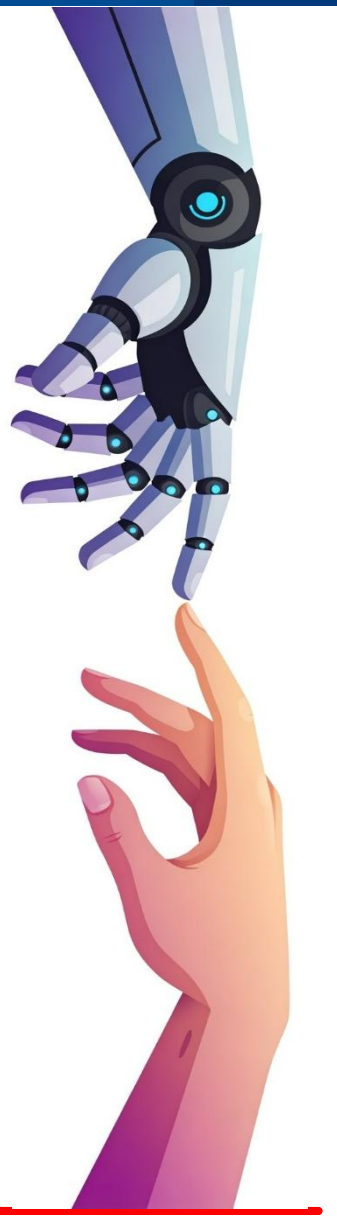
AI-Assisted Prescriptive Business Process Monitoring



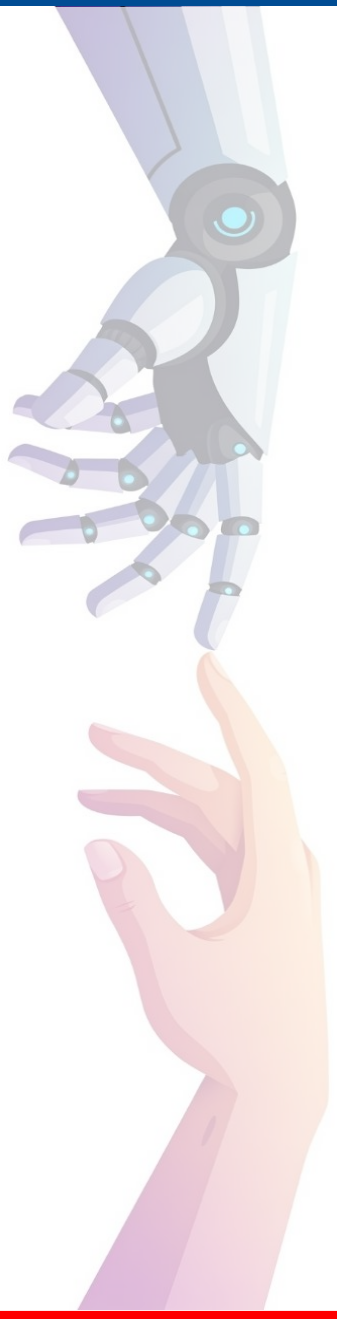
Andreas Metzger

Dagstuhl, May 04 – May 09, 2025

Presentation based on A. Metzger et al.:
Automatically reconciling the trade-off between prediction accuracy and earliness in prescriptive business process monitoring.
Information Systems, 2023, 102254; https://doi.org/10.1016/j.is.2023.102254

Agenda

1. Background and Motivation
2. Solutions
3. Comparative Evaluation
4. Practical Recommendations
5. Outlook



Predictive vs Prescriptive BPMon

Predictive BPMon

“What will happen and when?”

- Use process monitoring data “to forecast how a running process instance will unfold” [Pfeiffer et al. 2025 @ BISE]
- *E.g., will order-to-cash process complete successfully?*

