RESEARCH CENTRE FOR INFORMATION SYSTEMS ENGINEERING (LIRIS)

Keynote

Cracking the Nut: Unraveling Challenges in Predictive Process Monitoring

Jochen De Weerdt

jochen.deweerdt@kuleuven.be

ML4PM Workshop at ICPM 2023, Rome (Italy)

About me

- Professor at KU Leuven (Research Centre for Information Systems Engineering)
- PhD in Business Economics (2012)
- Postdoc at the IS School, Queensland University of Technology (QUT), Brisbane (2012-2013)
- Research expertise
  - Process Mining
    - Trace clustering
    - Discovery and conformance checking
    - Predictive Process Monitoring
  - Business Analytics
    - Fraud analytics, learning analytics, real-estate, NLP, marketing, etc.

Successfully supervised PhDs

Pieter De Koninck (2019)
Process mining – trace clustering
AI Lead at Silverfin

Sandra Mitrovic (2019)
Network analytics for churn prediction
Postdoc at IDSIA

Daria Bogdanova (2021)
Feedback in smart learning environments
Customer Manager at Sitecore

Galina Deeva (2021)
Learning analytics – Process mining
Data Scientist at KBC Bank

Rafael Van Belle (2023)
Network analytics for fraud detection
Machine Learning Engineer at Dataroots

Björn Rafn Gunnarsson (2023)
Predictive Process Monitoring with LSTMs
Data Scientist at NATO

Hans Weytjens (2023)
Uncertainty for Predictive Process Monitoring
Senior Researcher at TUM & KU Leuven

Jari Peeperkorn (2023)
Predictive Process Monitoring generalization & conformance
Postdoc at LIRIS

Current team

Philipp Borchert
NLP for business analytics

Margot Geerts
Real estate valuation modelling

Carlos Eduardo Ortega Vázquez
Imbalanced learning for fraud detection

Xin Pang
Process execution visualization

Yongbo Yu
Process Model Forecasting

Brecht Wuyts
Predictive Process Monitoring

Yannis Bertrand
IoT process mining

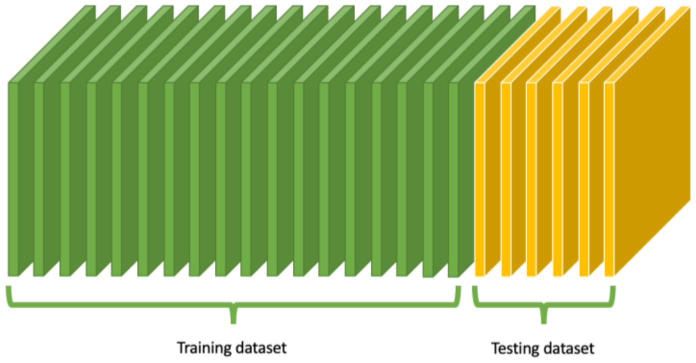
Kevin Biermans
Inter-case Featurization for Predictive Process Monitoring

Jakob De Moor
Prescriptive Process Monitoring

Zahra Ahmadi
IoT process mining

Agenda

- Five key challenges in Predictive Process Monitoring (PPM)
    1. Strategies for PPM evaluation
    2. Generalization in deep learning models
    3. The inter-case perspective
    4. From case-level to model-level predictions
    5. Increasing adoption



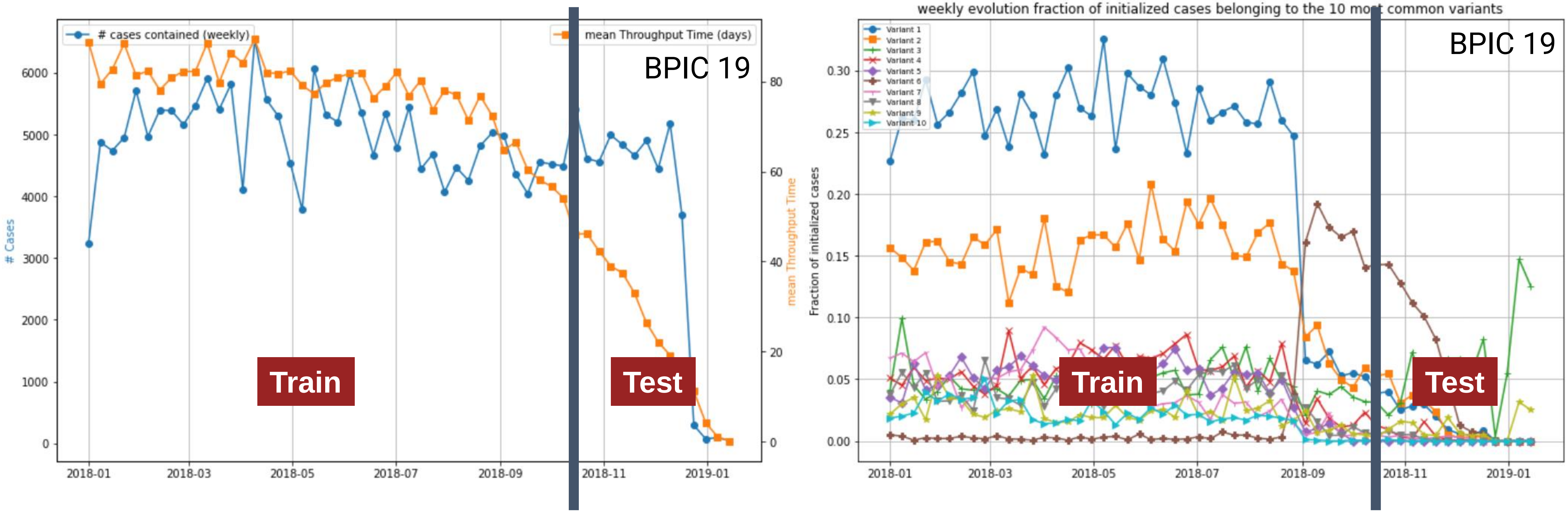
Challenge 1:
Strategies for PPM evaluation

Widespread bad practices in PPM

- 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

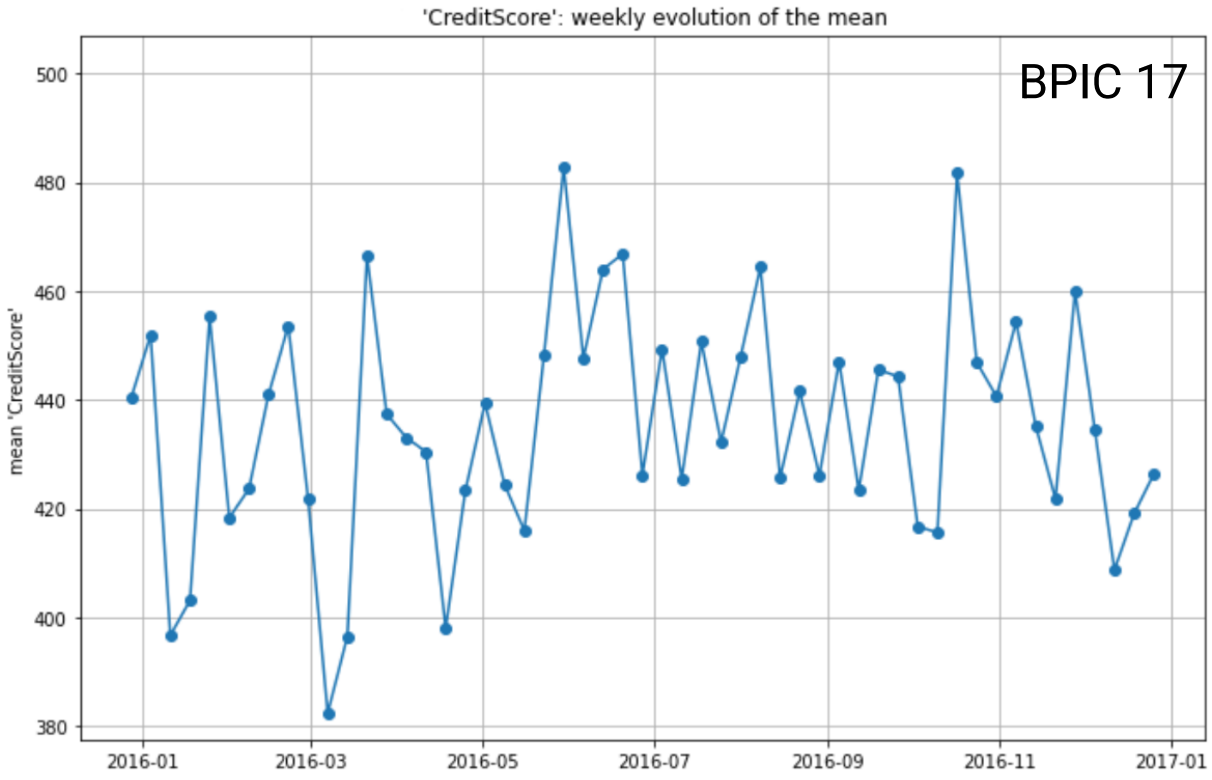
*Abb, L., Pfeiffer, P., Fettke, P., & Rehse, J. R. (2023). A Discussion on Generalization in Next-Activity Prediction. *arXiv preprint arXiv:2309.09618*.

