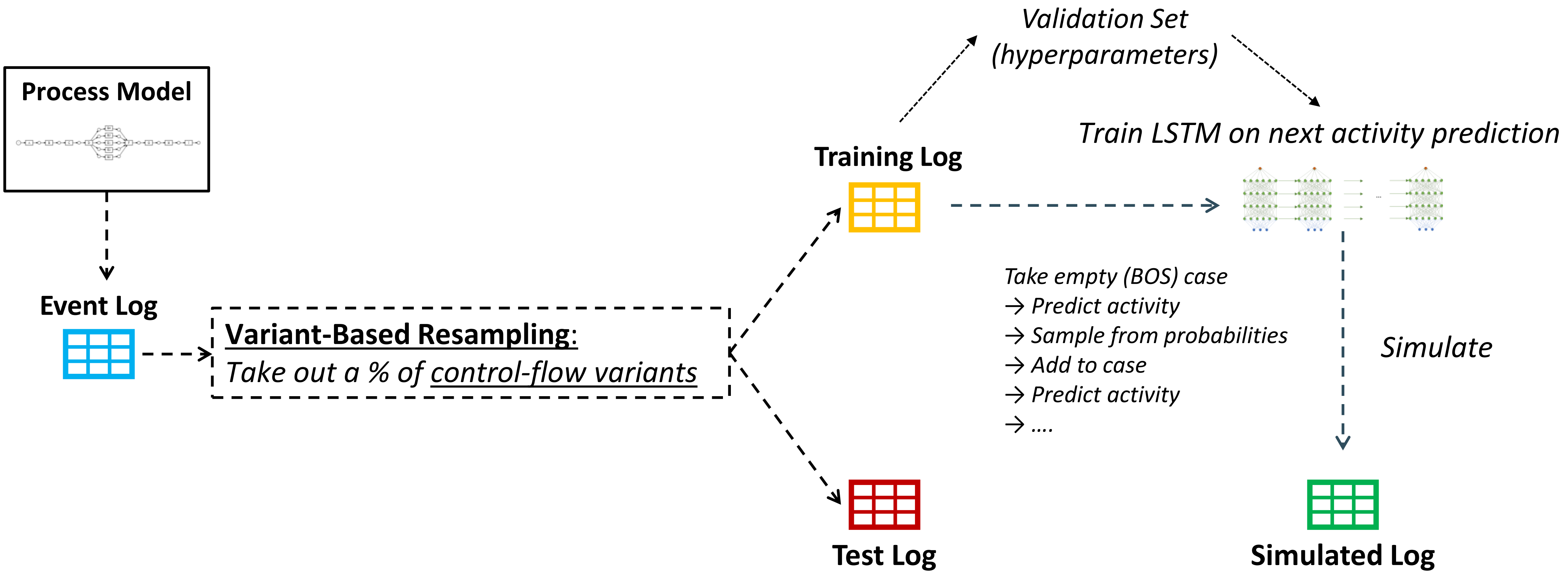
- Key question: How capable are deep learning models to generalize process behavior?

- Imagine, process discovery:
  - You provide an algorithm with a simulated event log based on the model below
  - You remove one of the 120 possible variants
  - Would you expect a process discovery technique to fail to detect the parallel construct?



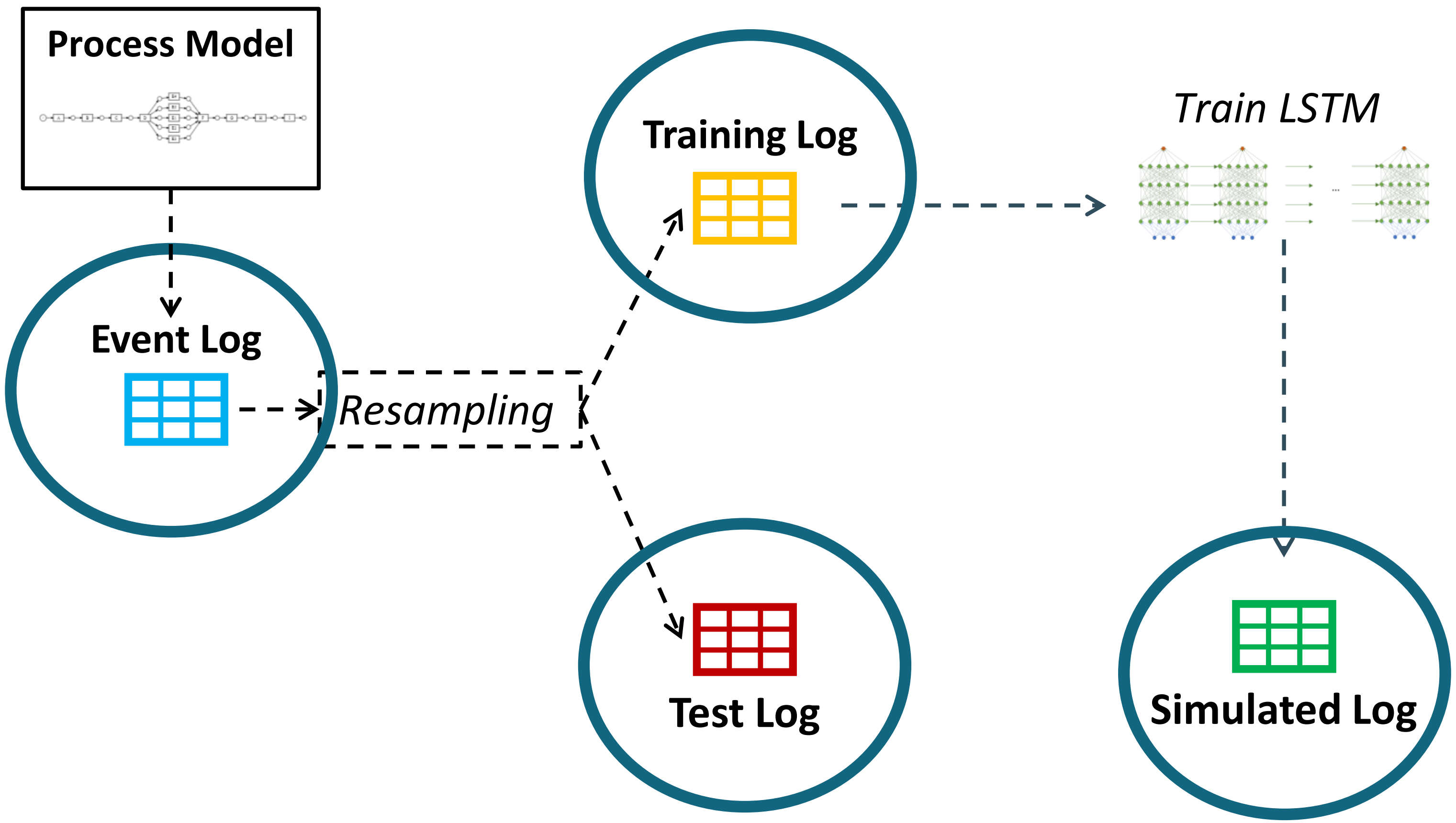
Peeperkorn, J., vanden Broucke, S., De Weerdt, J. (2022). Can recurrent neural networks learn process model structure? Journal Of Intelligent Information Systems. doi: 10.1007/s10844-022-00765-x

Conformance checking-like measures for PPM



Peeperkorn, J., vanden Broucke, S., De Weerdt, J. (2022). Can recurrent neural networks learn process model structure? Journal Of Intelligent Information Systems. doi: 10.1007/s10844-022-00765-x

Conformance checking-like measures for PPM



Training Log Variants ⊆ Simulated Log
→ Fitness

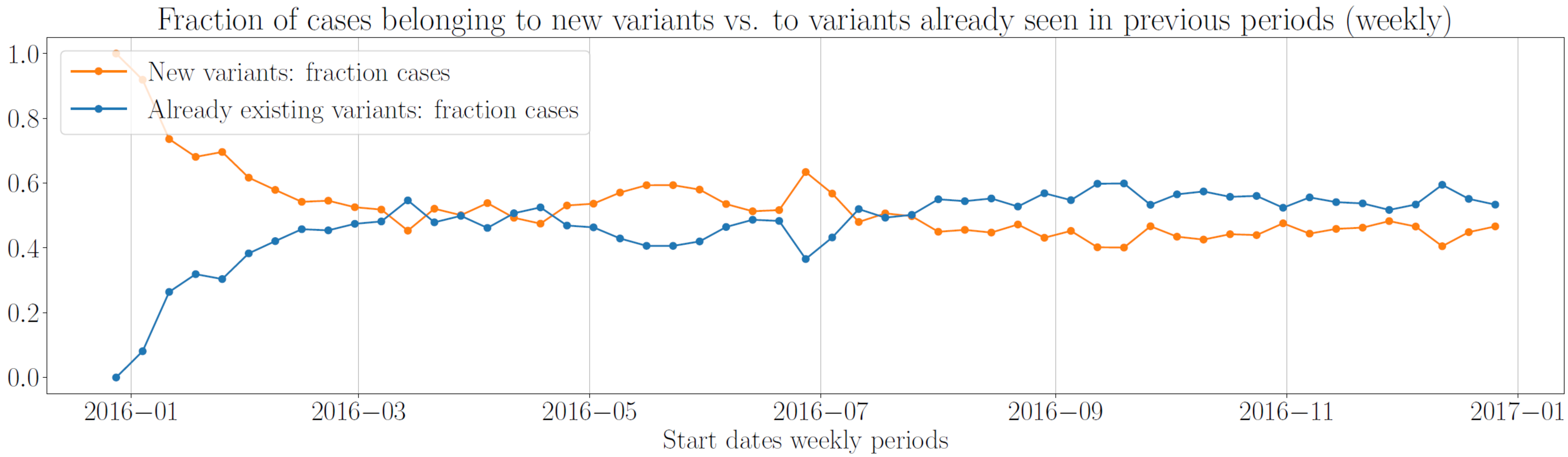
Simulated Log Variants ⊆ Event Log
→Precision

Test Log Variants ⊆ Simulated Log
→ Generalization

+ Take occurences into account

Peeperkorn, J., vanden Broucke, S., De Weerdt, J. (2022). Can recurrent neural networks learn process model structure? Journal Of Intelligent Information Systems. doi: 10.1007/s10844-022-00765-x

Motivation



[1] doi: 10.4121/uuid:5f3067df-f10b-45da-b98b-86ae4c7a310b

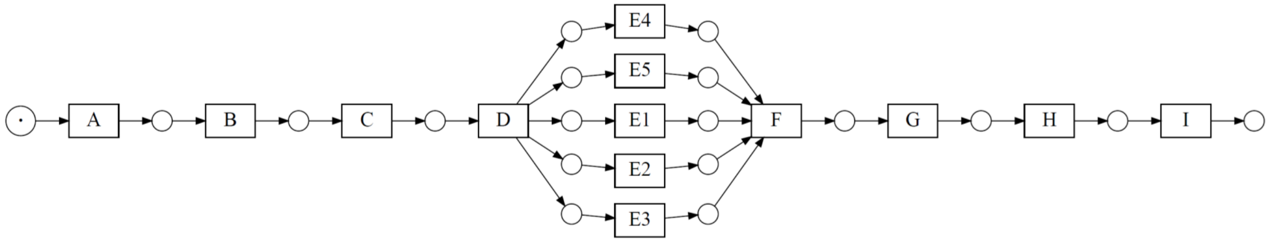
Plotted with **DyLoPro**

DyLoPro

Results

- We ran this experiment for LSTM recurrent neural networks
- If we tune the hyperparameters based purely on accuracy (of validation set)
  - Like done in standard setups
  - Even for simple models + only one variant in the test log

    → Low to **<u>no generalization</u>**, especially for parallel behavior

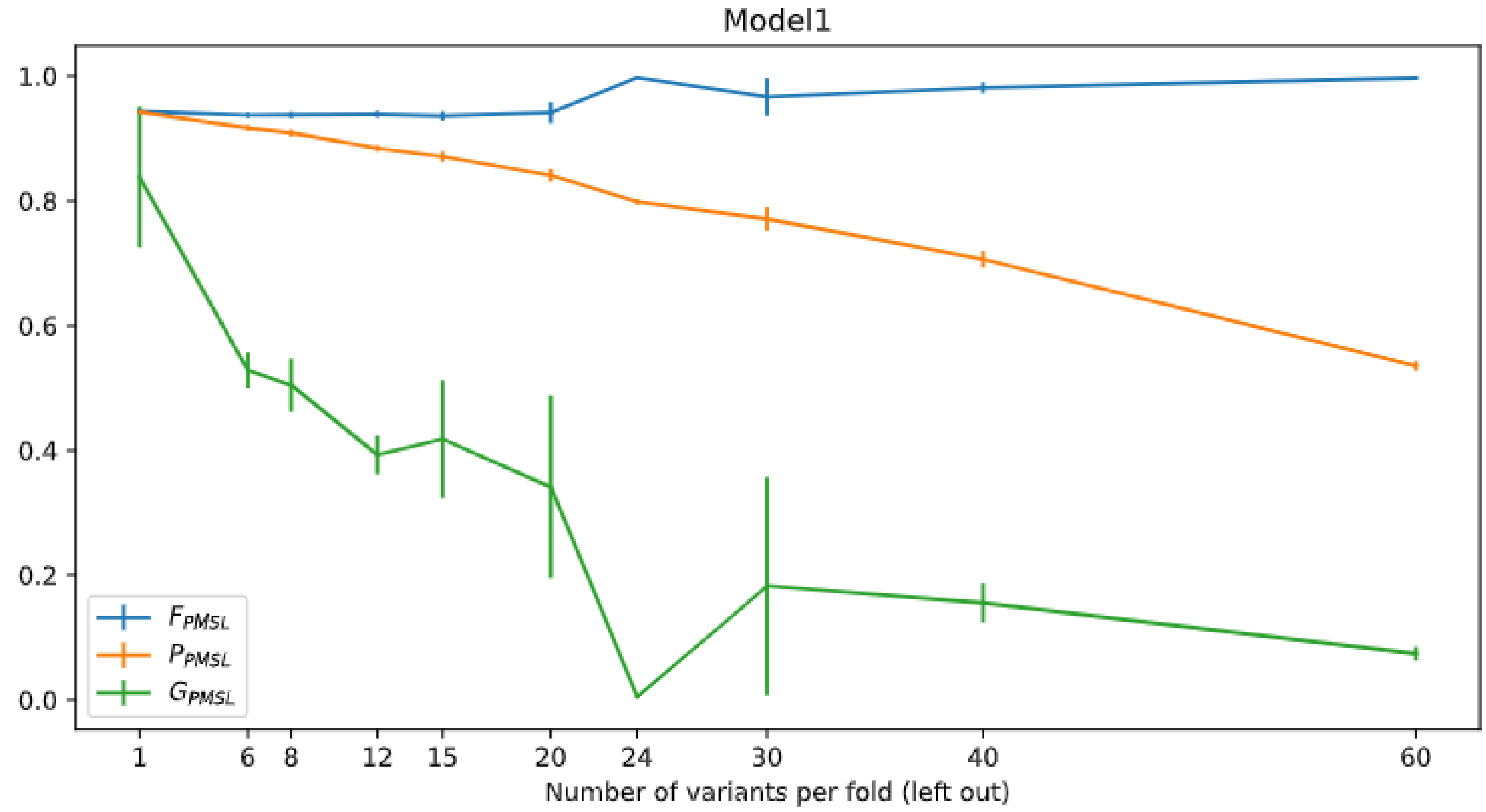


- If we adjust hyperparameter to enforce overfitting countermeasures
  - Dropout, Regularization
  - We do get higher generalization with only a little cost in precision