Prescriptive BPMon

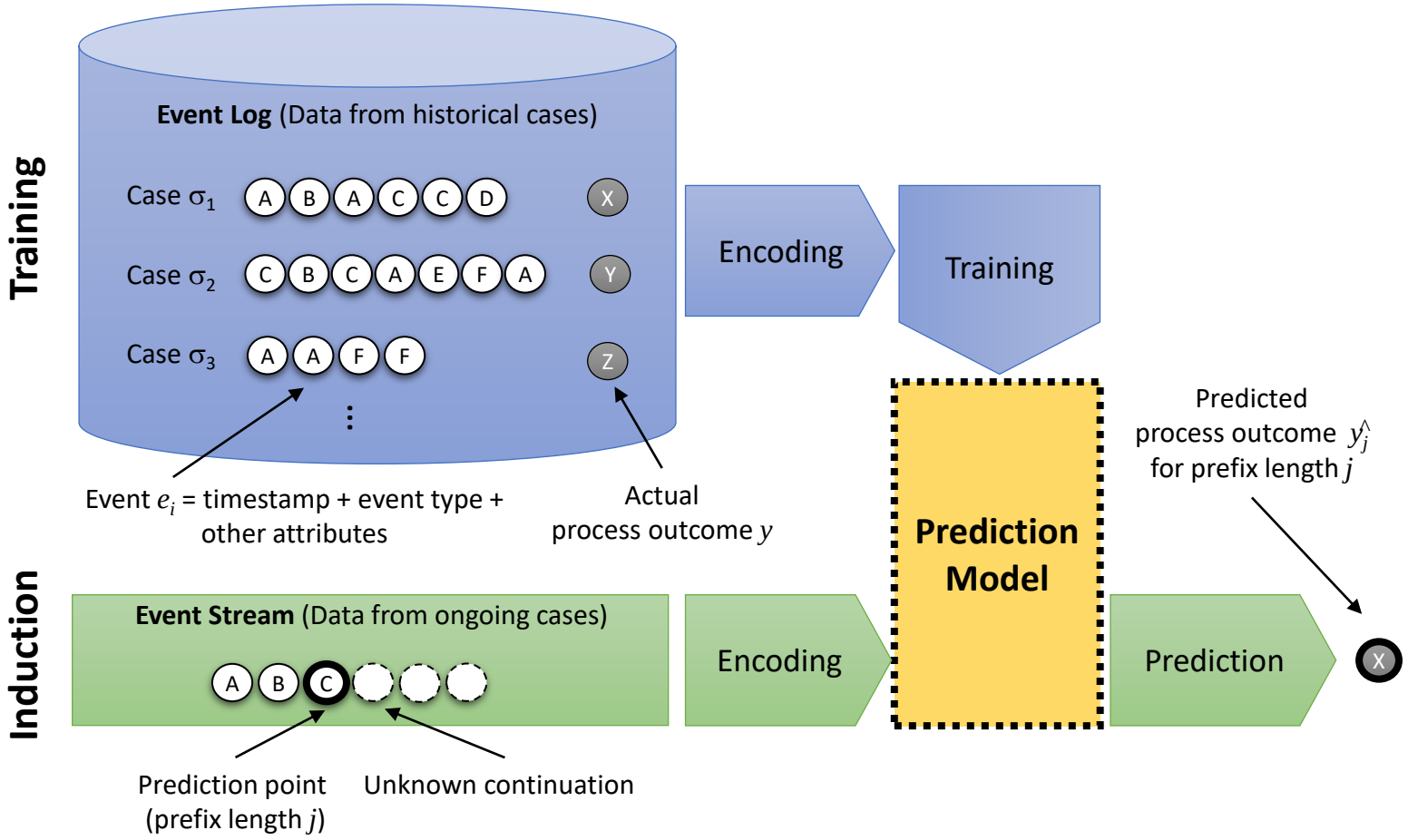
“When to intervene and how?”

- Assist process managers by raising alarms to ”trigger proactive process adaptations”
- *E.g., schedule air delivery instead of road delivery to ensure timely completion of transport process*

[Pfeiffer et al. 2025 @ BISE]: https://link.springer.com/article/10.1007/s12599-025-00936-4
[Metzger et al. 2023 @ BPM]: https://doi.org/10.1007/978-3-030-58666-9_16

Predictive BPMon

AI-based (ML-based) prediction



Predictive BPMon

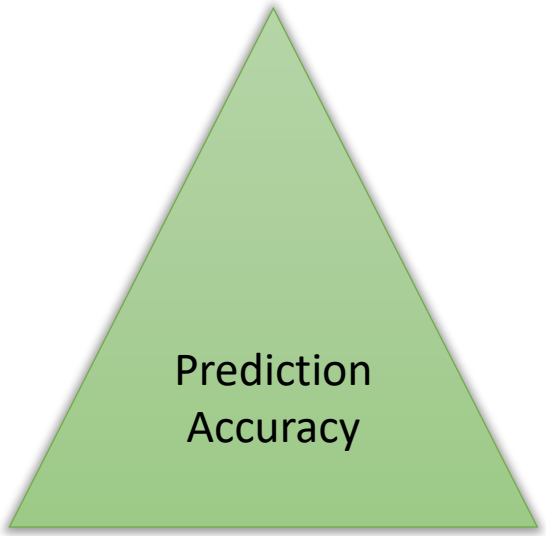
AI-based (ML-based) prediction

Prediction accuracy

- "predict as many **true** deviations as possible, while predicting as few **false** deviations as possible"

Typical ML models

- Decision trees
- Linear regression
- Random forests = ensembles of decision trees (e.g., gradient boosted trees)
- Deep artificial neural networks (e.g., RNNs, such as LSTMs)



Prescriptive BPMon

Problem Statement:

Which prediction to trust and act upon, i.e., when to raise an alarm?