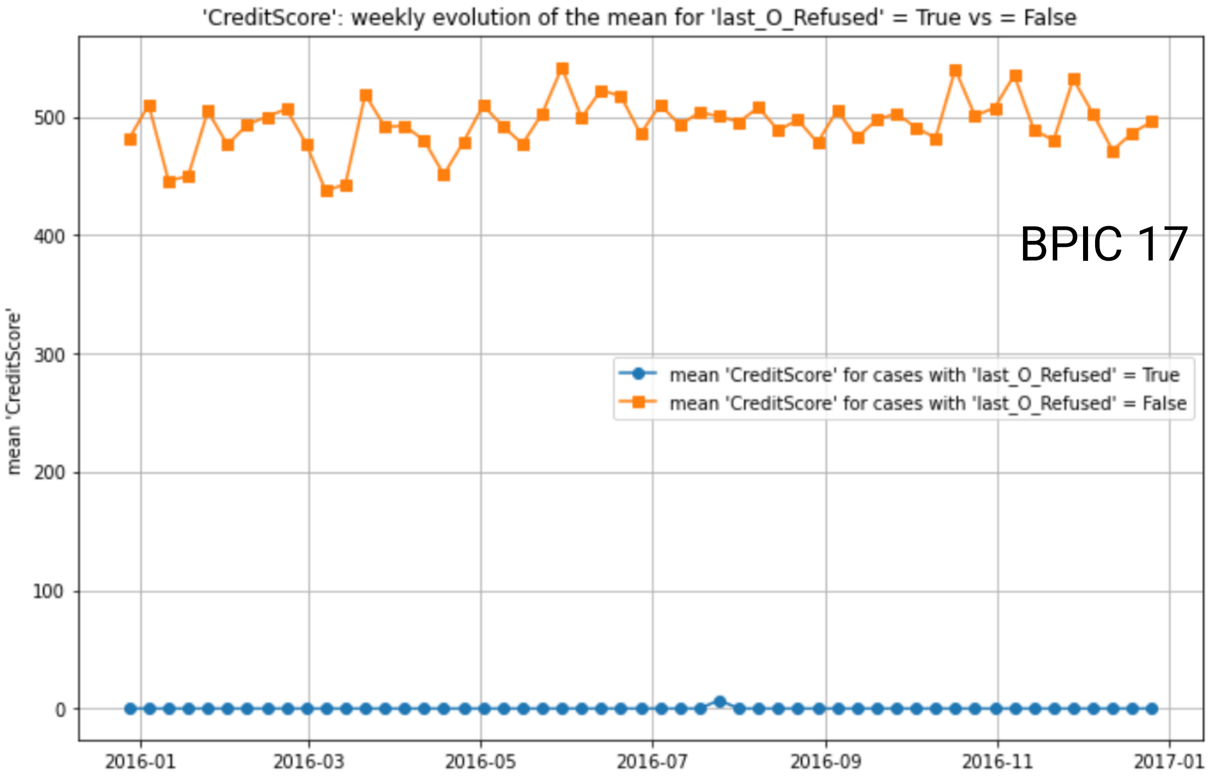
Failing to factor in the dynamics of event logs



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

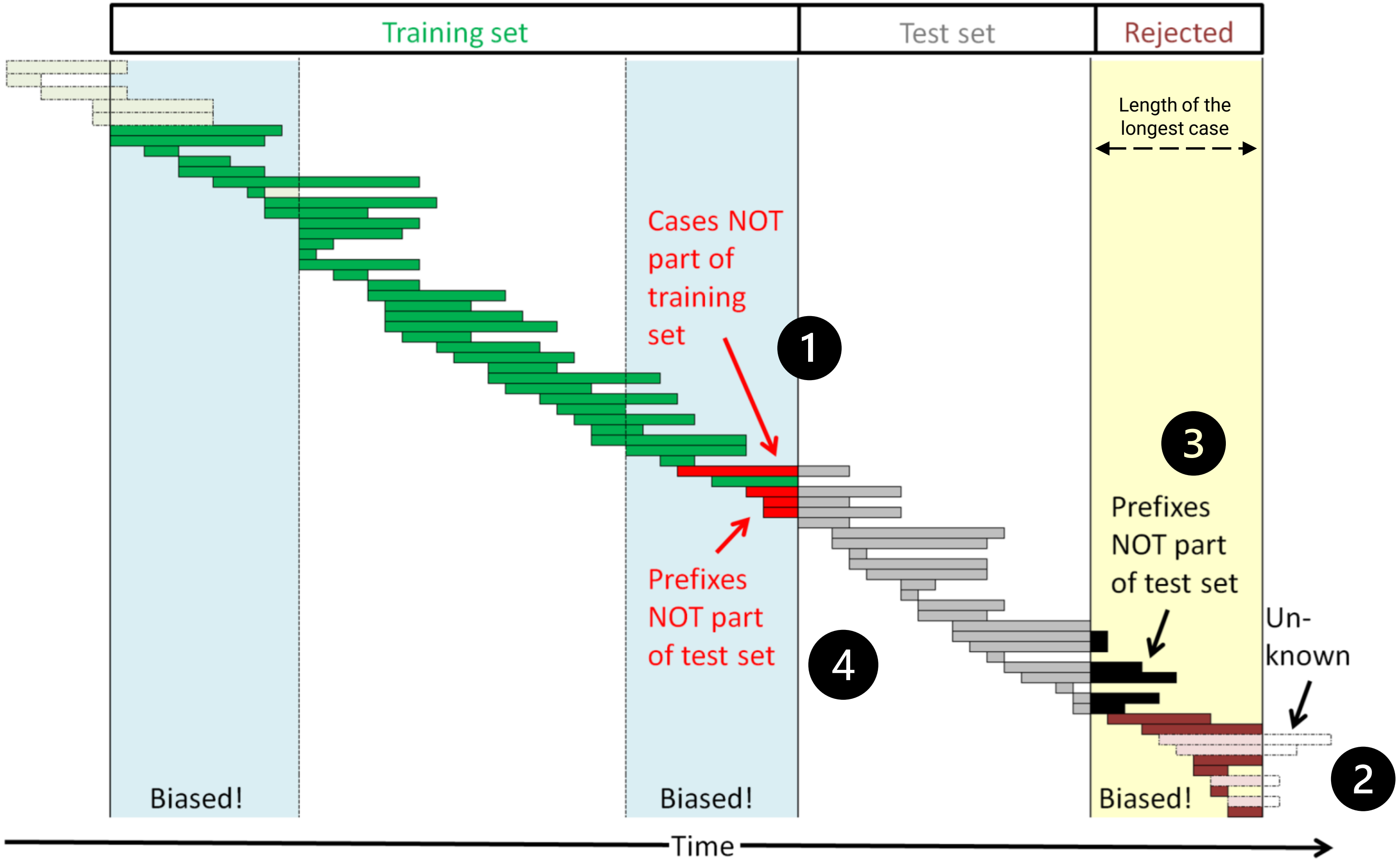
Data leakage





DyLoPro

Removing test set 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)