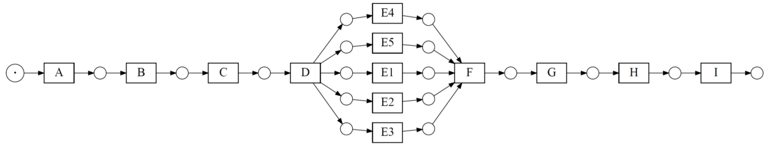
Peeperkorn, J., vanden Broucke, S., De Weerdt, J. (2022). Can recurrent neural networks learn process model structure? Journal Of Intelligent Information Systems. doi: 10.1007/s10844-022-00765-x

Results

- But… Adjusting these hyperparameters works up until a certain #variants missing
- Plots fitness, precision, and generalization, for different #variants in test set
- For LSTMs **<u>with optimized anti-overfitting</u>** hyperparameters! (not accuracy-based)

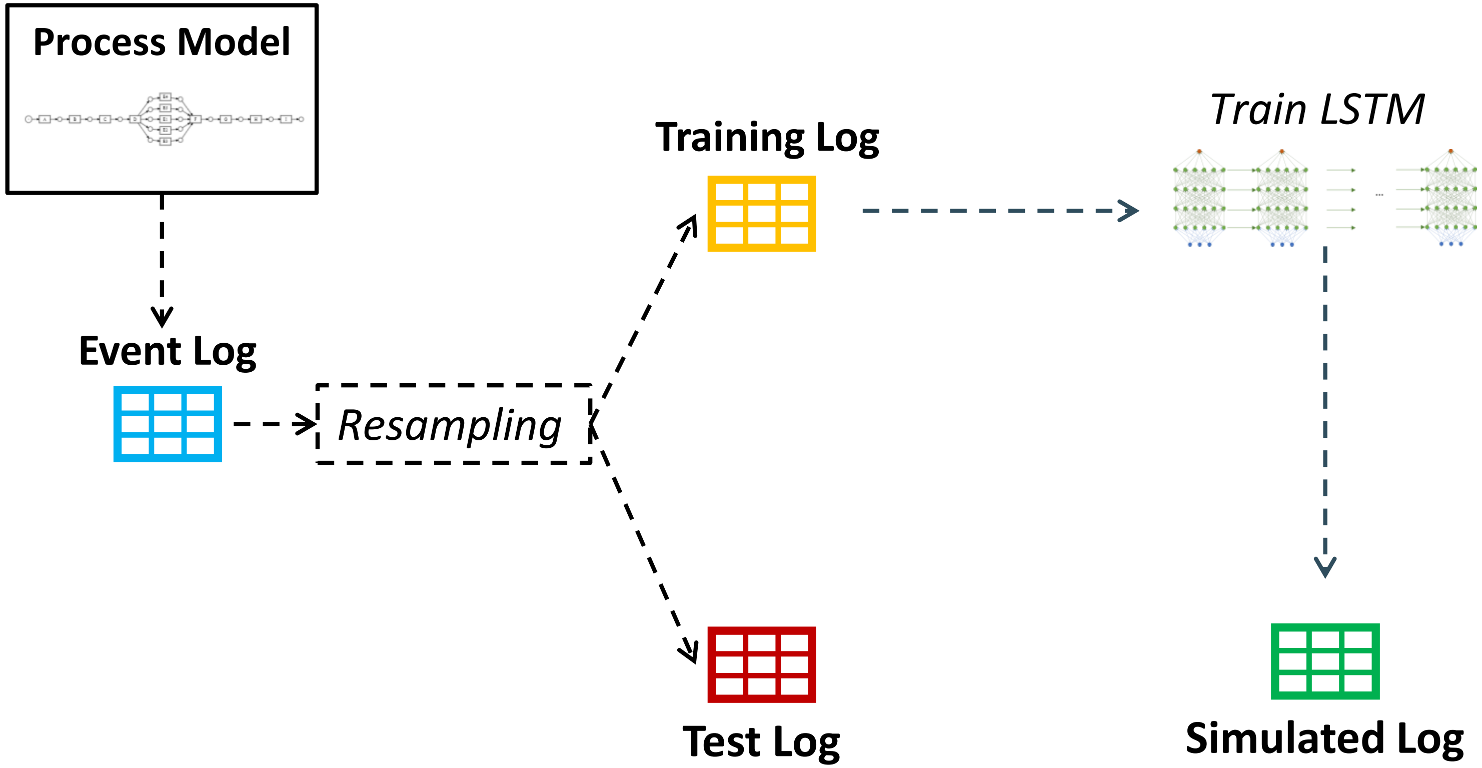


+ These hyperparameters were obtained in a post-hoc way → not suitable for end-users

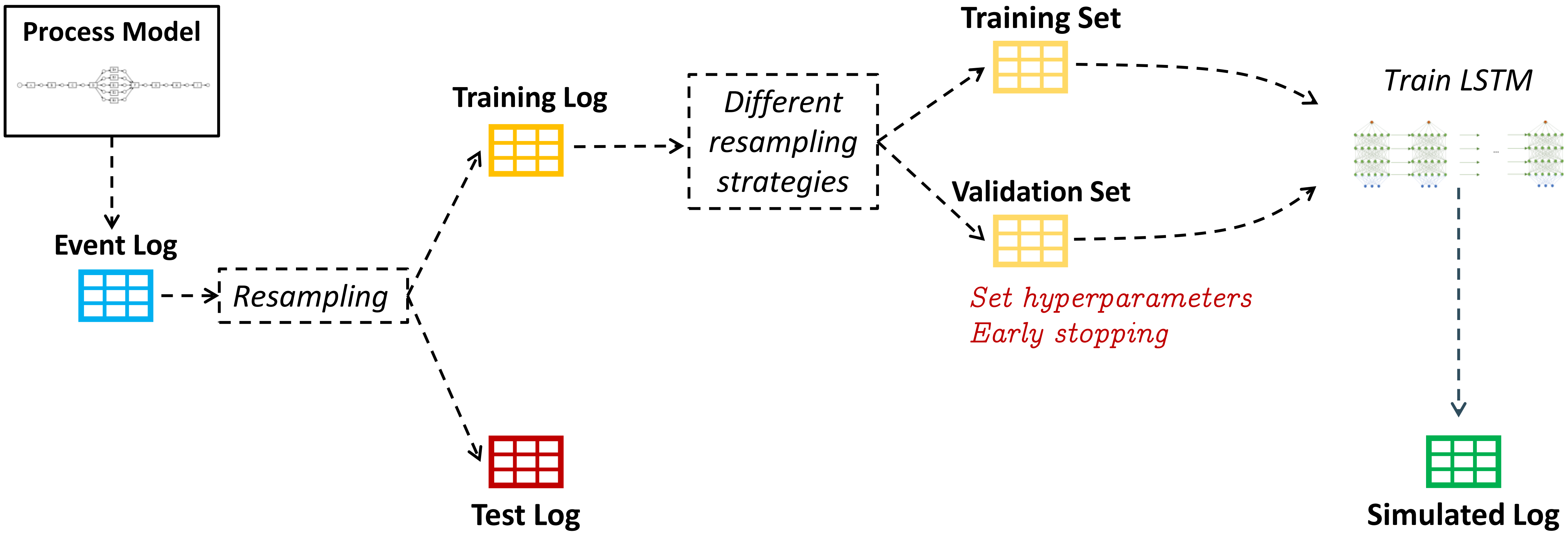


n Systems. doi: 10.1007/s10844-022-00765-x

Validation set sampling



Validation set sampling



Key take-aways

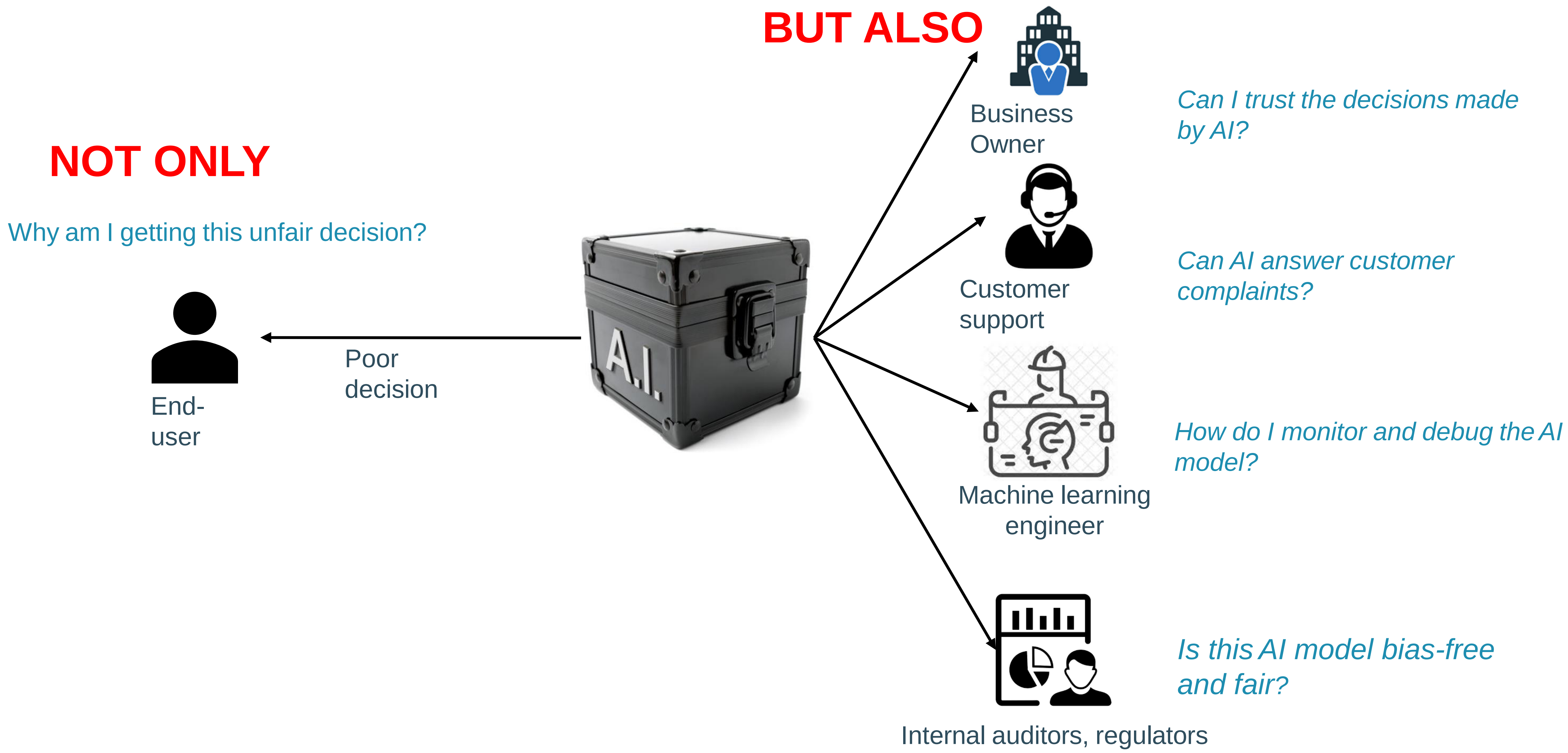
- On simple process models, we show that we can increase generalization at a little cost in fitness/precision by tuning the validation setup
- If models become too complex, the variant-based resampling becomes less effective
- Also important effect of event log incompleteness, but not yet fully understood

- We need to further investigate validation set sampling techniques
  - Factoring in the data perspective (case + event features)

- Alternative model architectures might work better
  - Moving away from classical RNNs
    - Transformers/attention
    - Graph Neural Networks (GNNs)
    - Transfer learning/finetuning

Interpretability, robustness & uncertainty

Why do we need Explainable AI (XAI)?



What is the goal of Explainable AI?