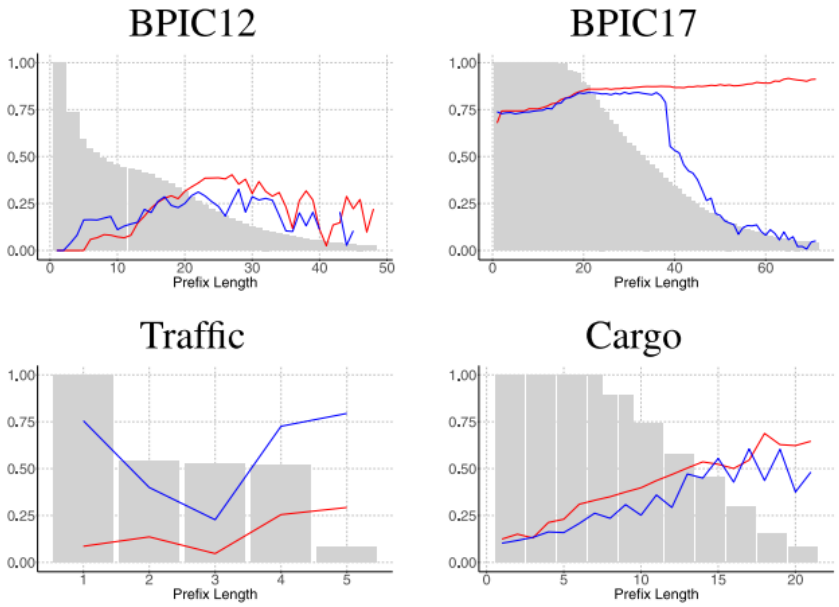
Fundamental trade-off between:

(1) Prediction accuracy

- False positive prediction → unnecessary adaptation
- False negative prediction → missed adaptation

(2) Prediction earliness

- Later predictions → less time and options for process adaptation
- Earlier predictions → lower prediction accuracy



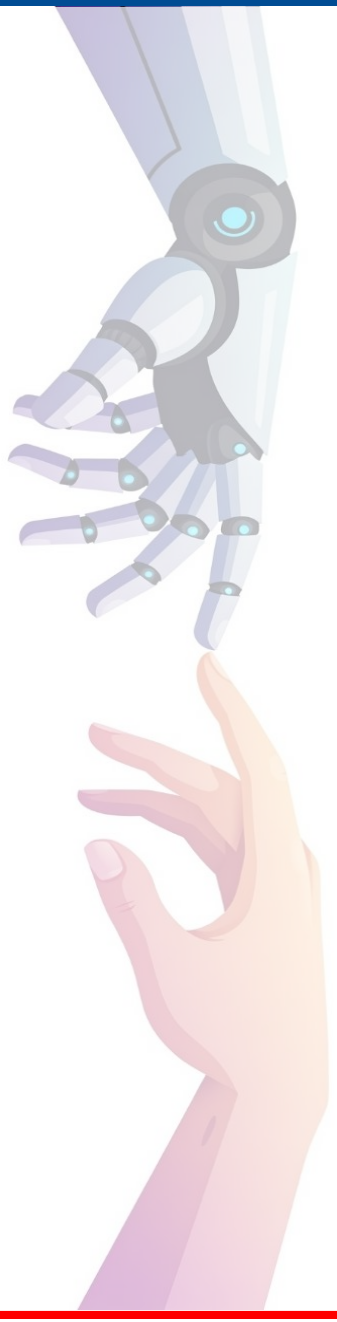
Prediction Accuracy: **LSTM**, **RF**

% of traces reaching prefix length *j*

→ How to reconcile this trade-off?