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

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](https://doi.org/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

Learn from other domains

CORE: A Few-Shot Company Relation Classification Dataset for Robust Domain Adaptation.

Philipp Borchert[1,2], **Jochen De Weerdt**[2], **Kristof Coussement**[1], **Arno De Caigny**[1], **Marie-Francine Moens**[3]
[1]IESEG School of Management, Univ. Lille, CNRS, UMR 9221 - LEM - Lille Economie Management, F-59000 Lille, France
[2]Research Centre for Information Systems Engineering, KU Leuven, Belgium
[3]Department of Computer Science, KU Leuven, Belgium

1 [cs.CL] 18 Oct 2023

Abstract

We introduce CORE, a dataset for few-shot relation classification (RC) focused on company relations and business entities. CORE includes 4,708 instances of 12 relation types with corresponding textual evidence extracted from company Wikipedia pages. Company names and business entities pose a challenge for few-shot RC models due to the rich and diverse information associated with them. For example, a company name may represent the legal entity, products, people, or business divisions depending on the context. Therefore, deriving the relation type between entities is highly dependent on textual context. To evaluate the performance of state-of-the-art RC models on the CORE dataset, we conduct experiments in the few-shot domain adaptation setting. Our results reveal substantial performance gaps, confirm-

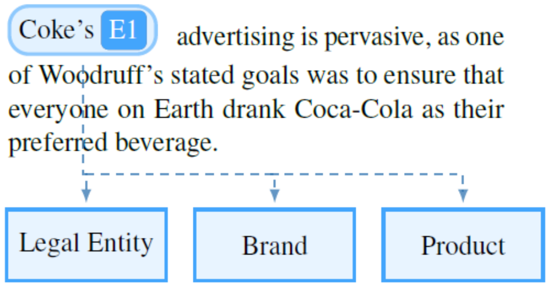


Figure 1: Example for information richness embedded in named business entities.

et al., 2019; Sabo et al., 2021), and out-of-domain adaptation (Gao et al., 2019). However, most existing RC datasets comprise a mixture of domains, and the sets of entities across different domains are often easily distinguishable from one another. The

EMNLP 2023

≡ FewRel   Home FewRel 1.0 FewRel 2.0 Domain Adaptation FewRel 2.0 None-of-the-Above Detection

Welcome to ***FewRel***

a ***Few***-shot ***Rel***ation classification dataset

We have now moved to [Codalab competition].

Paper GitHub

What is FewRel?

FewRel is a **Few**-shot **Rel**ation classification dataset, which features 70, 000 natural language sentences expressing 100 relations annotated by crowdworkers.

Please refer to our EMNLP 2018 paper to learn more about FewRel.

Han, Zhu, Yu, Wang, et al., 2018

We add two more challenging settings: few-shot domain adaptation (DA) and few-shot none-of-the-above detection (NOTA) in FewRel 2.0 dataset. Please refer to our EMNLP 2019 paper for more details.

Gao et al., 2019

Baseline models and a series of toolkits are released in this repo:

thunlp/FewRel

Get Started

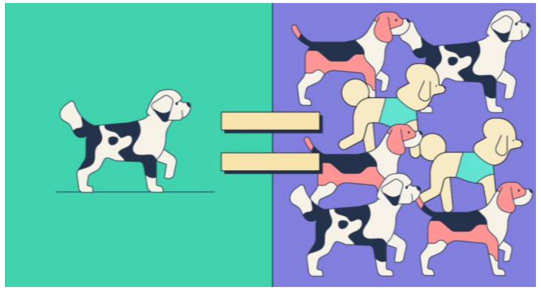
It is fairly easy for you to get started with **FewRel**. First, you can download a copy of the dataset by the following links. The dataset is distributed under the CC BY-SA 4.0 license:

Leaderboard

#	Model	Domain Adaptation 5-way-1-shot	Domain Adaptation 5-way-5-shot	Domain Adaptation 10-way-1-shot	Domain Adaptation 10-way-5-shot
1	GTP **Anonymous** (Aug 24, 2020)	80.04	92.58	69.25	86.88
2	DualGraph **Anonymous** (Aug 2, 2020)	80.11	91.01	73.89	82.34
3	PAMN **Anonymous** (Jun 8, 2020)	77.54	90.40	65.98	82.03
4	Anonymous Pony (Jan 17, 2020)	76.71	86.69	66.72	78.46
5	Anonymous Python (Dec 25, 2019)	66.41	83.52	51.85	73.60
6	Anonymous Groundhog (Dec 25, 2019)	67.23	82.09	54.32	71.01
7	BERT-PAIR[paper][code] **THUNLP, Tsinghua University** (Nov 3, 2019)	67.41	78.57	54.89	66.85

Challenge 1: 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)