Depending on the maturity of your AI system

- The goal of XAI differs with the maturity of AI
- AI versus human performance

  1) AI is weaker

  ➔ XAI is about improving the AI system

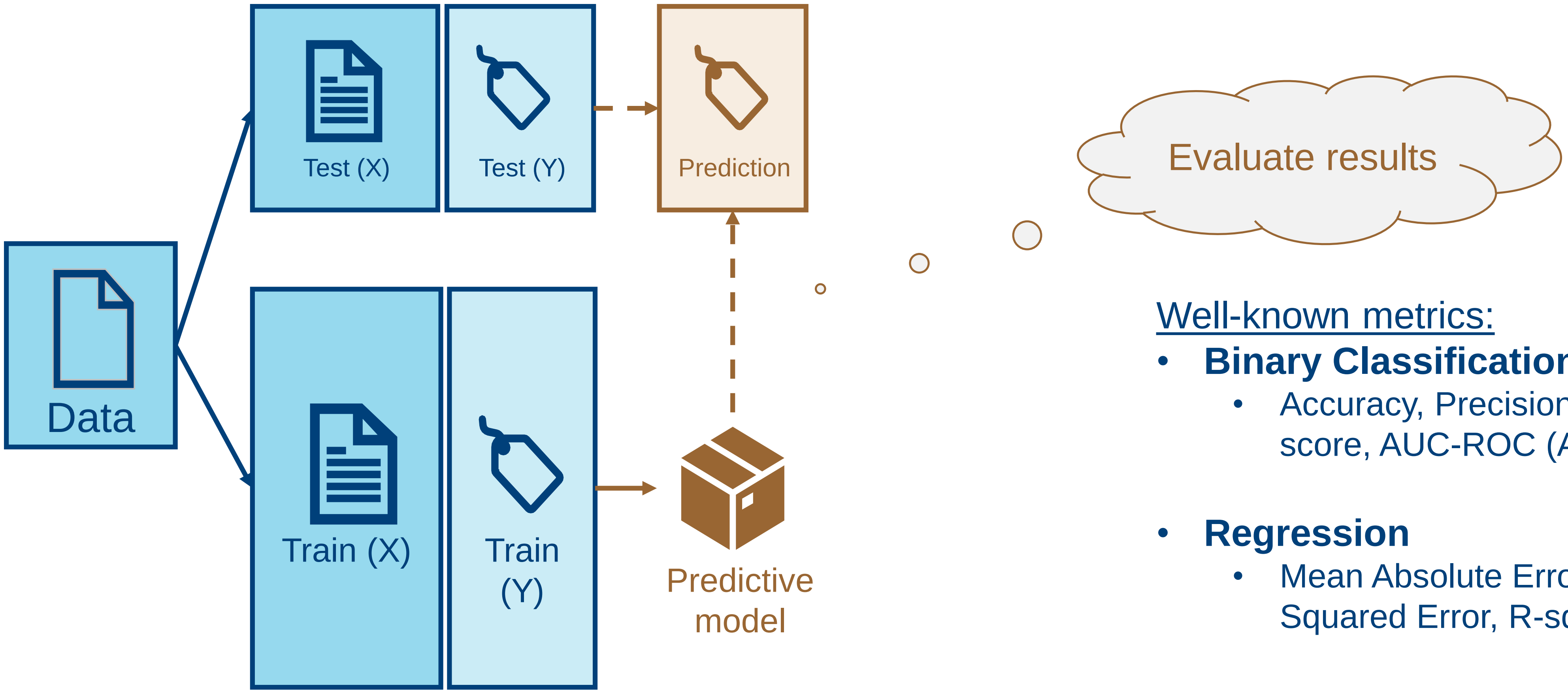
  2) AI is on par

  ➔ XAI is about building trust in the system

  3) AI is stronger

  ➔ XAI is about explaining a complicated concept & informing humans

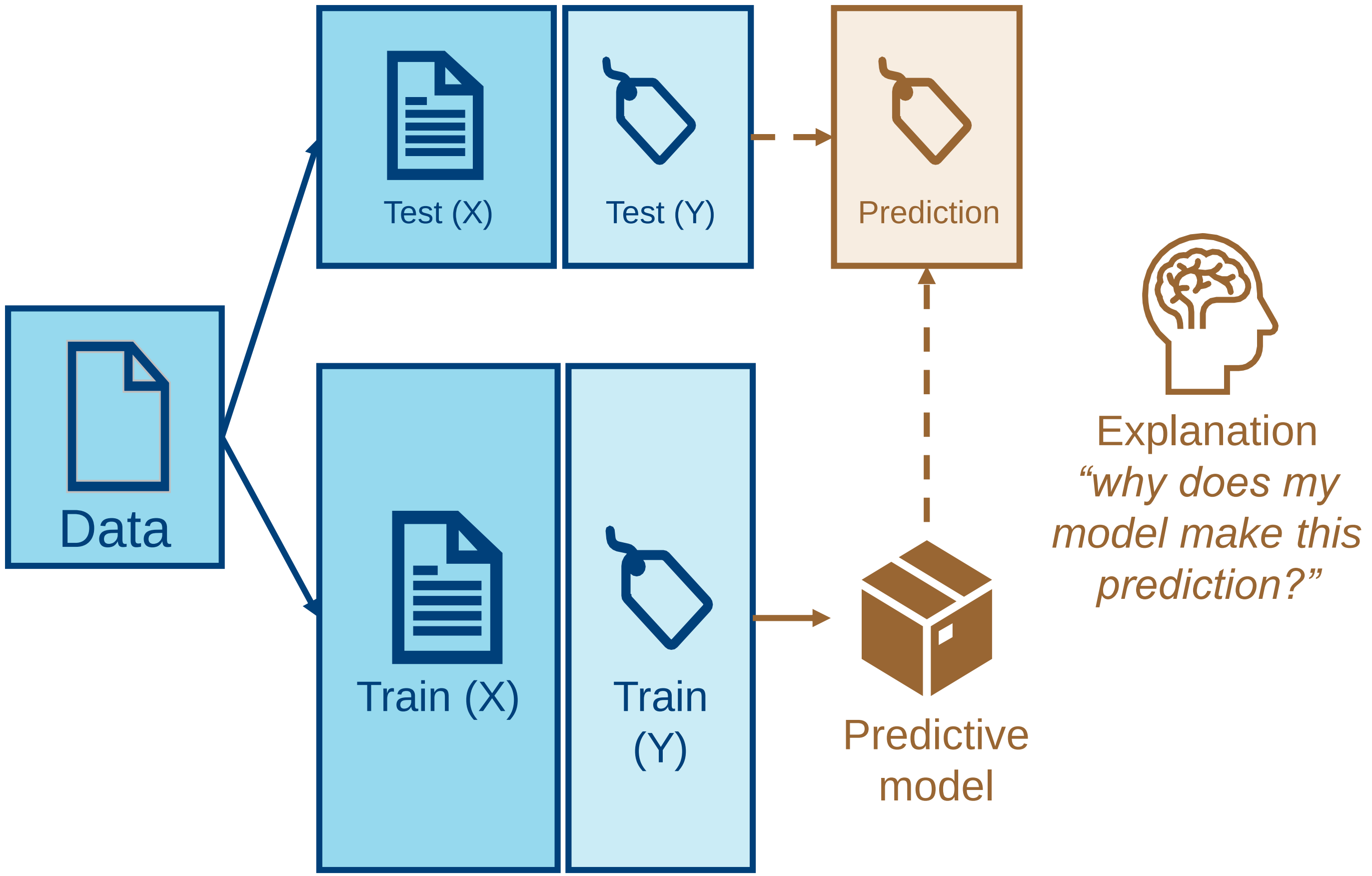
Explainability



Well-known metrics:

- **Binary Classification:**
  - Accuracy, Precision, Recall, F1-score, AUC-ROC (AUC-PRC)
- **Regression**
  - Mean Absolute Error, (Root) Mean Squared Error, R-squared

Explainability



- **Transparent models**
  - Logistic/linear regression
  - Decision trees
- **Post-hoc methods**
  - Permutation importance
  - Shapley plots
  - Local models (LIME, …)
  - Counterfactuals
  - Prototypes
  - …

Explainability

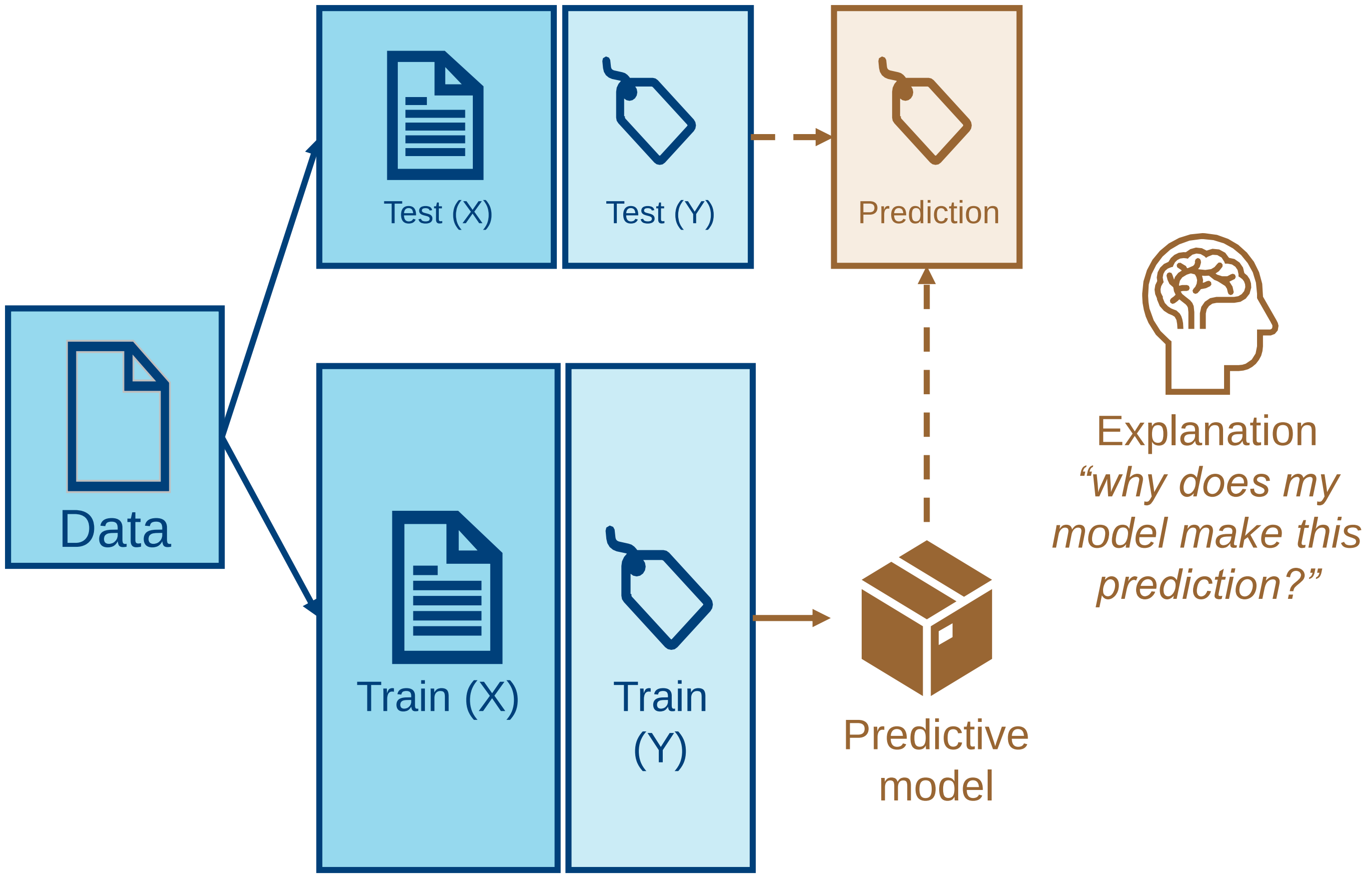
- However… explainability is more than just “getting a shapley plot”
- “Interpretable” is defined by people, not algorithms
- Explanations of correlated parameters?
  - Simpson’s paradox can also mess up explanations!
- Explaining the data or explaining the model?
  - *Chen, Hugh, Joseph D. Janizek, Scott Lundberg, and Su-In Lee. “True to the Model or True to the Data?” ArXiv Preprint ArXiv:2006.16234, June 29, 2020. http://arxiv.org/abs/2006.16234*
- Explanation itself should be interpretable + reliable (faithful)!
- Explainability is about communication
- Ideally it should help us to get actionable insights!

Explainability

- XAI tailored to PPM seems a must to have any chance at improved adoption

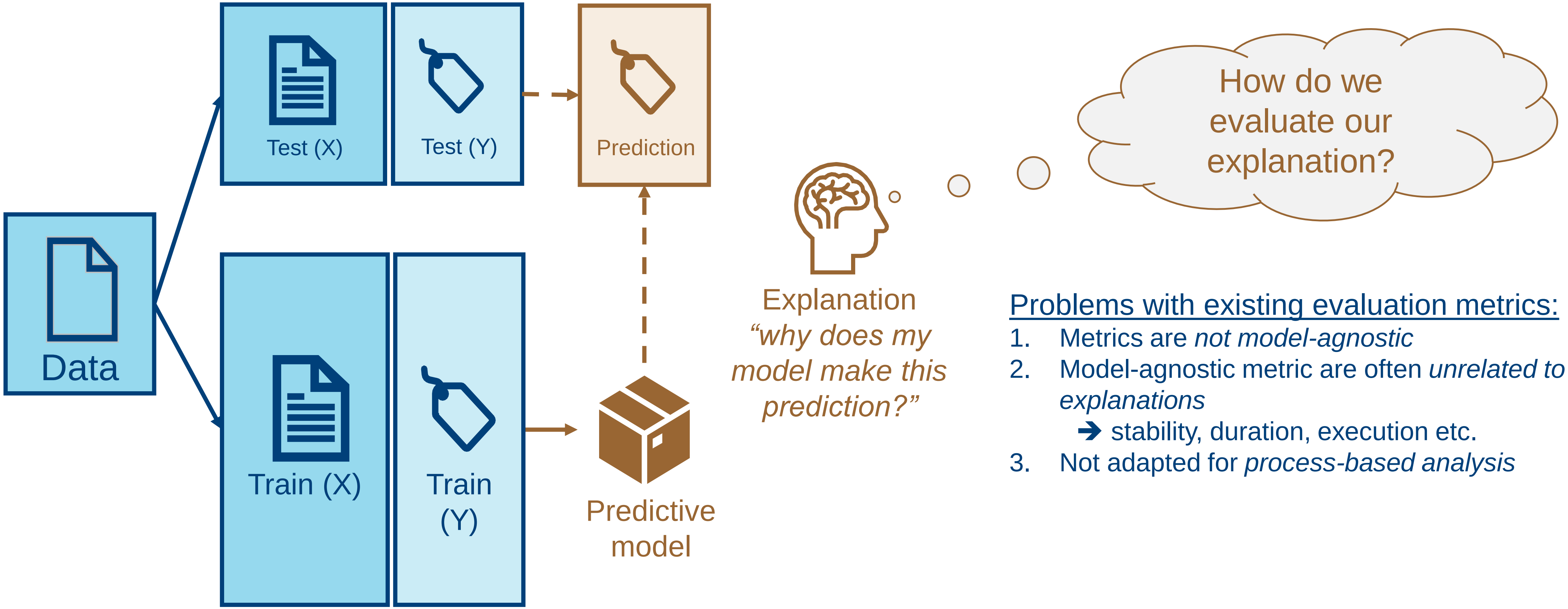
- Growing number of works in PPM addressing the problem directly:
  - Rizzi, W., Di Francescomarino, C., & Maggi, F. M. (2020). Explainability in predictive process monitoring: when understanding helps improving. In International Conference on Business Process Management (pp. 141-158). Cham: Springer International Publishing.
  - Huang, T. H., Metzger, A., & Pohl, K. (2021). Counterfactual explanations for predictive business process monitoring. In European, Mediterranean, and Middle Eastern Conference on Information Systems (pp. 399-413). Cham: Springer International Publishing.
  - Stevens, A., & De Smedt, J. (2023). Explainability in process outcome prediction: Guidelines to obtain interpretable and faithful models. European Journal of Operational Research
  - Wickramanayake, B., Ouyang, C., Xu, Y., & Moreira, C. (2023). Generating multi-level explanations for process outcome predictions. Engineering Applications of Artificial Intelligence, 125, 106678.
  - …

Explainability



- **Transparent models**
  - Logistic/linear regression
  - Decision trees
- **Post-hoc methods**
  - Permutation importance
  - Shapley plots
  - Local models (LIME, …)
  - Counterfactuals
  - Prototypes
  - …

Explainability: how to evaluate the explanations?

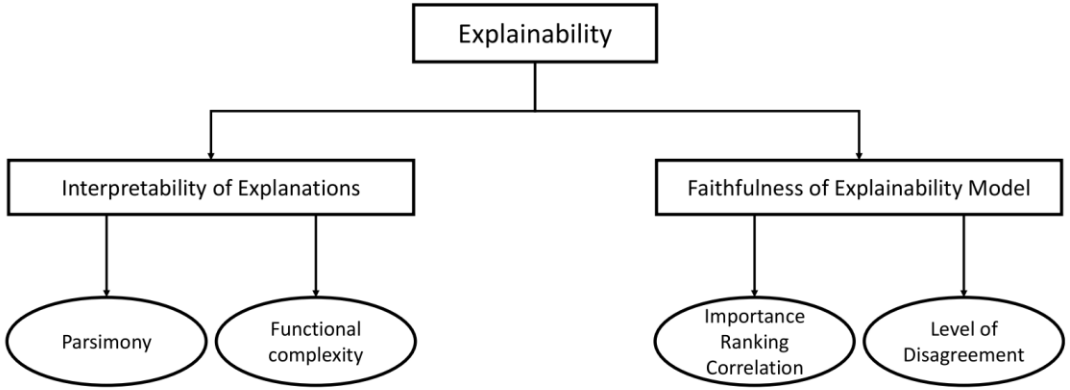


<u>Problems with existing evaluation metrics:</u>

1. Metrics are *not model-agnostic*
2. Model-agnostic metric are often *unrelated to the explanations*
   ➔ stability, duration, execution etc.
3. Not adapted for *process-based analysis*

[1]*Stevens, A., De Smedt, J., & Peeperkorn, J. (2022, March). Quantifying explainability in outcome-oriented predictive process monitoring. In Process Mining Workshops: ICPM 2021 International Workshops, Eindhoven, The Netherlands, October 31–November 4, 2021, Revised Selected Papers (pp. 194-206). Cham: Springer International Publishing.*
[2]*Stevens, A., & De Smedt, J. (2022). Explainability in Process Outcome Prediction: Guidelines to Obtain Interpretable and Faithful Models (EJOR)*

Explainability: how to evaluate the explanations?