Agenda

1. Background and Motivation
2. **Solutions**
3. **Comparative Evaluation**
4. **Practical Recommendations**
5. **Outlook**



Using a Static Prediction Point

Idea

- Considers fact that later predictions typically are more accurate
- Use predictions of well-chosen, static prediction point at prefix length j_{fix}

Approach [Metzger et al. 2017 @ CAiSE]

- Calculate j_{fix} by analyzing average prediction accuracy of model for each prediction point j
- Choose earliest prediction point with highest accuracy

Shortcomings

- **No alarms raised for cases shorter than j_{fix}**
- **Average accuracy no direct indicator for accuracy of *individual* case**

[Metzger et al. 2017 @ CAiSE]: https://doi.org/10.1007/978-3-319-59536-8_28

Empirical Thresholding

Idea

- Use *reliability estimates* to account for accuracy of individual predictions
- Use earliest prediction with reliability estimate > threshold
- Reliability estimates can be computed from ensembles of prediction models (e.g., using bagging)

Approach [Fahrenkrog-Petersen et al. 2002 @ Knowl. Inf. Syst.]

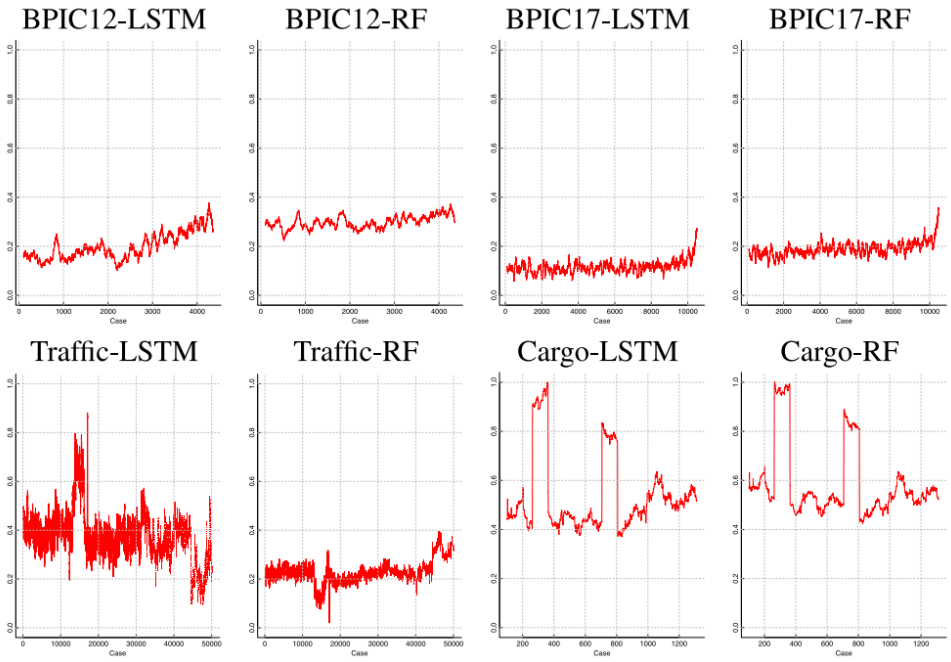
- Use dedicated training process (involving dedicated training data set) to determine suitable threshold
- Apply cost model (which defines adaptation, compensation and penalty costs)

[Fahrenkrog-Petersen et al. 2002 @ Knowl. Inf. Syst.]: https://doi.org/10.1007/s10115-021-01633-w

Empirical Thresholding

Shortcomings

- Threshold is optimal for training data, **but may not remain optimal over time, as concept drifts of process environment and data may impact prediction accuracy**