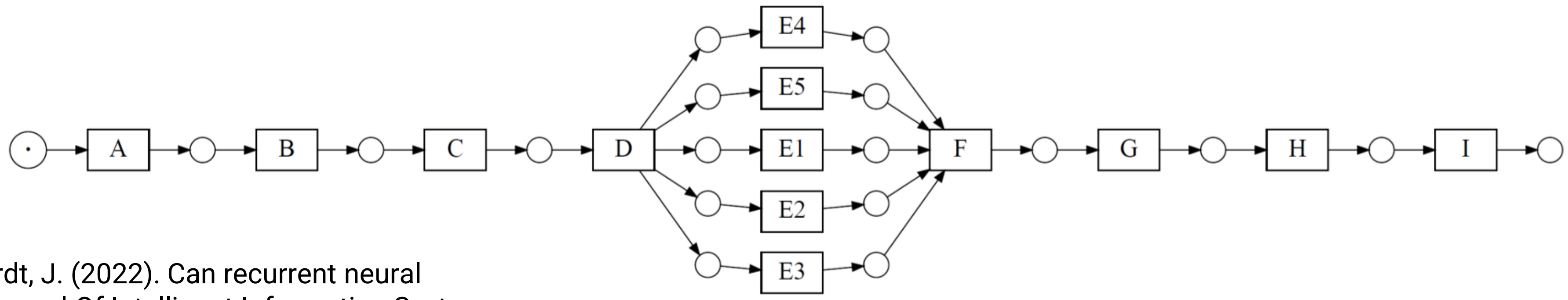


Challenge 2:
Generalization in deep learning models

Key question

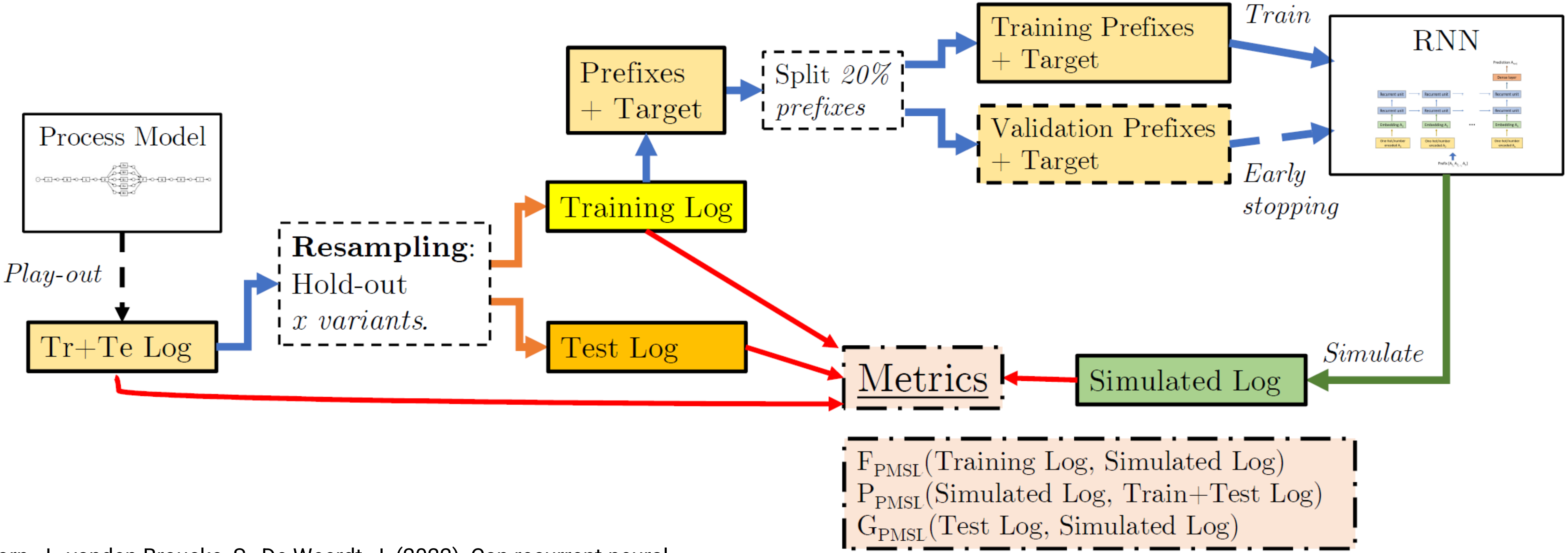
- How capable are deep learning models to generalize process behavior?

- 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](https://doi.org/10.1007/s10844-022-00765-x)

Leave One Variant Out Cross-Validation (LOVOCV)



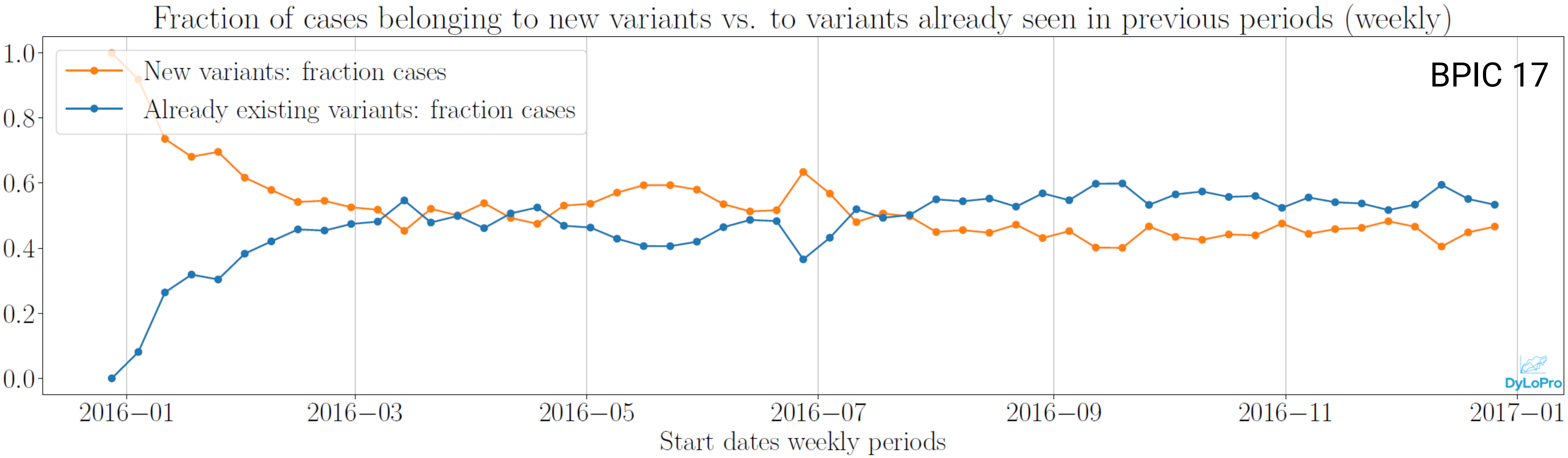
Peeperkorn, J., vanden Broucke, S., De Weerdt, J. (2022). Can recurrent neural networks learn process model structure? Journal Of Intelligent Information Systems. [doi: 10.1007/s10844-022-00765-x](https://doi.org/10.1007/s10844-022-00765-x)

Results

- We built LSTMs to predict the next activity
- Optimized hyperparameters: nr. layers, hidden unit size, dropout rate, L1/L2 regularization

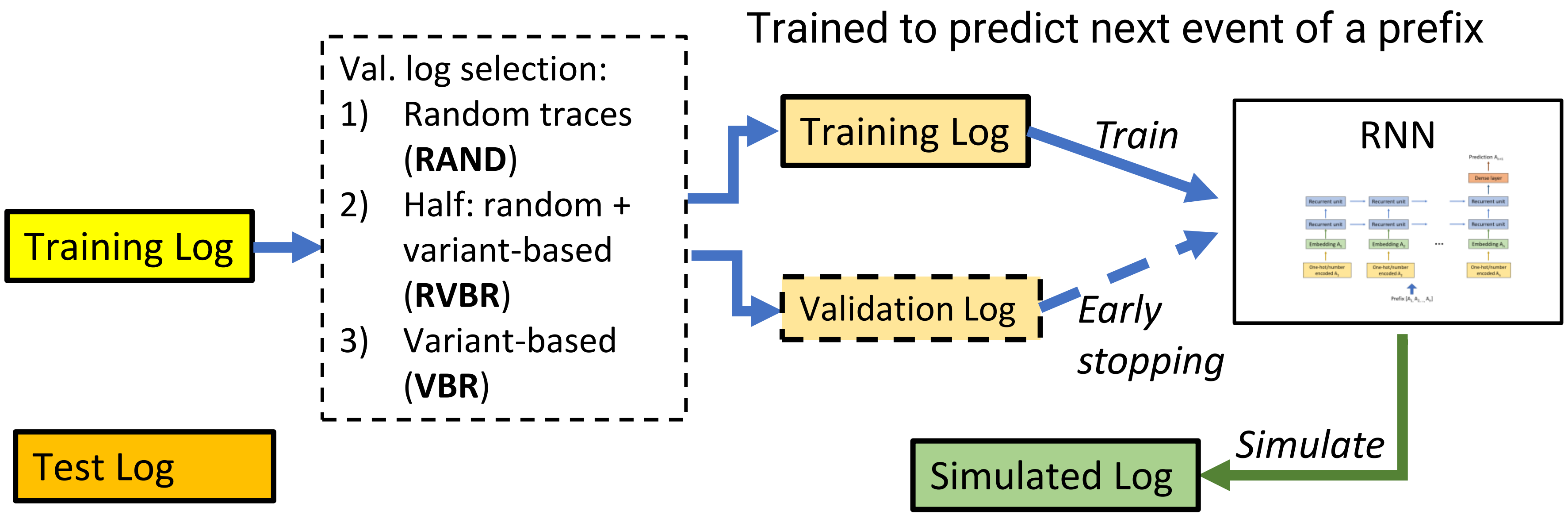
Mod.	Pattern	#Var.	Acc. based Hyperparameters F_{PMSL}	P_{PMSL}	G_{PMSL}
1	PAR	120	0.96 ± 0.01	0.95 ± 0.01	0.00 ± 0.01
2	XOR	128	0.94 ± 0.01	0.94 ± 0.01	0.80 ± 0.14
3	XOR+LTD	128	0.95 ± 0.01	0.94 ± 0.00	0.04 ± 0.17
4	IOR	64	0.93 ± 0.02	0.92 ± 0.02	0.66 ± 0.17
5	PAR	126	0.95 ± 0.02	0.94 ± 0.02	0.07 ± 0.20
6	LOOP	27	0.87 ± 0.02	0.86 ± 0.02	0.75 ± 0.28