- Different aspects **evaluating the explanations**

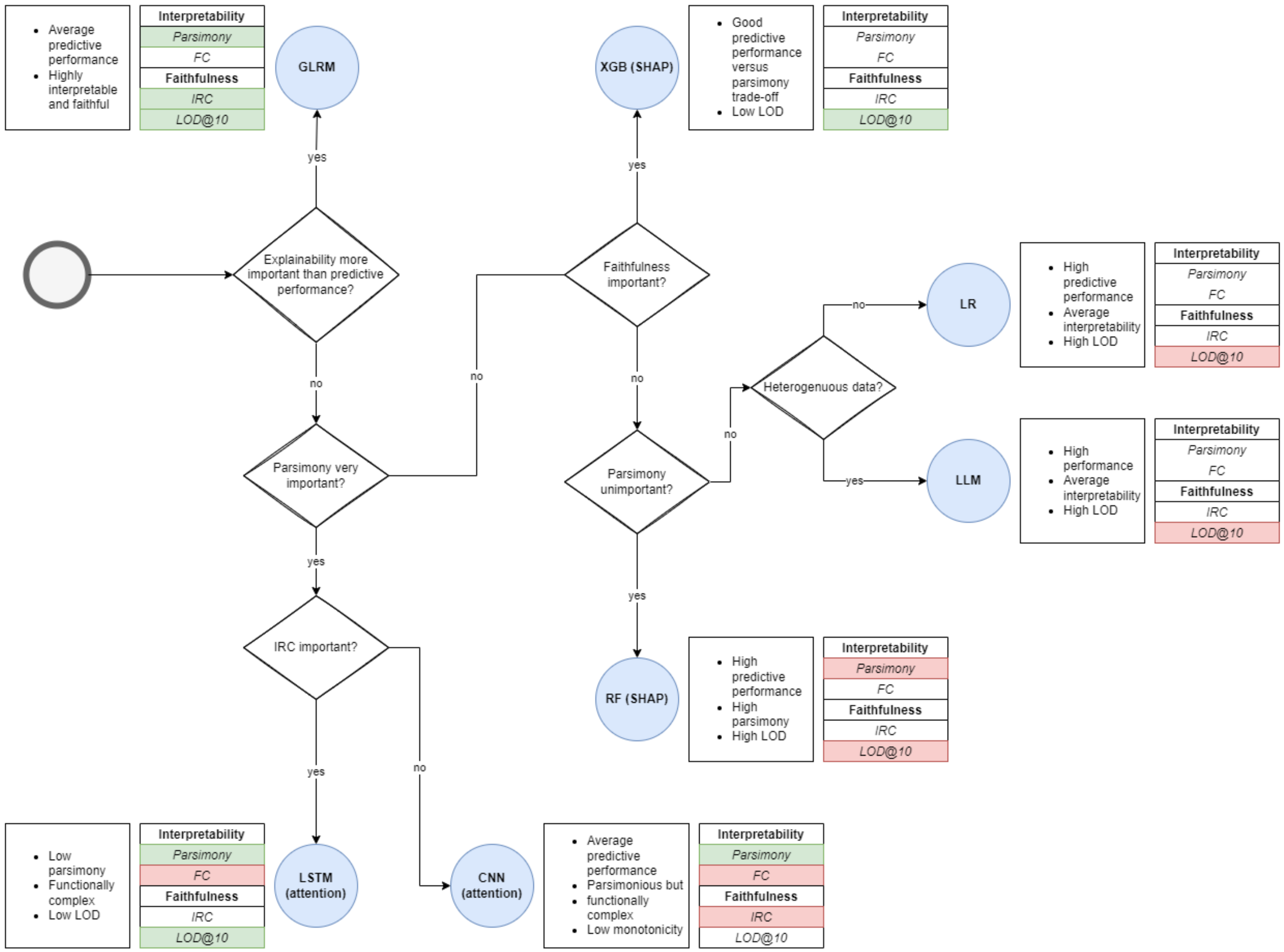
- **Parsimony** represents the complexity of a model
  - *Number of non-zero weights*
- **Functional complexity**
  - *(Advanced) permutation importance to get the complexity of the explanation*
- **Importance ranking correlation**
  - *How faithful the attribute importance ranking of the explainability model is to the ranking made by the permutation attribute importance (PI)*
- **Level of disagreement (LOD@10)**
  - *Investigates whether the PI and the explainability model focus on the same attribute type (rather than whether they predicted the same thing)*

X-MOP: A Framework of guidelines to obtain interpretable and faithful models

✓ 7 state-of-the-art models

✓ 13 real-life event logs

[2]*Stevens, A., & De Smedt, J. (2022). Explainability in Process Outcome Prediction: Guidelines to Obtain Interpretable and Faithful Models (EJOR)*



[2] *Stevens, A., & De Smedt, J. (2022). Explainability in Process Outcome Prediction: Guidelines to Obtain Interpretable and Faithful Models (EJOR)*

Robustness

- Needed to build *trust*

- How robust are the predictions against noise?
  - Or event log incompleteness? (generalization)

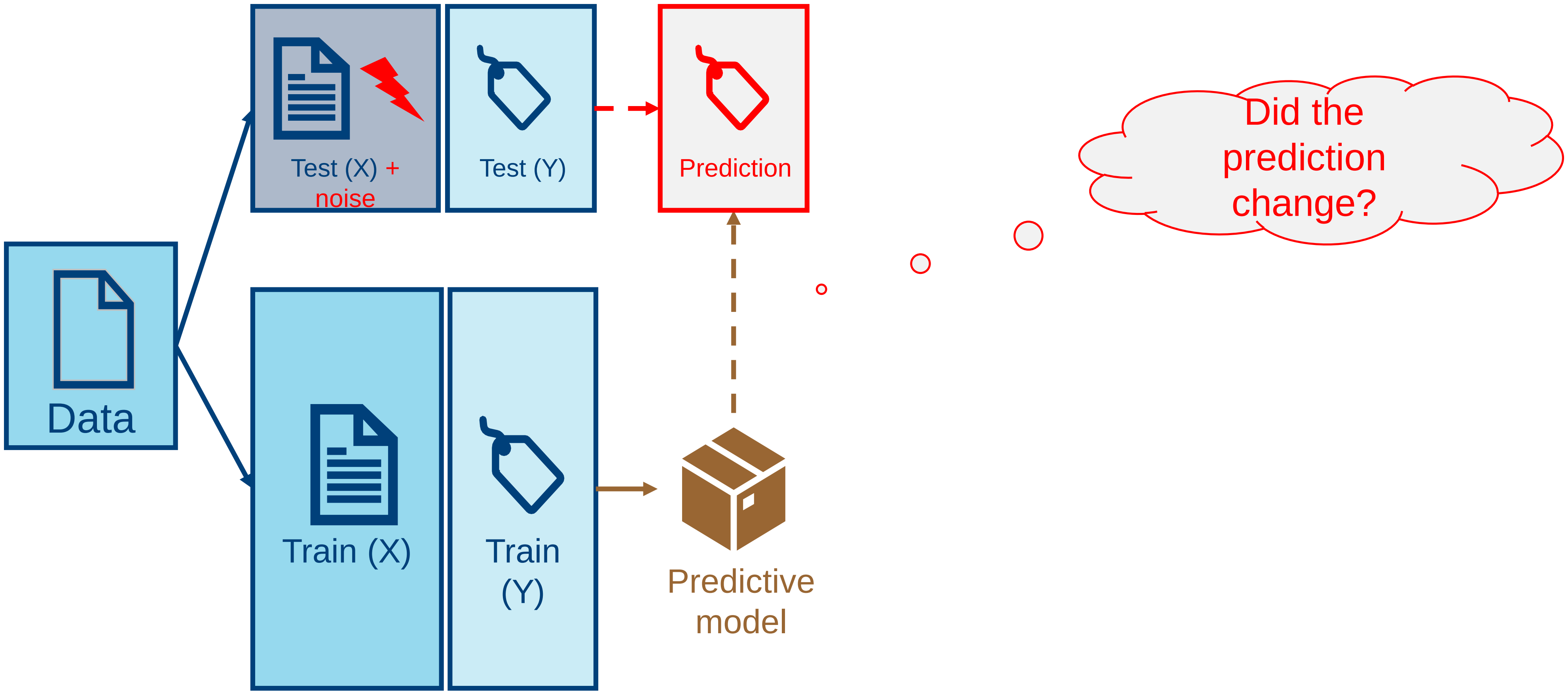
- One option developed at our department

  → adversarial attacks for process event data

Alexander Stevens, Jari Peeperkorn, Johannes De Smedt , Jochen De Weerdt. Assessing the Robustness in Predictive Process Monitoring through Adversarial Attacks. ICPM (2022)

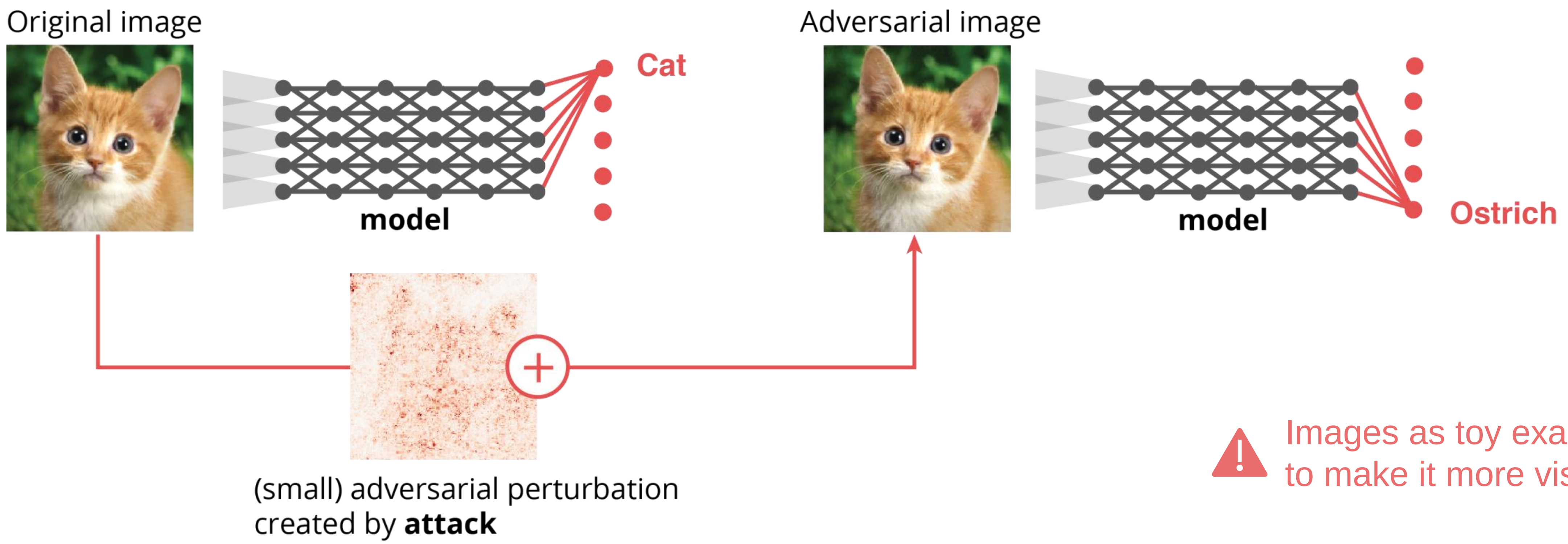
Alexander Stevens, Jari Peeperkorn, Johannes De Smedt , Jochen De Weerdt. Manifold Learning for Adversarial Robustness in Predictive Process Monitoring. ICPM (2023)

Adversarial attacks



KU LEUVEN

Adversarial attacks

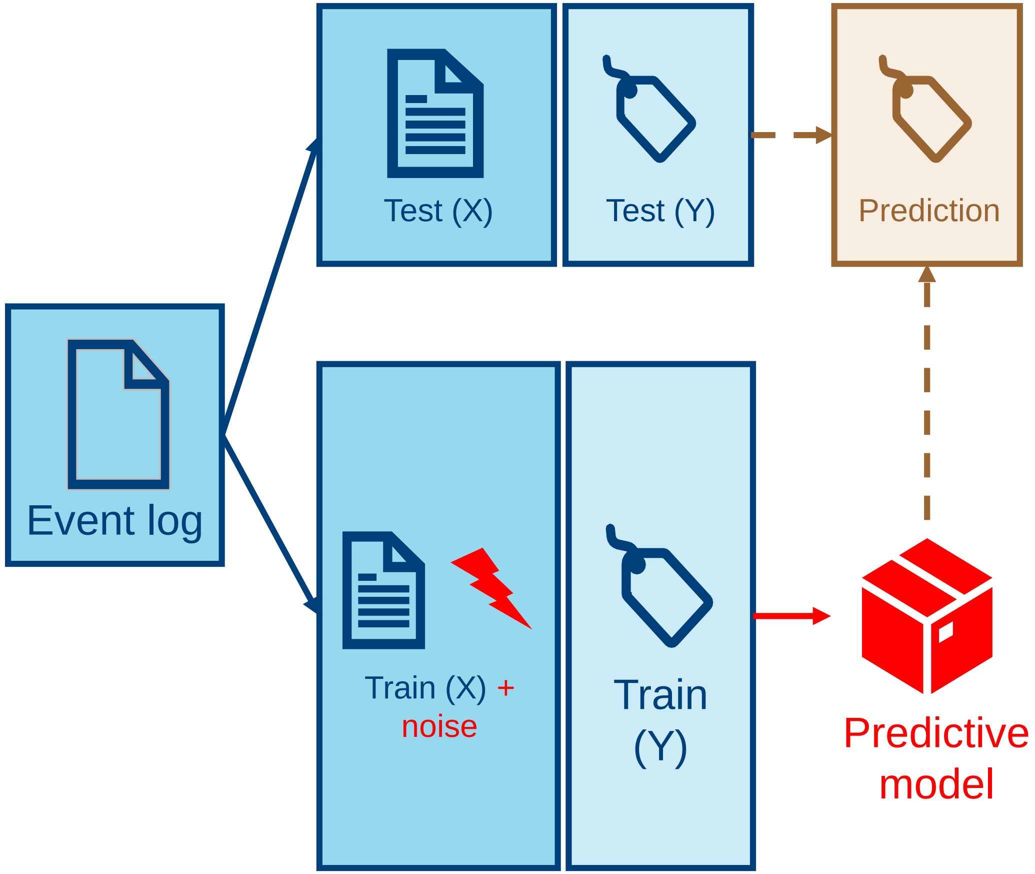


Images as toy example to make it more visual

Small *perturbation* causes the model to make a false prediction"[1,2]

[1]*Molnar, C. (2022). Interpretable Machine Learning: A Guide for Making Black Box Models Explainable (2nd ed.). christophm.github.io/interpretable-ml-book/*
[2]*Figure: NIPS 2018 Adversarial Vision Challenge*

Adversarial training



- Did prediction change?
- **Did the explanation change?**

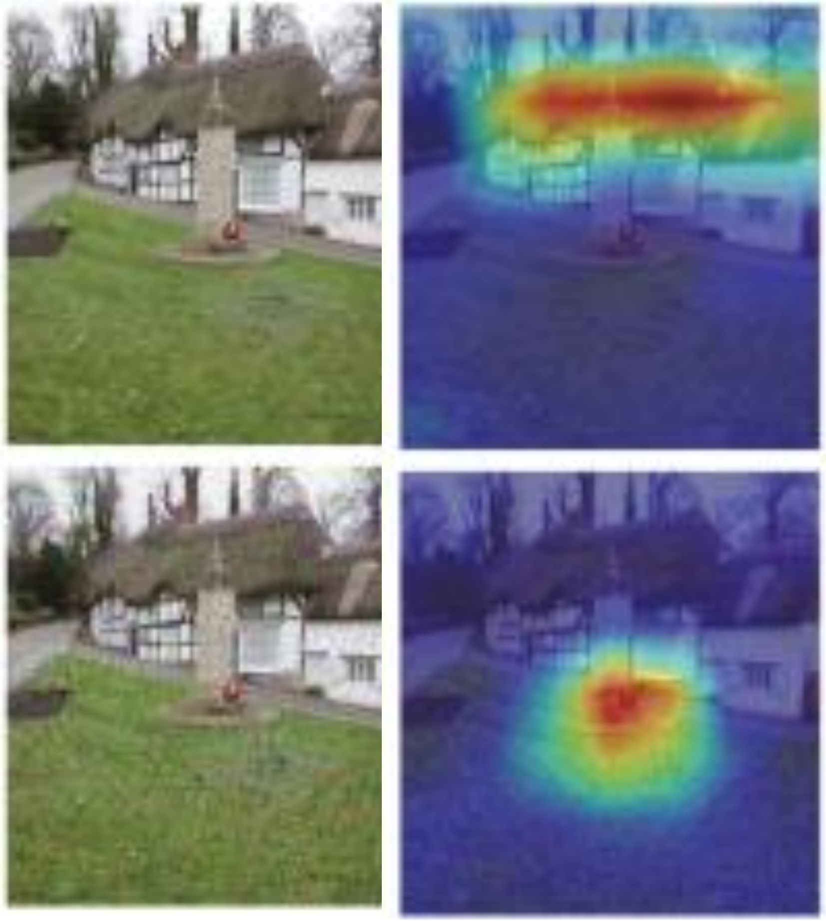
Adversarial training

- **The prediction is unchanged, but the XAI method is focusing on something completely different!**

“this picture contains a house”

Explanation

Image



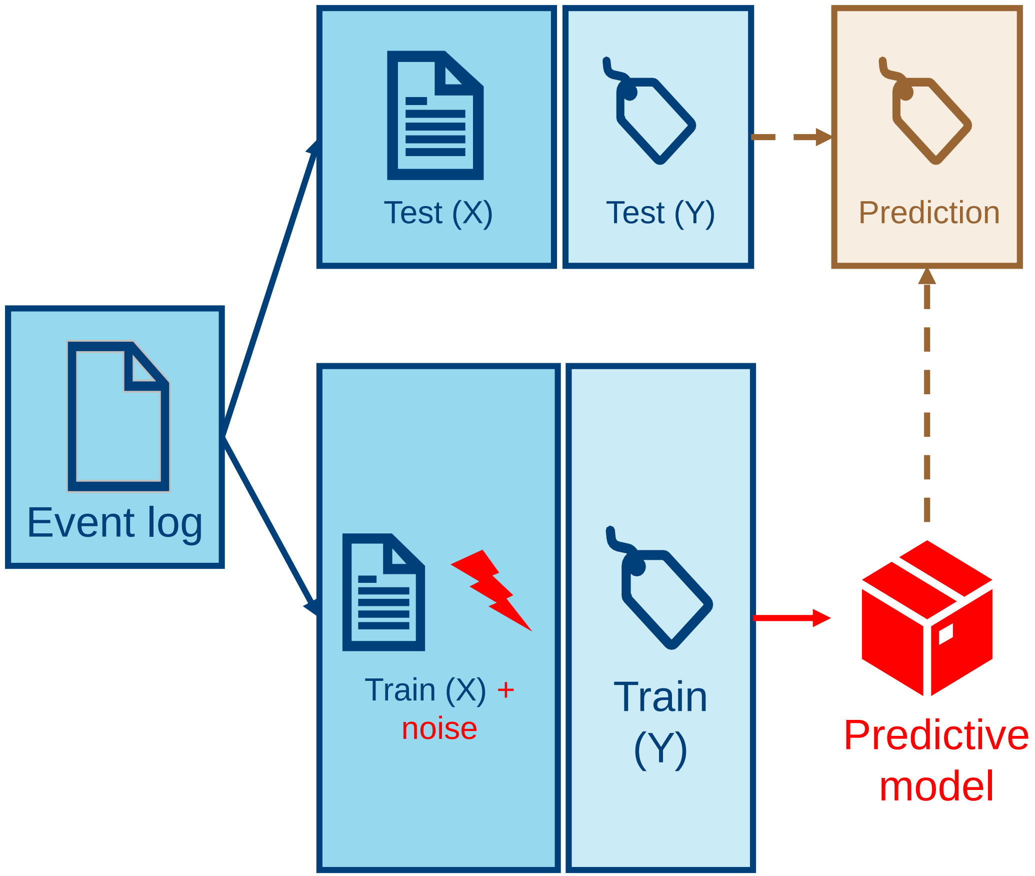
Image

\+ Noise

➔ Small perturbations can cause explanations to change, although the prediction is unchanged

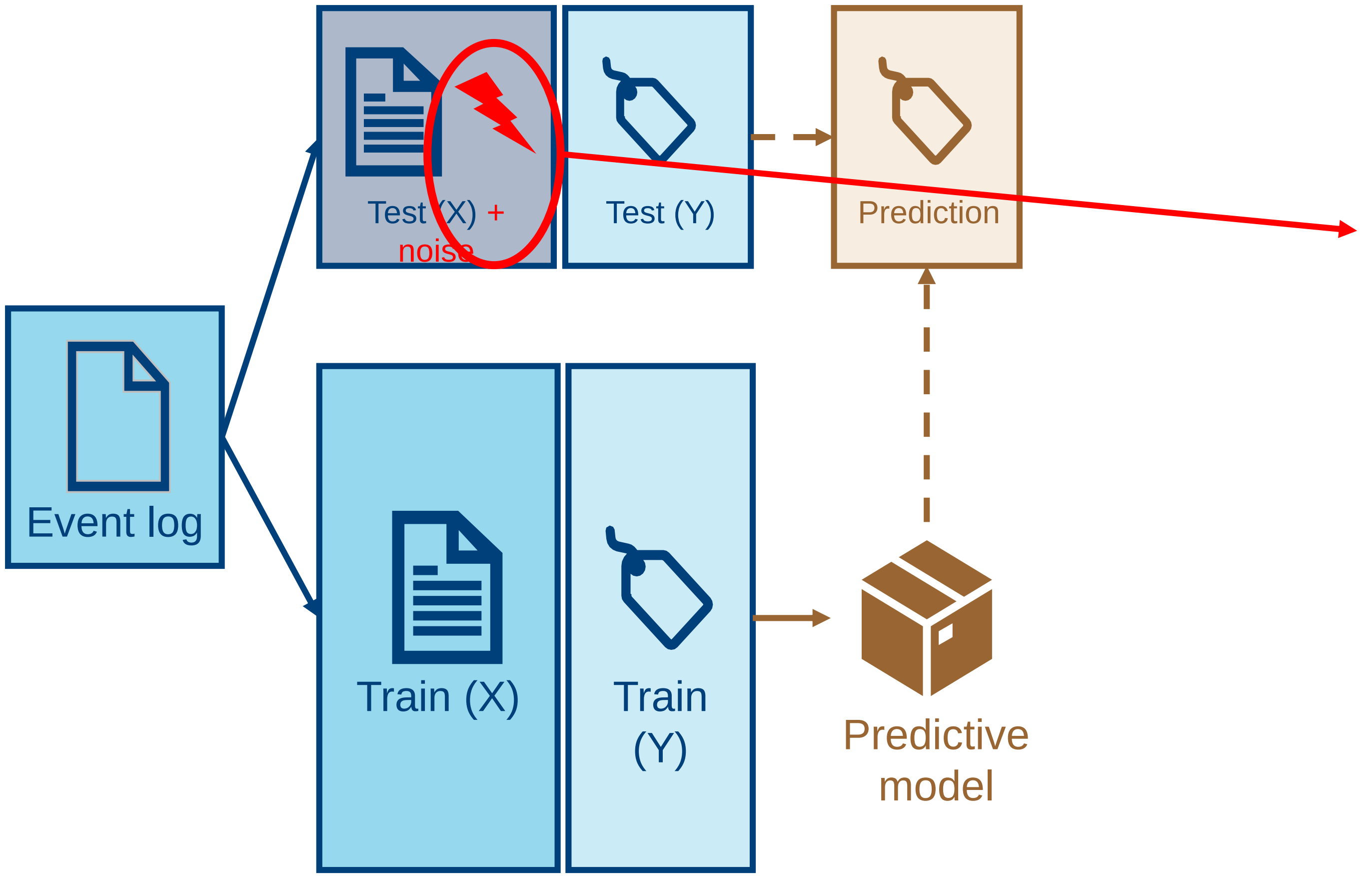
➔ Your setup model + explanations is not robust

Adversarial training



- Did prediction change?
- Did the explanation change?
- **Use to enhance training set**
  → increase robustness!
  “Adversarial training as a proactive defense mechanism”

Adversarial attacks



What is this noise?
What is an adversarial example?

- Replacing last event attributes with noise
- Replacing other events with noise

→ not ideal

KU LEUVEN

Adversarial attacks

- Adding just noise as attacks is not necessarily a smart thing to do
  - Our induced noise can be unnatural

    → values that can never happen or are very unlikely in that situation
  - No guarantee that the underlying label of the instance after the adversarial attack did not change

    → Because a certain value changed → ground truth should have also changed

- On-manifold attacks
  - Train an auto-encoder on training data (one for each class)
  - Take an adversarial attack example (with noise)

    → put through autoencoder (both encoder and decoder)

    → use the decoded example

    "within the data manifold"

Uncertainty

- The (un)certainty with which a prediction is taken by predictive models

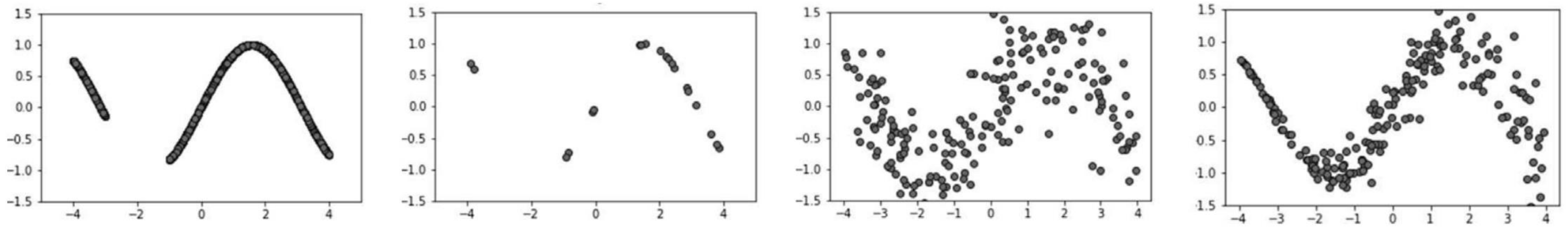
= could be crucial for decision-making (certainly when there is a human-in-the-loop)

→ Using predictive probability *p* in classification rather than only the predicted class

- Input data is also not perfect

→ introduces uncertainty

1. Epistemic uncertainty (*lack of knowledge*): e.g. limited # of measurements
2. Aleatoric uncertainty (*randomness*): variability



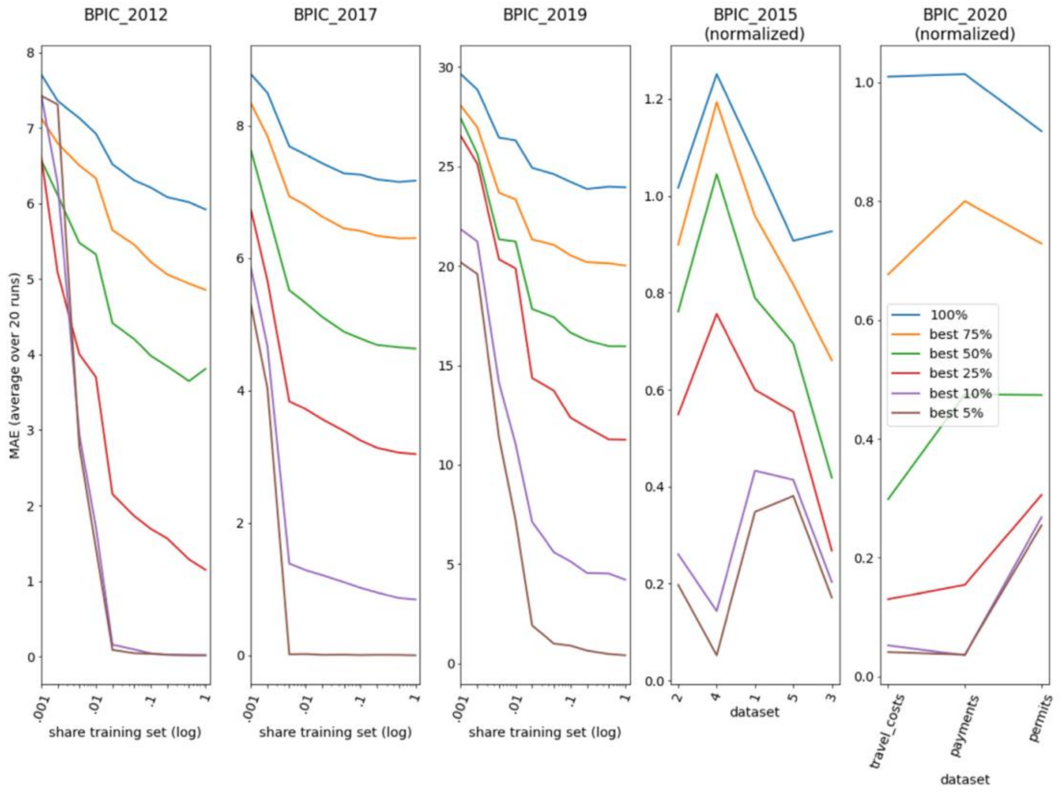
(a) Epistemic uncertainty: example 1. (b) Epistemic uncertainty: example 2. (c) Aleatoric uncertainty: homoscedastic. (d) Aleatoric uncertainty: heteroscedastic.

Uncertainty

- Bayesian Neural Networks
  - *Each weight is a distribution rather than a single number*
  - Allowing them to express uncertainty (epistemic and aleatoric)
  - Assess the probability of each prediction and provide greater insight into the accuracy of the output
  - At our department → developed a PPM technique for this purpose

- Models allow to predict point value together with the uncertainty (confidence interval)
  - Allows for enriched symbiosis of automated and manual decision making
  - One can also apply PPM to smaller datasets

- Future work: conformal prediction?