Is generalization important?



Can we solve the problem?

- Better validation set sampling strategies can help:



Peeperkorn J., vanden Broucke, S., & De Weerdt, J. Validation Set Sampling Strategies for Predictive Process Monitoring, *Under Review*

Insights

- On 6 simple process models, we show that we can increase generalization at a little cost in fitness/precision
  - See table

- If models become too complex, the variant-based resampling becomes less effective

- Also important effect of event log incompleteness, but not yet fully understood

Model	Val.	Test Acc.	F_{PMSL}	P_{PMSL}	G_{PMSL}
1	**RAND**	0.726	0.982	0.889	0.000
	RVBR	0.791	0.885	0.873	0.764
	VBR	0.759	0.872	0.845	0.591
2	**RAND**	0.749	0.932	0.917	0.772
	RVBR	0.817	0.880	0.878	0.860
	VBR	0.813	0.815	0.815	0.814
3	**RAND**	0.735	0.975	0.880	0.0
	RVBR	0.810	0.884	0.877	0.813
	VBR	0.812	0.876	0.870	0.817
4	**RAND**	0.794	0.931	0.895	0.532
	RVBR	0.851	0.865	0.841	0.593
	VBR	0.799	0.877	0.853	0.607
5	**RAND**	0.798	0.988	0.881	0.0
	RVBR	0.843	0.813	0.795	0.650
	VBR	0.850	0.838	0.805	0.538
6	**RAND**	0.844	0.840	0.841	0.868
	RVBR	0.862	0.863	0.862	0.838
	VBR	0.862	0.844	0.840	0.779

Challenge 2: Key takeaways

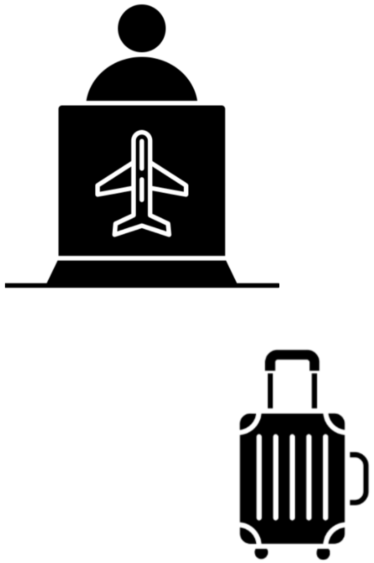
- We need to further investigate validation set sampling techniques
  - Factoring in the data perspective (case + event features)

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

Challenge 3:
The inter-case perspective

Motivation

- Research in predictive process monitoring has generally relied on **intra-case features** in order to make predictions
- Therefore, assume that the processing of 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**



Motivation

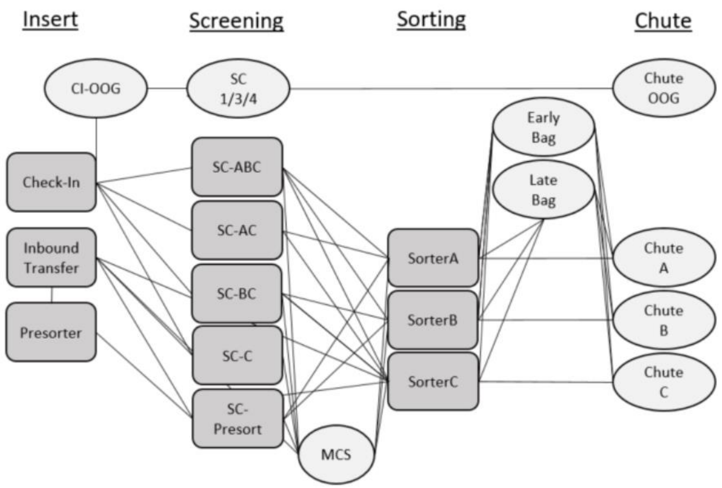
- Research in predictive process monitoring has generally relied on **intra-case features** in order to make predictions
- Therefore, assume that the processing of case is solely dependent on the attributes of the case itself
- However, cases are not processed in isolation
    - Can be influenced by 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**

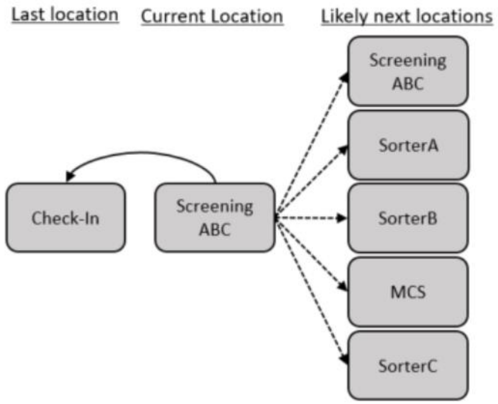
Examples of PPM research including an intercase perspective

- 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.

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)





Remaining trace prediction performance

Additional value of the MLS-ICE framework for remaining trace prediction

- Consistent performance gain for models that use features encoded using the proposed MLS-ICE framework
- Up to 5.1% compared to solely relying on intra-case features
- Up to 5.7% compared to using the "Senderovich" feature vector

Table 4.6: Prediction results for remaining trace prediction as evaluated using the normalized Levenshtein similarity. Models that use inter-case features during prediction and outperform a model that solely relies on using intra-case features in order to make predictions are given in bold. An underscore is used to indicate the overall best performing model per event logs. The performance differences between models that use inter-case features and the baseline model that solely relies on intra-case features in order to make prediction are given within brackets.