Weytjens, H., & De Weerdt, J. (2022). Learning uncertainty with artificial neural networks for predictive process monitoring. *Applied Soft Computing*, 109134. doi: 10.1016/j.asoc.2022.109134

Key takeaways

- Explainability should be regarded from a holistic perspective
- Adapting XAI techniques to process data is not always straight forward
- PPM models needs to be robust, however not much metrics exist
  - Adversarial attacks
- Models that inherently take into account uncertainty can be of added value for decision support

KU LEUVEN

The inter-case perspective

Inter-case perspective

- Research in predictive process monitoring has generally relied on **intra-case features** in order to make predictions
  - Features that provide information about the execution history of a specific ongoing case of interest

- Therefore, assume that the processing of a case is solely dependent on the attributes of the case itself

- However, cases are not processed in isolation
  - Can be influenced by the processing of other cases
  - Can be influenced by the general state of a business process
  - These dynamics can be captured by **inter-case features**

KU LEUVEN

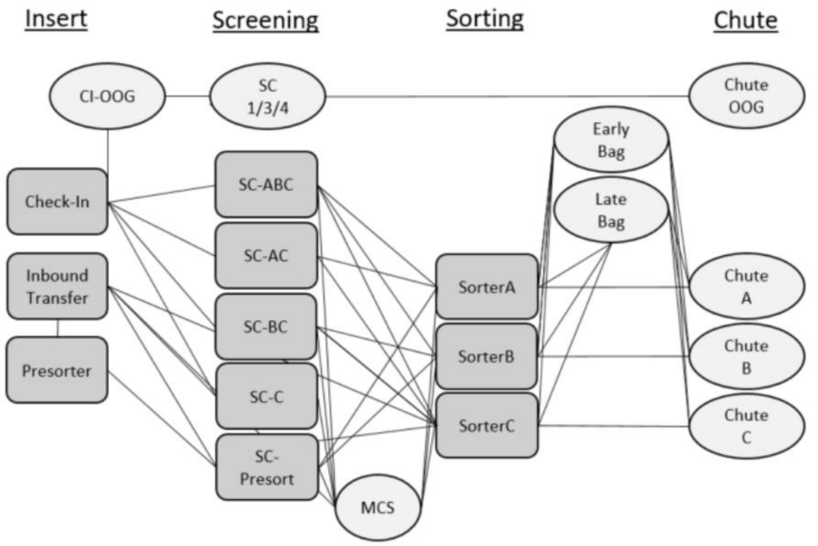
Examples of PPM research including an inter-case perspective

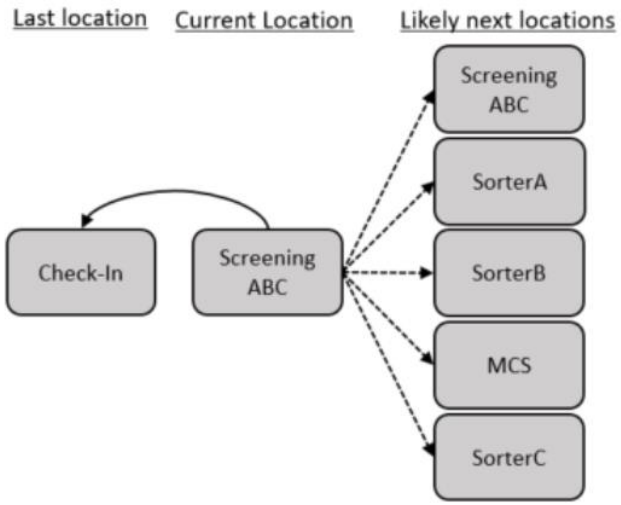
- Senderovich, A., Di Francescomarino, C., & Maggi, F. M. (2019). From knowledge-driven to data-driven inter-case feature encoding in predictive process monitoring. *Information Systems*, *84*, 255-264.
- Klijn, E. L., & Fahland, D. (2020). Identifying and reducing errors in remaining time prediction due to inter-case dynamics. In *2020 2nd International Conference on Process Mining (ICPM)* (pp. 25-32). IEEE.
- Grinvald, A., Soffer, P., & Mokryn, O. (2021). Inter-case properties and process variant considerations in time prediction: A conceptual framework. In *International Conference on Business Process Modeling, Development and Support* (pp. 96-111). Cham: Springer International Publishing.
- Kim, J., Comuzzi, M., Dumas, M., Maggi, F. M., & Teinemaa, I. (2022). Encoding resource experience for predictive process monitoring. *Decision Support Systems*, *153*, 113669.
- Gunnarsson, B.R., De Weerdt, J. and vanden Broucke, S. (2022). A framework for encoding the multi-location load state of a business process. In 2022 Proceedings of the International IJCAI Workshop on Process Management in the AI era.
- ...

→ often very specific, designed features

MLS-ICE: A Load Point Inter-Case Encoding Framework for PPM

- The MLS-ICE framework enriches events with the load state of relevant “load points” in a business process
  - Load points can be physical locations, activities, etc.
- Can be configured in several ways, MLS-ICE framework includes:
  - Two approaches for deriving the load state of a single location in a business process
    - Number of cases currently processed at a load point
    - Number of cases in an optimal time window at each load point
  - Two approaches for identifying relevant locations in a business process
    - System-based load point state (all important locations in the system)
    - Case-based load point state (encodes the state at load points in close proximity)





Gunnarsson, B.R., De Weerdt, J. and vanden Broucke, S. (2022). A framework for encoding the multi-location load state of a business process. In 2022 Proceedings of the International IJCAI Workshop on Process Management in the AI era.

Key takeaways

- The inter-case featurization / inter-case prediction model learning problem is far from solved!

- Inter-case featurization and prediction requires:
  1. Even more robust evaluation setups (remember the previous part!)
     - Debiased test set
  2. Capable model learning architectures required (high-dimensional, dynamic event attributes)
     - Transformers?

- My take: *often ignored because it is challenging but in most processes very important!*

- How to characterize the system?
  - What about context? E.g. **IoT**

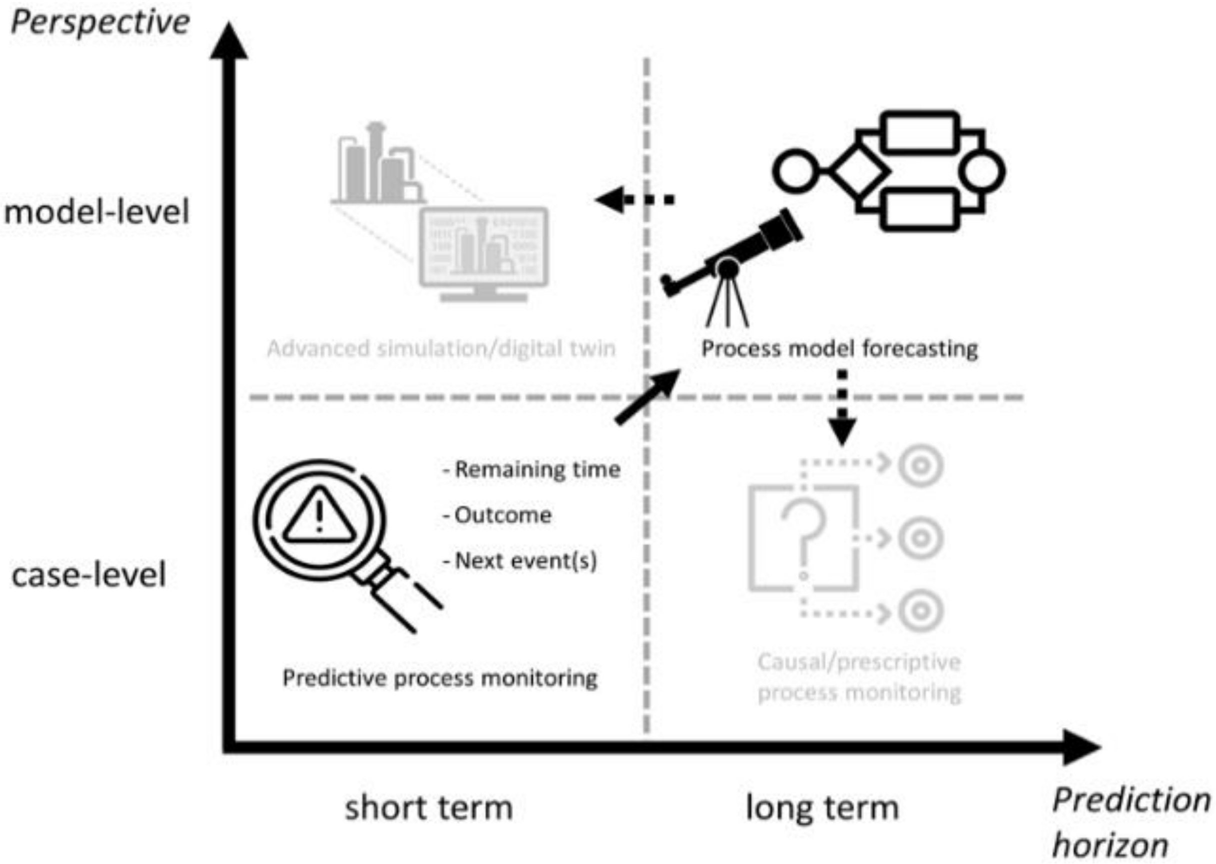
  Work in our research group by Yannis Bertrand
  - What about the **object-centric perspective**? → natural fit!

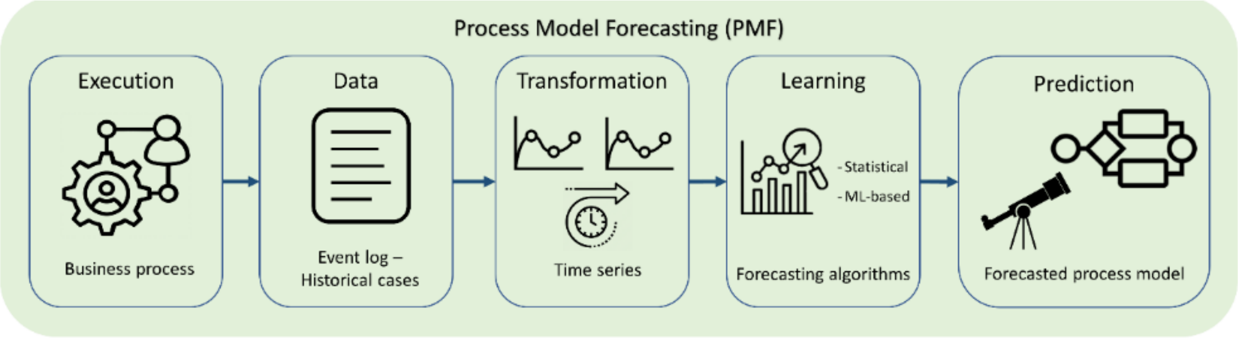
  Adams, Jan Niklas; Drescher, Hannes; Swoboda, Andreas; Günnemann, Nikou; Park, Gyunam; and van der Aalst, Wil, "Improving Predictive Process Monitoring Using Object-Centric Process Mining" (2024). ECIS 2024 Proceedings. 7
  - What about interprocess dependencies?

From case-level to process level predictions

Process Model Forecasting (PMF)

- Shift from operational to tactical decision support
- *Micro to macro*





De Smedt, J., Yeshchenko, A., Polyvyanyy, A., De Weerdt, J., & Mendling, J. (2021, October). Process model forecasting using time series analysis of event sequence data. In *International Conference on Conceptual Modeling* (pp. 47-61). Cham: Springer International Publishing.

De Smedt, J., Yeshchenko, A., Polyvyanyy, A., De Weerdt, J., Mendling (2023). Process model forecasting and change exploration using time series analysis of event sequence data. *Data & Knowledge Engineering*, *145*, Art.No. ARTN 102145. doi: 10.1016/j.datak.2023.102145

Turning event logs into DF (directly-follows) time series

Case ID	Activity	Timestamp
1	a_1	11:30
1	a_2	11:45
1	a_1	12:10
1	a_2	12:15
2	a_1	11:40
2	a_1	11:55
3	a_1	12:20
3	a_2	12:40
3	a_2	12:45

3 intervals

Equitemporal: 12:45-11:30 = 75 minutes
3 intervals of 25 minutes:
11:30-11:55, 11:55-12:20,12:20-12:45

Equisized: 9 events: 3 intervals of 3 events

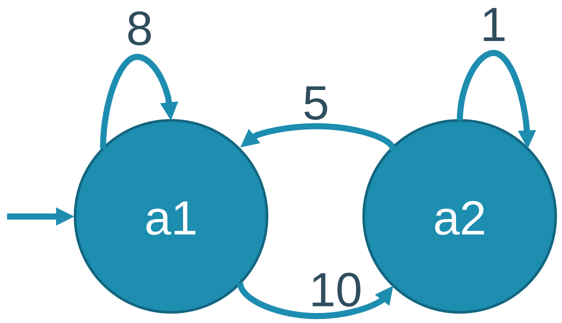
Directly-follows	Equitemporal	Equisized
$<_{L_s} (a_1, a_1)$	[0,1,0]	[1,0,0]
$<_{L_s} (a_1, a_2)$	[1,1,1]	[1,1,1]
$<_{L_s} (a_2, a_1)$	[0,1,0]	[0,1,0]
$<_{L_s} (a_2, a_2)$	[0,0,1]	[0,0,1]

→ DF timeseries

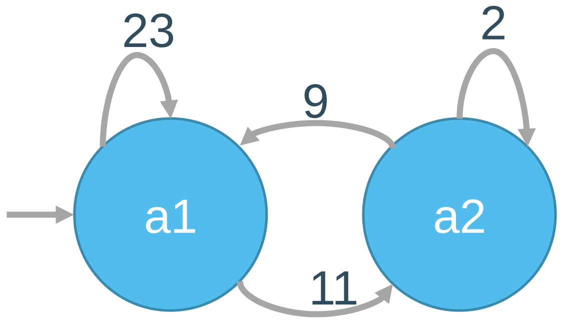
De Smedt, J., Yeshchenko, A., Polyvyanyy, A., De Weerdt, J., Mendling (2023). Process model forecasting and change exploration using time series analysis of event sequence data. *Data & Knowledge Engineering*, *145*, Art.No. ARTN 102145. doi: 10.1016/j.datak.2023.102145

"Predict" the future DFG

Directly-follows	Equitemporal (encoding)
$<_{L_s} (a_1, a_1)$	[0,1,3,2,2,4,5,5,1,8]
$<_{L_s} (a_1, a_2)$	[1,1,1,1,1,3,1,1,6,0]
$<_{L_s} (a_2, a_1)$	[0,1,0,3,6,2,4,1,0,2]
$<_{L_s} (a_2, a_2)$	[0,0,1,0,0,1,0,0,0,1]



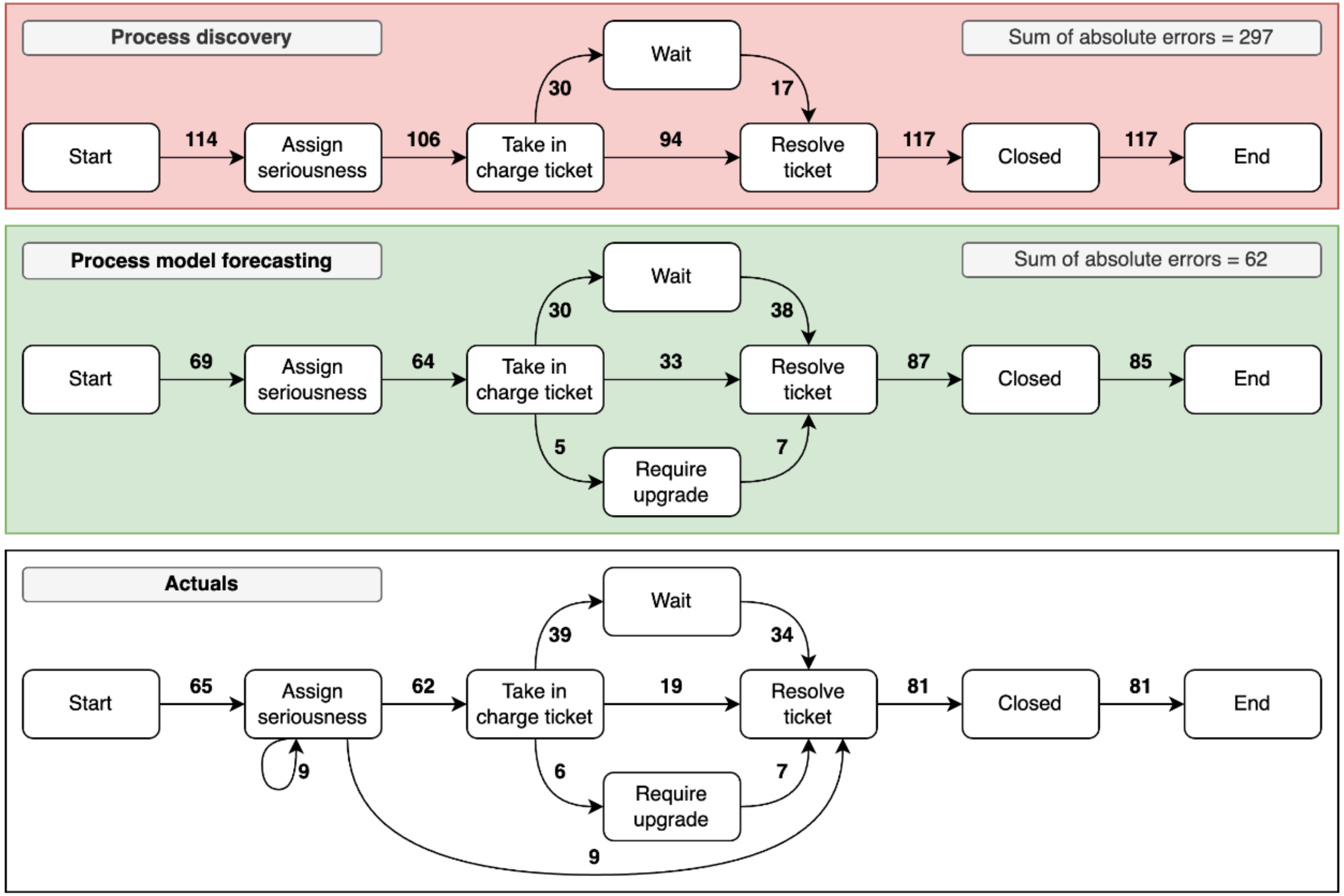
DFG of the past



DFG of the future

De Smedt, J., Yeshchenko, A., Polyvyanyy, A., De Weerdt, J., Mendling (2023). Process model forecasting and change exploration using time series analysis of event sequence data. *Data & Knowledge Engineering*, *145*, Art.No. ARTN 102145. doi: 10.1016/j.datak.2023.102145

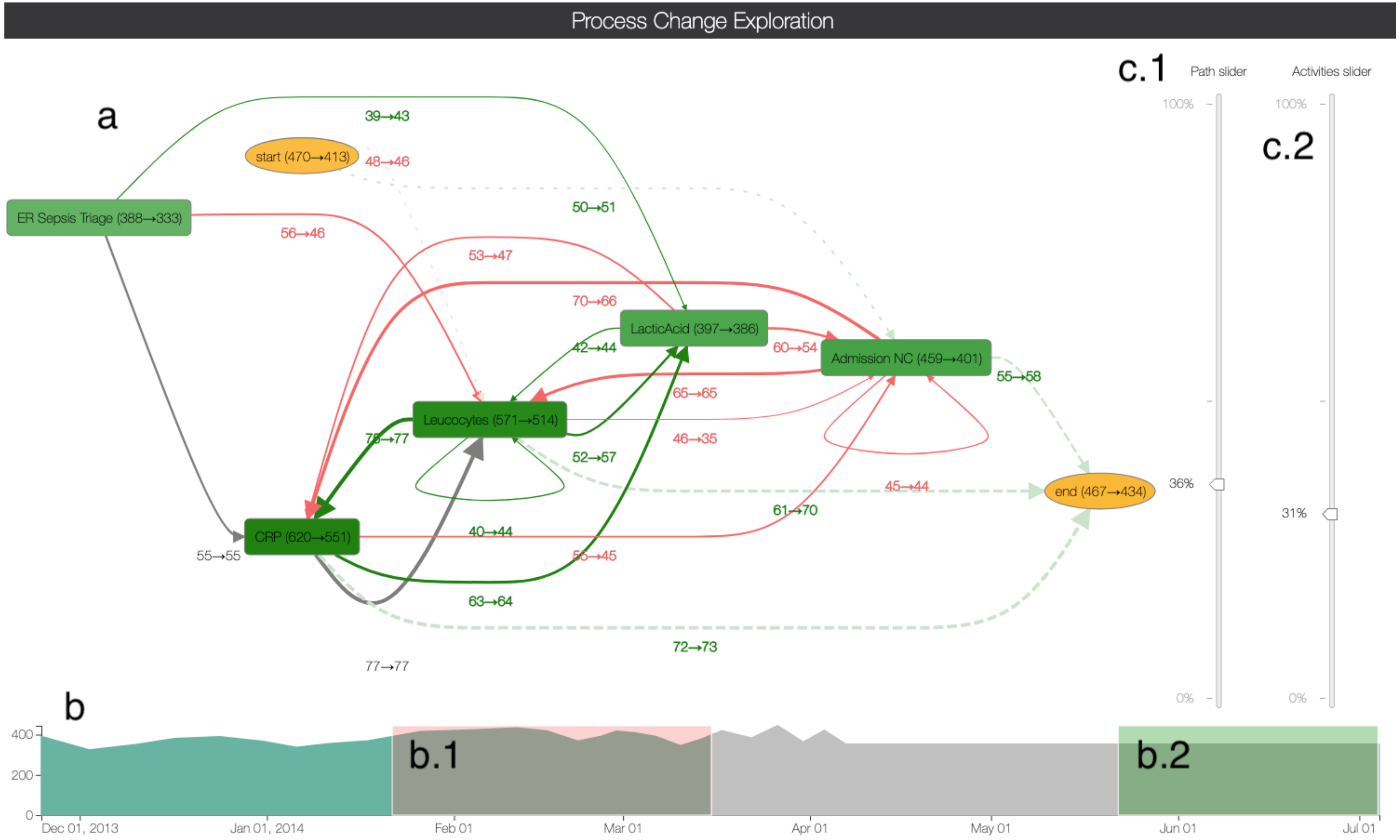
Process Model Forecasting



Italian help desk dataset, https://doi.org/10.4121/uuid:0c60edf1-6f83-4e75-9367-4c63b3e9d5bb

De Smedt, J., Yeshchenko, A., Polyvyanyy, A., De Weerdt, J., Mendling (2023). Process model forecasting and change exploration using time series analysis of event sequence data. *Data & Knowledge Engineering*, *145*, Art.No. ARTN 102145. doi: 10.1016/j.datak.2023.102145

Process Change Exploration tool



De Smedt, J., Yeshchenko, A., Polyvyanyy, A., De Weerdt, J., Mendling (2023). Process model forecasting and change exploration using time series analysis of event sequence data. *Data & Knowledge Engineering*, *145*, Art.No. ARTN 102145. doi: 10.1016/j.datak.2023.102145

Key takeaways

- Process Model Forecasting (PMF) can predict the to-be process model (in the future)

- Current technique relies on simple univariate time series modeling
  - Does not use that much process information
  - *Graph neural network-based approach?*

- How about object-centric processes?
  - Natural fit!
  - Predict “object states”?
  - Or forecast future states of an event knowledge graph (log)?
  - …

Predictive process model monitoring

- In the middle of both predictive process monitoring and process model forecasting

- Processes-As-Movies (PAM)
  - Mining and predicting declarative process constraints between activities in various windows of a process' execution

  (Using encoder-decoder LSTMs and CNNLSTMs)

  - PAM predicts what declarative rules hold for a trace (objective-based)
  - Also supports the prediction of all constraints together as a process model (model-based)

- *Johannes De Smedt, Jochen De Weerdt, Predictive process model monitoring using long short-term memory networks, Engineering Applications of Artificial Intelligence, Volume 133, Part D, 2024*

Some extra remarks

Prescriptive process mining

- How far can we get with PPM?
- Shouldn't we build models that can tell us "what to do" instead of "what will happen"?

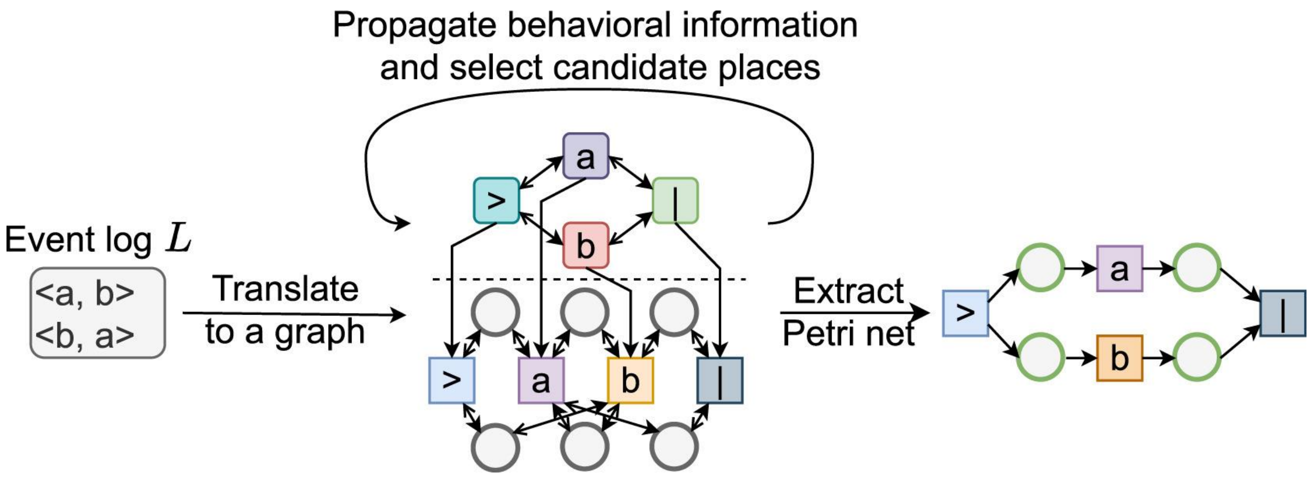
→ Prescriptive process monitoring

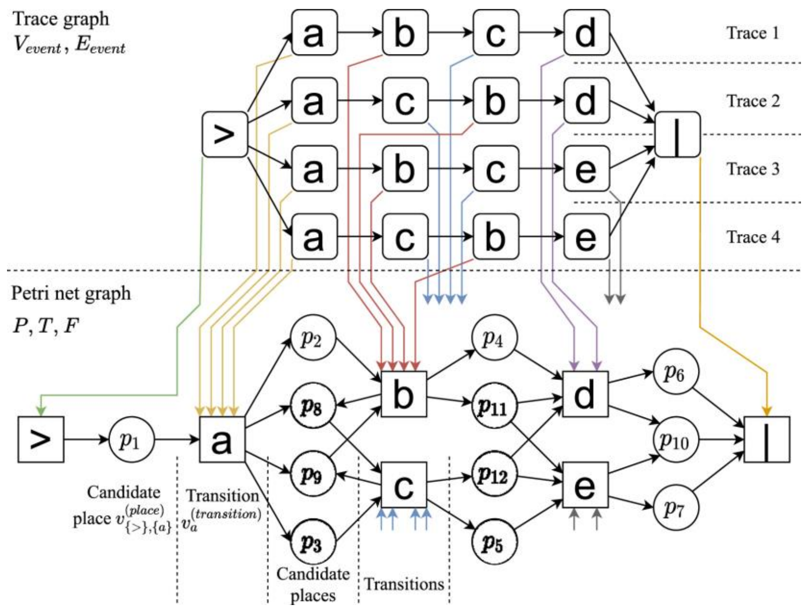
- Difficult to demonstrate effectiveness in offline setting
  - How to define "correct" counterfactuals?
    - Can XAI help?
  - Difficult to manage complexity (isolate decision, intervention timing, intervention types, resource constraints, etc.)
- Online setting: a variety of challenges

- My two cents: you probably need good and reliable prediction to obtain good prescriptions (digital twin)

ML in other PM tasks?

- Process discovery
  - Process discovery as a supervised learning problem using graph neural networks
    - Trained on a large corpus of artificial process models
    - *“Foundation model”*
    - *Dominique Sommers, Vlado Menkovski, Dirk Fahland, Supervised learning of process discovery techniques using graph neural networks, Information Systems, Volume 115, 2023*





KU LEUVEN

ML in other PM tasks?

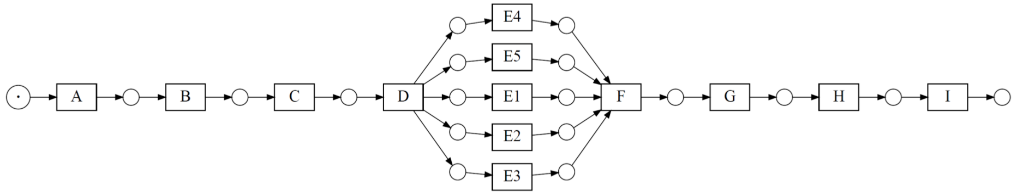
- Conformance checking
  - Work by me

    Jari Peeperkorn, Seppe vanden Broucke, Jochen De Weerdt, Global conformance checking measures using shallow representation and deep learning, Engineering Applications of Artificial Intelligence, Volume 123, Part B, 2023

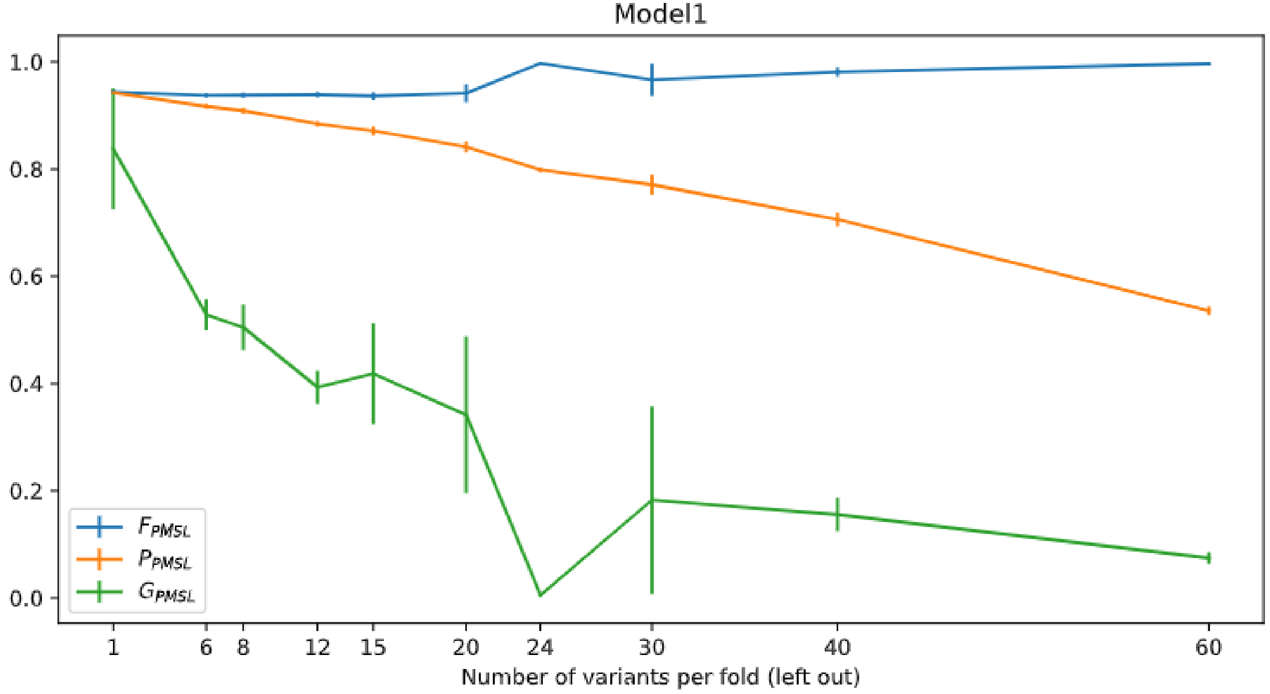
  - Comparing traces in the log to traces in a played-out model log, and in this way obtain fitness and precision scores
    - Using embeddings trained with a shallow neural network (word2vec or doc2vec style)
    - Using a supervised approach with a recurrent neural network

  - Works, and is in principle extendable to perspectives beyond control-flow
  - But dependent on simulation (and completeness of that simulation)