Feature vector	BPIC20	BPIC17	BPIC12	BPIC12-Sub	BAG
Intra-case	0.9326 (-)	0.6125 (-)	0.5030 (-)	0.6011 (-)	0.7166 (-)
Senderovich vector					
Last activity	0.9313 (−0.1%)	0.6100 (−0.3%)	**0.5053** (+0.2%)	0.5953 (−0.6%)	**0.7204** (+0.4%)
Case based MLS-ICE vector					
Act. Cases	0.9326 (0.0%)	**0.6137** (+0.1%)	**0.5465** (+4.4%)	**0.6495** (+4.8%)	**0.7276** (+1.1%)
Opt. window	0.9281 (−0.5%)	0.6086 (−0.4%)	0.5030 (0.0%)	**0.6178** (+1.7%)	**0.7259** (+0.9%)
System based MLS-ICE vector					
Act. cases	**0.9334** (+0.1%)	**0.6146** (+0.2%)	**0.5385** (+3.6%)	**0.6524** (+5.1%)	**0.7217** (+0.5%)
Opt. window	0.9313 (−0.1%)	**0.6144** (+0.2%)	0.4945 (−0.9%)	**0.6356** (+3.5%)	**0.7250** (+0.8%)
System and case based MLS-ICE vector					
Act. cases	**0.9327** (0.0%)	**0.6161** (+0.4%)	0.5002 (−0.3%)	**0.6217** (+2.1%)	**0.7292** (+1.3%)
Opt. window	**0.9327** (0.0%)	0.6103 (−0.2%)	0.5020 (−0.1%)	**0.6454** (+4.4%)	**0.7260** (+0.9%)

Remaining time prediction performance

Additional value of the MLS-ICE framework for remaining time prediction

- Consistent performance gain for models that use features encoded using the proposed MLS-ICE framework
- Up to 9.5% compared to solely relying on intra-case features
- Up to 6.9% compared to using the “Senderovich” feature vector

Table 4.7: Prediction results for remaining runtime prediction evaluated using MAE. Models that use inter-case features during prediction and outperform a model that solely relies on using intra-case features in order to make predictions are given in bold. An underscore is used to indicate the overall best performing model per event log. The performance differences between models that use inter-case features and the baseline model that solely relies on intra-case features in order to make prediction are given within brackets.

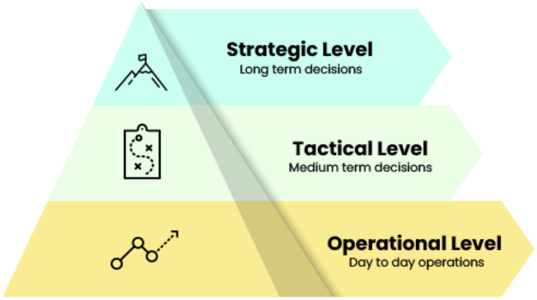
Feature vector	BPIC20	BPIC17	BPIC12	BPIC12-Sub	BAG (sec.)
Intra-case	6206 (-)	11882 (-)	5219 (-)	5936 (-)	526 (-)
Senderovich vector					
Last activity	6241 (−0.5%)	**11672** (+1.8%)	5404 (−3.5%)	**5541** (+6.7%)	**525** (+0.2%)
Case based MLS-ICE vector					
Act. cases	6279 (−1.2%)	**<u>11667</u>** (+1.8%)	5236 (−0.3%)	**5549** (+6.5%)	**523** (+0.5%)
Opt. window	6228 (−0.4%)	**11756** (+1.1%)	**5179** (+0.8%)	**5774** (+2.7%)	**<u>520</u>** (+1.1%)
System based MLS-ICE vector					
Act. cases	6273 (−1.1%)	**11670** (+1.8%)	**<u>5029</u>** (+3.6%)	**5403** (+9.0%)	527 (−0.2%)
Opt. window	6234 (−0.5%)	**11676** (+1.7%)	**5156** (+1.2%)	**5472** (+7.8%)	**522** (+0.8%)
System and case based MLS-ICE vector					
Act. cases	6240 (−0.5%)	**11827** (+0.4%)	**5169** (+1.0%)	**5507** (+7.2%)	**522** (+0.8%)
Opt. window	**<u>6164</u>** (+0.7%)	**11752** (+1.1%)	5406 (−3.6%)	**<u>5375</u>** (+9.5%)	**521** (+1.0%)

Challenge 3: Key takeaways

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

- Inter-case featurisation and prediction requires:
    1. Even more robust evaluation setups, cfr. challenge 1
        - Debiased test set
    2. Capable model learning architectures required (high-dimensional, dynamic event attributes)
        - Transformers?

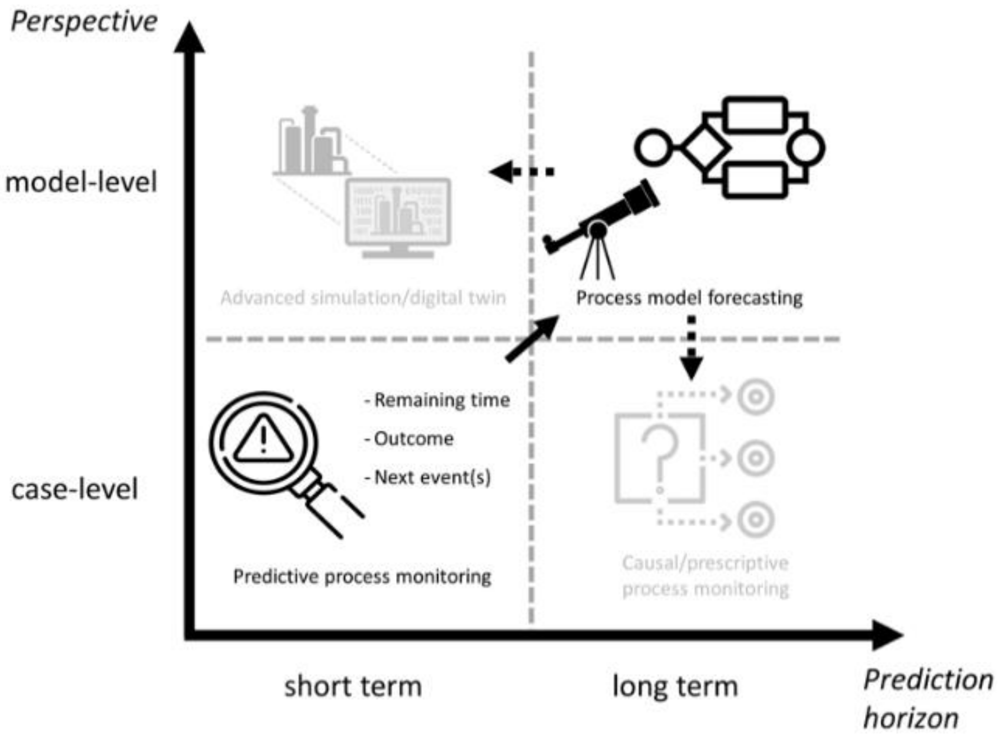
- How to characterize the system?
    - What about context? E.g. IoT
    - What about the object-centric perspective?
    - What about interprocess dependencies?

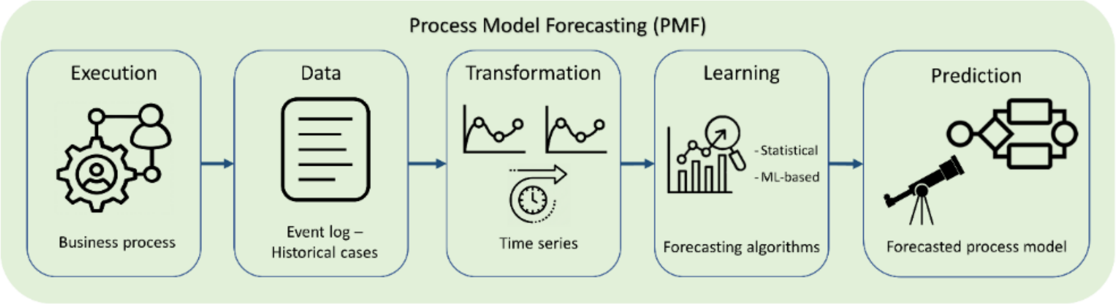


Challenge 4: From case-level to model-level predictions

Process Model Forecasting (PMF)

- Shift from operational to tactical decision support





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](https://doi.org/10.1016/j.datak.2023.102145)

Turning event logs into DF 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

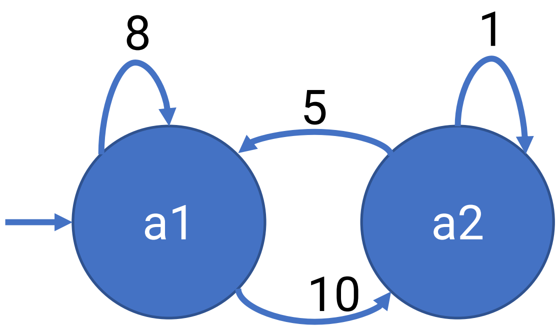
Directly-follows	Equitemporal	Equisized
$<_{L_s} (a_1, a_1)$	[0,1,0]	[1,0,0]
$<_{L_s} (a_1, a_2)$	[1,1,1]	[1,1,1]
$<_{L_s} (a_2, a_1)$	[0,1,0]	[0,1,0]
$<_{L_s} (a_2, a_2)$	[0,0,1]	[0,0,1]

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

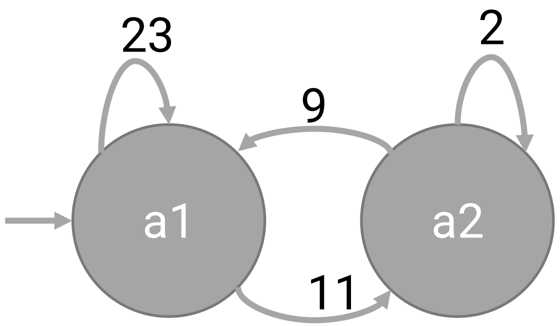
Equisized: 9 events: 3 intervals of 3 events

“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]