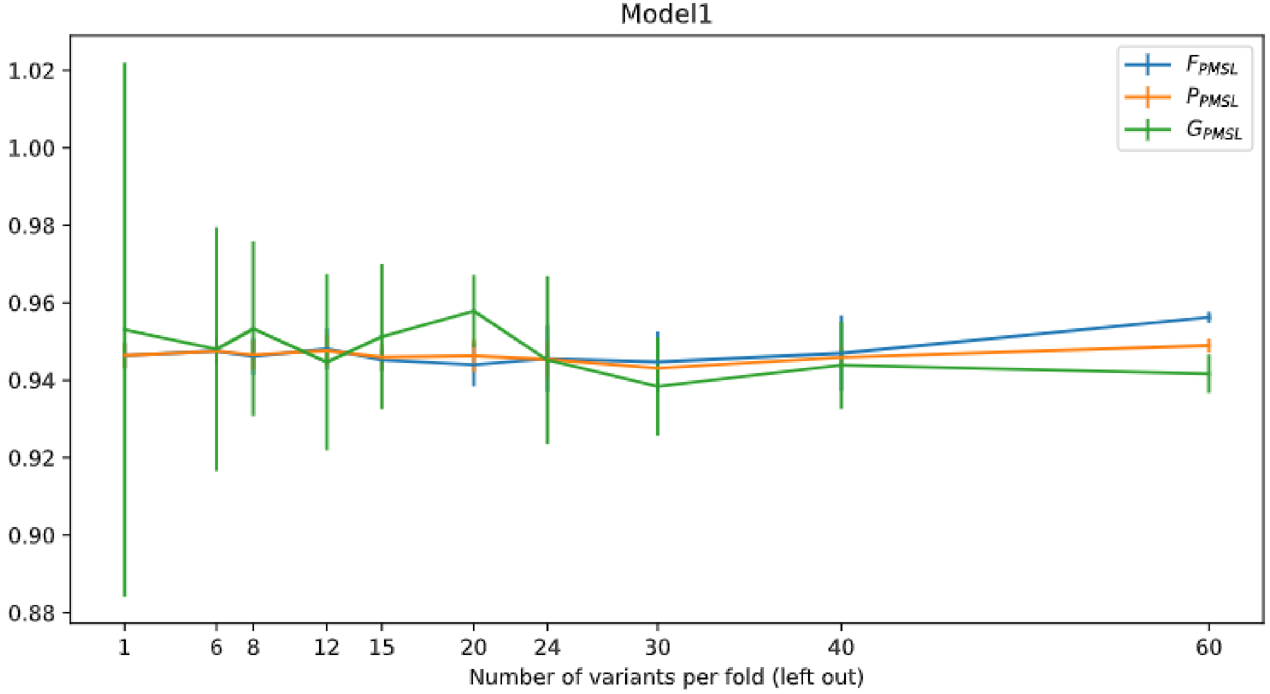
What about hybrid models?



- Optimized LSTM **(with anti-overfitting!)**



- Inductive miner



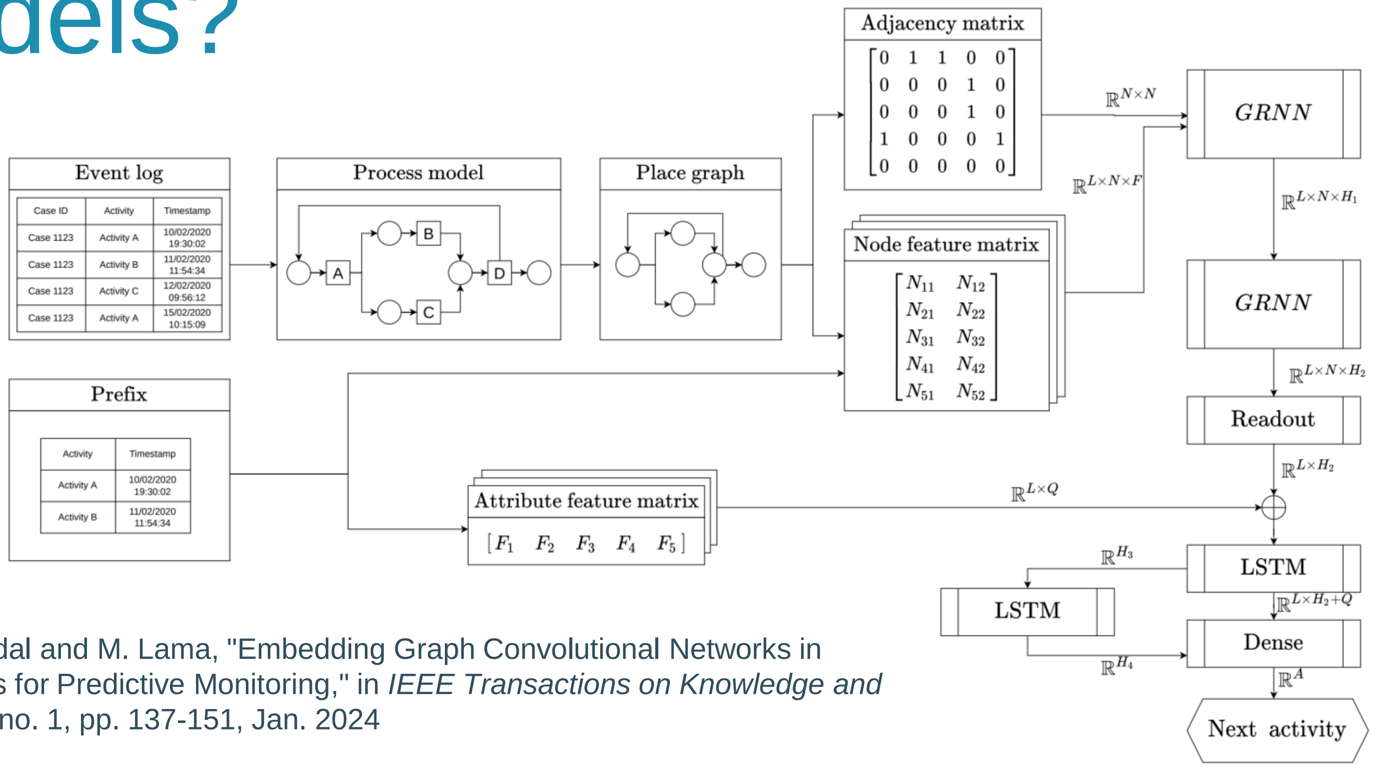
(different scales)

KU LEUVEN

What about hybrid models?

- Case for hybrid models? Discovery + ML?

- Some work
  - E.g. Using node features and adjacency information + prefix in a recurrent neural network



E. Rama-Maneiro, J. C. Vidal and M. Lama, "Embedding Graph Convolutional Networks in Recurrent Neural Networks for Predictive Monitoring," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 36, no. 1, pp. 137-151, Jan. 2024

- Incorporate process-model patterns including e.g. parallelism and loops directly

- Incorporation of discovered process models into deep learning architectures
  - *Local process models?*
  - *Or constraints?*
  - *Tokens based on discovered process model replay?*

KU LEUVEN

Where do LLMs/GenAI fit in all this?

- Can be important in the bigger picture of AI-augmented business process management at different points
  - GenAI can bring context recognition:
    - What types of processes, activities, KPIs are we talking about?
  - LLMs can enable conversational process optimization
  - Retrieval Augmented Generation
    - Or GraphRAG on (object-centric) event logs?
  - …

Where do LLMs/GenAI fit in all this?

- But for PPM… not that much
  - Computationally expensive (<u>monitoring!</u>)
  - Need to incorporate the specific process context

… for now

- Zero-shot learning
- Foundation model
  - “AI process modeler” that knows process constructs
  - Transfer learning
    - Lower data requirements
- …

What about multimodality?

- Isolated view – only process event data

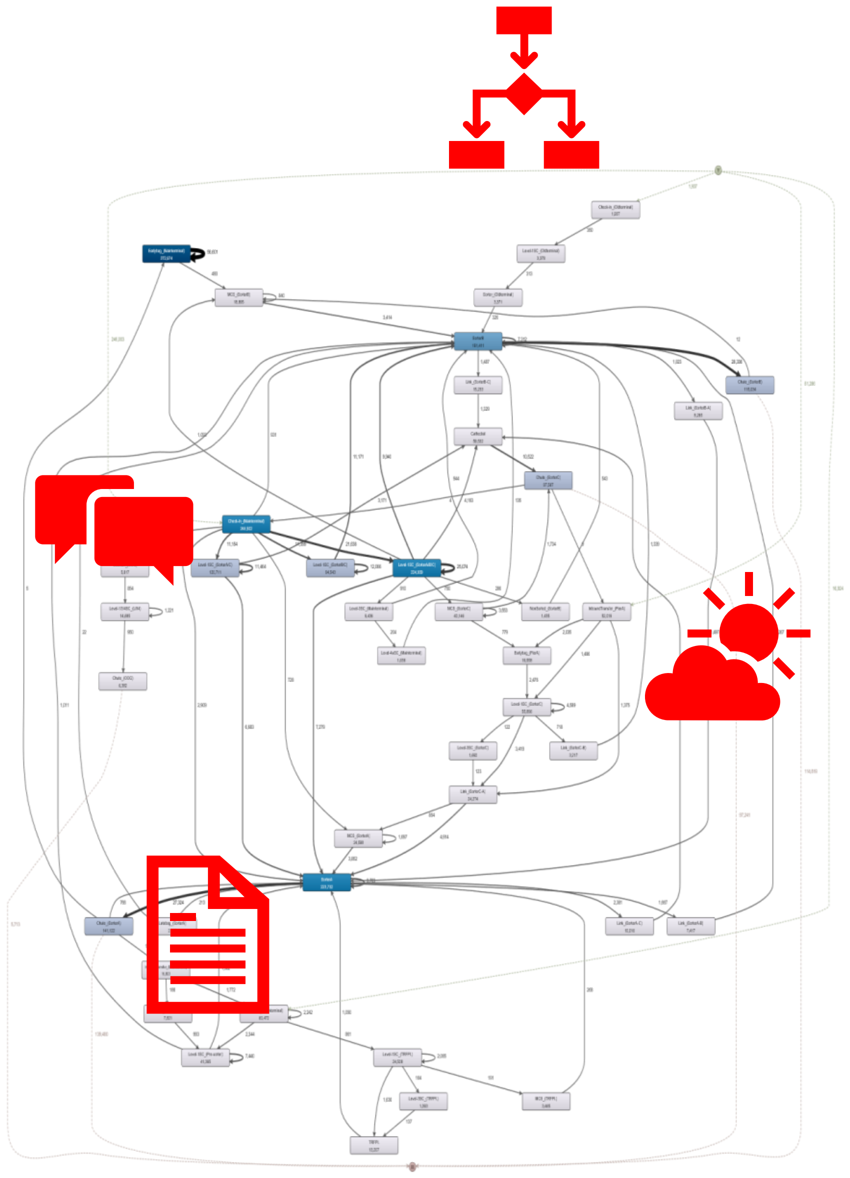
→ Other valuable data sources ignored

- Research gap: multimodal process prediction

→ ML models capable of leveraging multi-modal data sources at once

→ Keep up with trends in deep learning!

- Applications in:
  - Logistics (Brussels airport)
  - Finance (e.g. logs + rules + textual reports)
  - Manufacturing (e.g. logs + sensors + images)



KU LEUVEN

Conclusion

Conclusions

- For the adoption of PPM we need to further uncover…
  1. Adapted encoding techniques or process-specific custom architectures
  2. A uniform and correct way of evaluation
  3. Understand how models interpret process behavior
  4. Take into account a holistic perspective on interpretability and reliability
  5. Include inter-case information

  …

- And we can explore other approaches…
  1. Going from case to process-level predictions
  2. Including object information
  3. Hybrid approaches
  4. Generative AI
  5. Multimodal input data

  …

Questions

Resources

- **Cracking the Nut: Unraveling Challenges in Predictive Process Monitoring, Jochen De Weerdt, keynote ML4PM ICPM 2023**

- Weytjens, H., De Weerdt, J. (2022). Creating Unbiased Public Benchmark Datasets with Data Leakage Prevention for Predictive Process Monitoring. In: A. Marrella, B. Weber (Eds.), Business Process Management Workshops, BPM 2021: vol. 436, (18-29). doi: 10.1007/978-3-030-94343-1_2
- Wuyts, B., Weytjens, H., vanden Broucke, S., and De Weerdt, J. (2023) DyLoPro: Profiling the Dynamics of Event Logs. BPM 2023, 146–162.
- Trace encoding in process mining: A survey and benchmarking. GM. Tavares, RS Oyamada, S Barbon Jr., P Ceravolo. EAAI, 2023.
- Empowering Predictive Process Monitoring through time-related Features. RS Oyamada, GM Tavares, S Barbon Jr., P Ceravolo. CAiSE, 2024.
- Peeperkorn, J., vanden Broucke, S., De Weerdt, J. (2022). Can recurrent neural networks learn process model structure? Journal Of Intelligent Information Systems. doi: 10.1007/s10844-022-00765-x
- Peeperkorn J., vanden Broucke, S. & De Weerdt, J (2024) Validation Set Sampling Strategies for Predictive Process Monitoring, Information Systems, Volume 121
- Stevens, A., De Smedt, J., & Peeperkorn, J. (2022, March). Quantifying explainability in outcome-oriented predictive process monitoring. In Process Mining Workshops: ICPM 2021 International Workshops, Eindhoven, The Netherlands, October 31–November 4, 2021, Revised Selected Papers (pp. 194-206). Cham: Springer International Publishing.
- Stevens, A., & De Smedt, J. (2022). Explainability in Process Outcome Prediction: Guidelines to Obtain Interpretable and Faithful Models (EJOR)
- M. Kabierski, M. Richter and M. Weidlich, "Addressing the Log Representativeness Problem using Species Discovery," 2023 5th International Conference on Process Mining (ICPM), Rome, Italy, 2023

Resources

- Alexander Stevens, Jari Peeperkorn, Johannes De Smedt , Jochen De Weerdt. Assessing the Robustness in Predictive Process Monitoring through Adversarial Attacks. ICPM (2022)
- Alexander Stevens, Jari Peeperkorn, Johannes De Smedt , Jochen De Weerdt. Manifold Learning for Adversarial Robustness in Predictive Process Monitoring. ICPM (2023)
- A Scikit-learn Extension Dedicated to Process Mining Purposes (DEMO Paper). RS Oyamada, GM Tavares, S Barbon Jr., P Ceravolo. CoopIS, 2023.
- P. Ceravolo, S. B. Junior, E. Damiani and W. Van Der Aalst, "Tuning Machine Learning to Address Process Mining Requirements," in IEEE Access, vol. 12, pp. 24583-24595, 2024, doi: 10.1109/ACCESS.2024.3361650
- Weytjens, H., & De Weerdt, J. (2022). Learning uncertainty with artificial neural networks for predictive process monitoring. Applied Soft Computing
- Gunnarsson, B.R., De Weerdt, J. and vanden Broucke, S. (2022). A framework for encoding the multi-location load state of a business process. In 2022 Proceedings of the International IJCAI Workshop on Process Management in the AI era.
- Gunnarsson, Björn & vanden Broucke, Seppe & Weerdt, Jochen. (2023). A Direct Data Aware LSTM Neural Network Architecture for Complete Remaining Trace and Runtime Prediction. IEEE Transactions on Services Computing. PP. 1-13. 10.1109/TSC.2023.3245726.
- Dominique Sommers, Vlado Menkovski, Dirk Fahland, Supervised learning of process discovery techniques using graph neural networks, Information Systems, Volume 115, 2023
- Jari Peeperkorn, Seppe vanden Broucke, Jochen De Weerdt, Global conformance checking measures using shallow representation and deep learning, Engineering Applications of Artificial Intelligence, Volume 123, Part B, 2023
- E. Rama-Maneiro, J. C. Vidal and M. Lama, "Embedding Graph Convolutional Networks in Recurrent Neural Networks for Predictive Monitoring," in IEEE Transactions on Knowledge and Data Engineering, vol. 36, no. 1, pp. 137-151, Jan. 2024