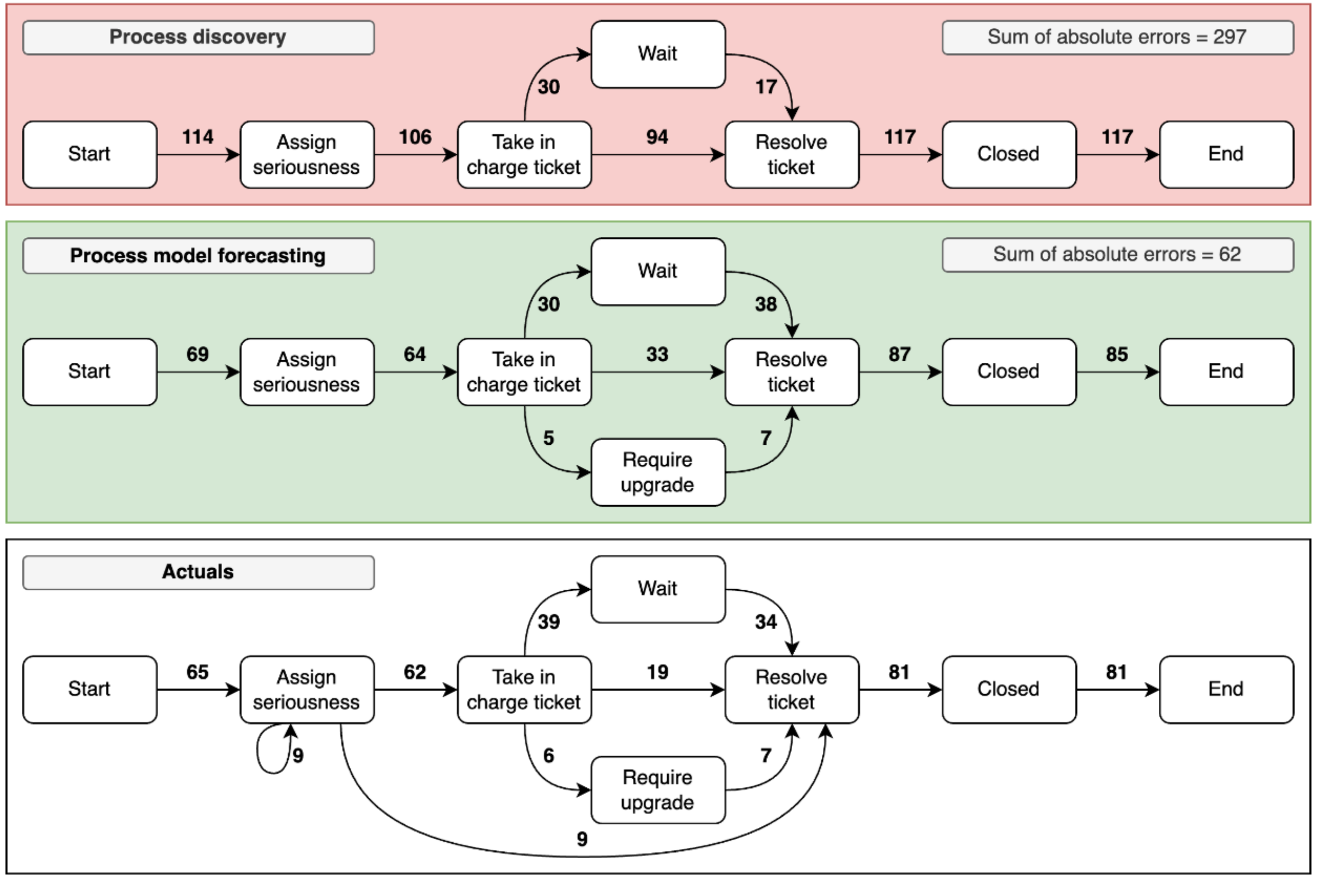


DFG of the past



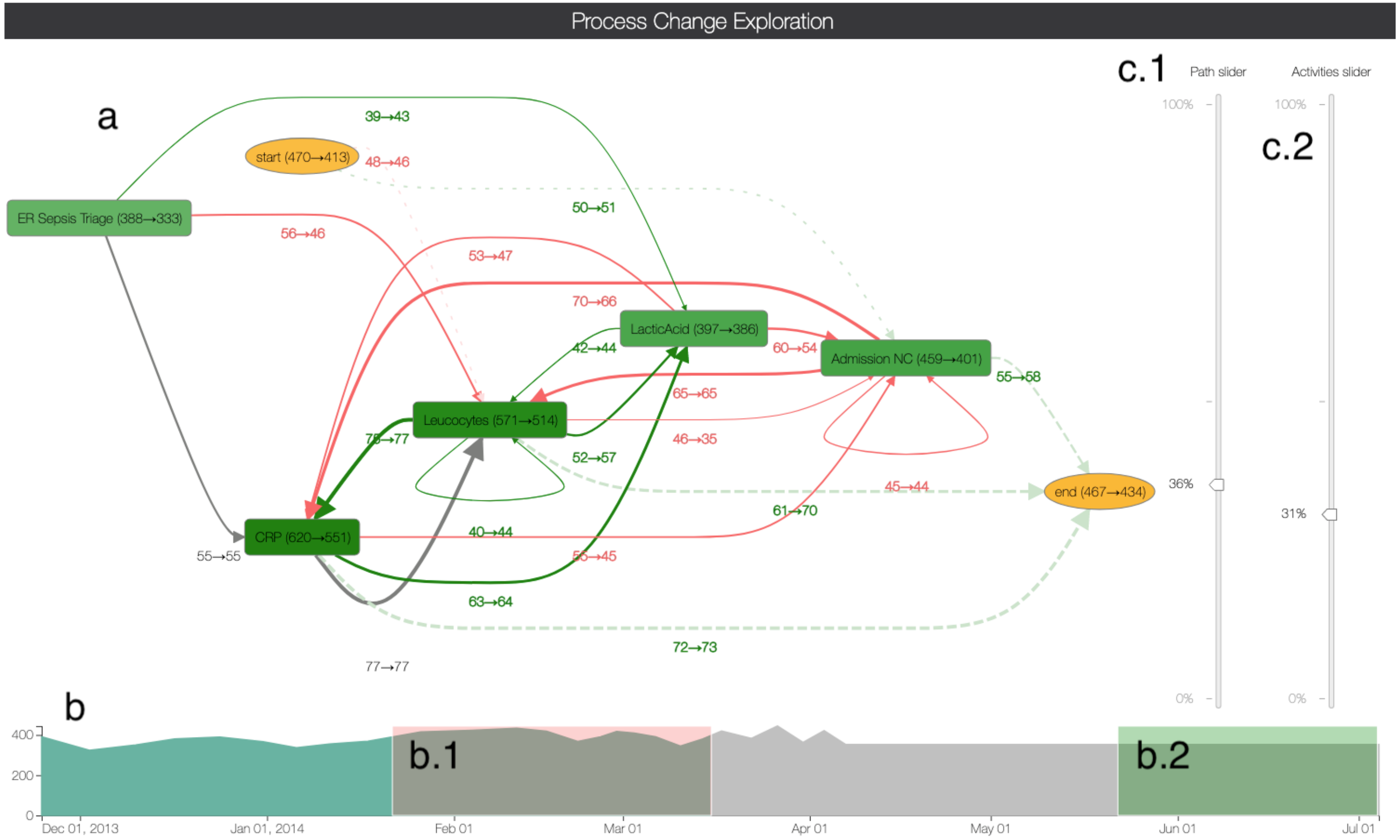
DFG of the future

Process Model Forecasting



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

Process Change Exploration tool



Challenge 4: Key takeaways

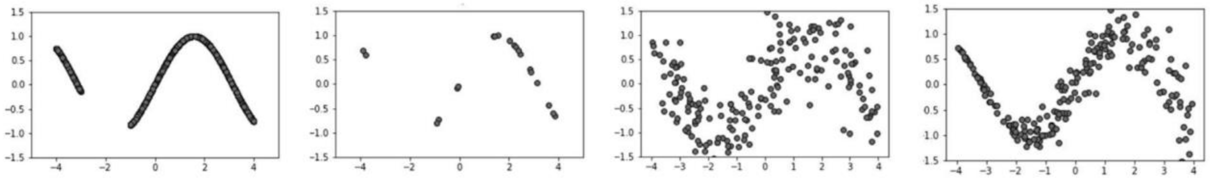
- Process Model Forecasting (PMF) can predict the to-be process model (in the future)
- Could yield opportunities to apply predictive modelling at a different level of granularity
  - Answering different questions
- Current technique relies on simple univariate time series modelling

Challenge 5: Increasing adoption

Pathways to increased adoption

1. Uncertainty
2. Interpretability
3. Robustness
4. Prescriptive process monitoring
5. Data scarcity

1. Uncertainty

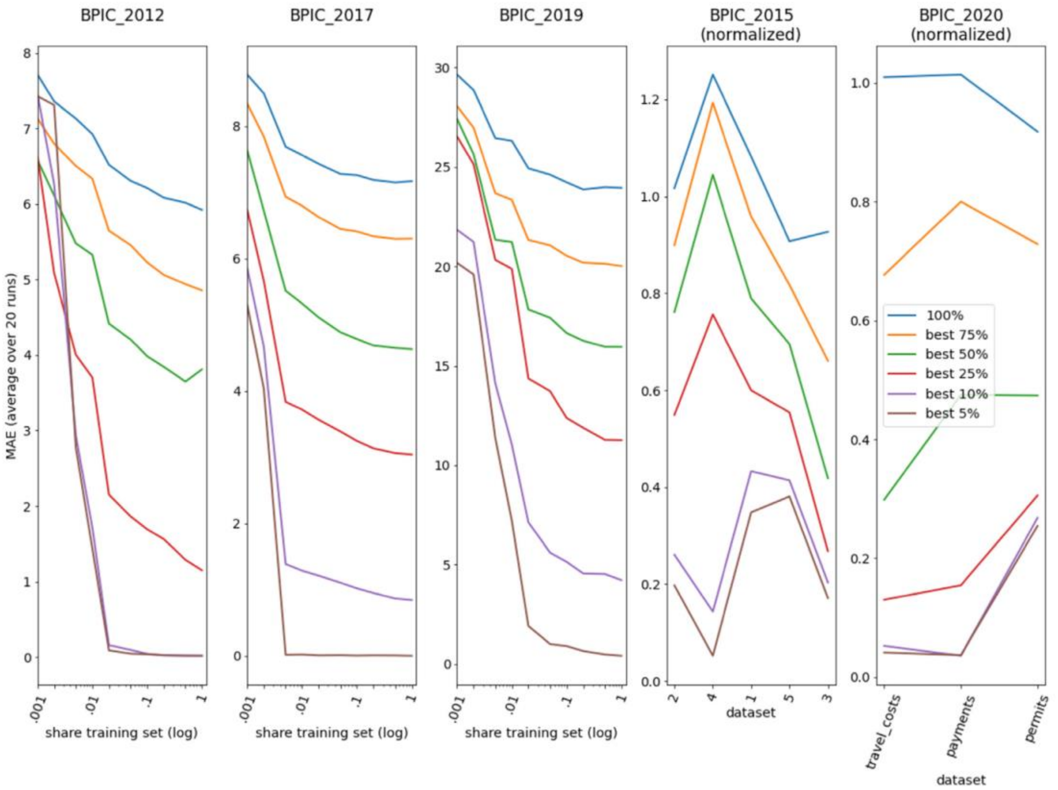


(a) Epistemic uncertainty: example 1. (b) Epistemic uncertainty: example 2. (c) Aleatoric uncertainty: homoscedastic. (d) Aleatoric uncertainty: heteroscedastic.

- Bayesian Neural Networks
  - Allow to estimate epistemic and aleatoric uncertainty
  - We 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

- 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

2. Interpretability

- XAI tailored to PPM seems a must to have any chance at improved adoption
- Only few works in PPM already addressed 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.

3. Robustness

- We need to build trust
- Show that PPM models are robust
- E.g. robustness against adversarial attacks
  - See: Stevens, A., De Smedt, J., Peeperkorn, J., & De Weerdt, J. (2022). Assessing the Robustness in Predictive Process Monitoring through Adversarial Attacks. In 2022 4th International Conference on Process Mining (ICPM) (pp. 56-63). IEEE.

- Adversarial training as a proactive defense mechanism
  - Stevens, A., Peeperkorn, J., De Smedt, J., & De Weerdt, J. (2023). Manifold Learning for Adversarial Robustness in Predictive Process Monitoring. In 2023 5th International Conference on Process Mining (ICPM) (pp. 17-24). IEEE.
  - Warmly invited to attend our presentation on Tuesday at 2pm

4. Prescriptive process monitoring

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

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

- Industry Keynote Marlon Dumas
  - “Walking the Way from Process Mining to AI-Driven Process Optimization”
  - Wednesday at 1.30pm

5. Data scarcity

- Current datasets often used in PPM research are not necessarily very significant to the problem
  - E.g. labels used for outcome prediction
- If we want to grow as a community, we should take inspiration from related fields
  - In ML domains ranging from NLP to graph learning, there exist a wide variety of benchmark datasets and agreed upon evaluation approaches
- This should bring not only better techniques/models, but also a better recognition in practice
- Data privacy and other AI regulations might have an important negative impact, with a large number of companies already totally opposed to any form of data sharing

Conclusion

Conclusion

- Five key challenges in Predictive Process Monitoring (PPM)
    1. Strategies for PPM evaluation
    2. Generalization in deep learning models
    3. The inter-case perspective
    4. From case-level to model-level predictions
    5. Increasing adoption

Questions