Mean absolute prediction error (MAE) per case

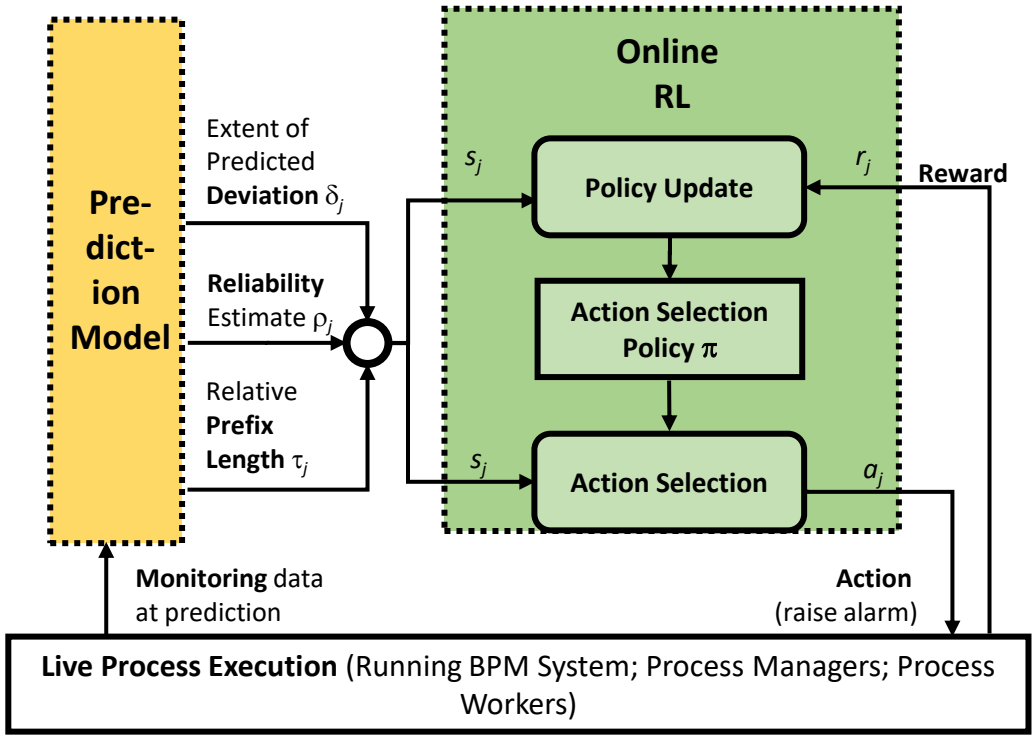
Online RL

Idea

- Learn action selection policy π at runtime
- π gives action a_j in state s_j
- Positive rewards r_j if action a_j (here: raising alarm) was a good decision

Challenges

- **Balancing exploration ↔ exploitation**
  - Learn new knowledge
  - Leverage learned knowledge
  - Typical approach: ε-decay
  - Challenged by concept drift
- **Reward engineering**
  - Defining an effective reward function r



Our Online RL Approach

Balancing exploration ↔ exploitation

- **Policy-based Deep RL (PPO)** as RL algorithm
  - Uses and optimizes parametrized stochastic action selection policy π represented as **Deep ANN**
    - Can handle multi-dimensional, continuous state spaces
    - Generalizes well over unseen neighboring states
    - Can natively handle non-stationarity and thus concept drifts of prediction model (no need to balance exploration vs. exploitation)

Reward engineering

- SOTA approaches e.g., [Branchi et al. 2022 @ BPM; Dasht Bozorgi et al. 2012 @ InfoSys] assume alternative process outcomes if not adapted is known
  → **Not realistic in practice!**
- → **Artificial curiosity** to capture above shortcoming
  - Use *intrinsic* rewards (from within the RL algorithm) in addition to *extrinsic* rewards (from environment)

[Branchi et al., 2022 @ BPM]: https://doi.org/10.1007/978-3-031-16171-1_9
[Dasht Bozorgi et al. 2023 @ InfoSys]: https://doi.org/10.1016/j.is.2023.102198

Our Online RL Approach

Strong reward signal

Facilitates faster learning convergence than using actual costs

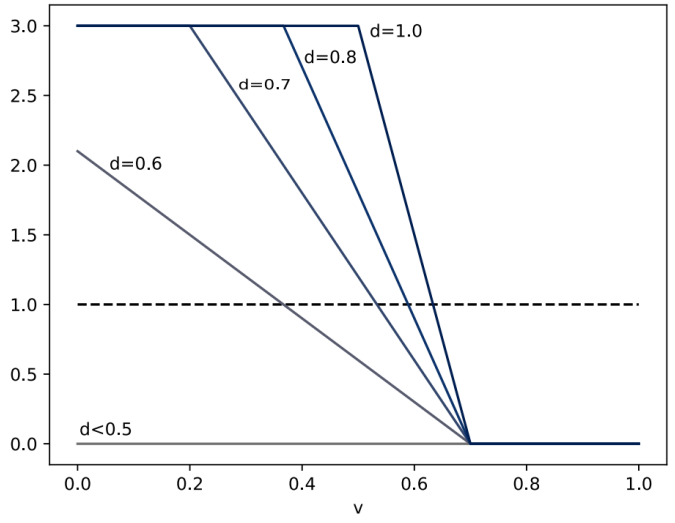
Reward Function

	Adaptation	No adaptation
Actual = Deviation	$R = b(1-c) - 2d$	$R = -1$
Actual = No deviation		$R = +1.5$

Reward signal including intrinsic rewards

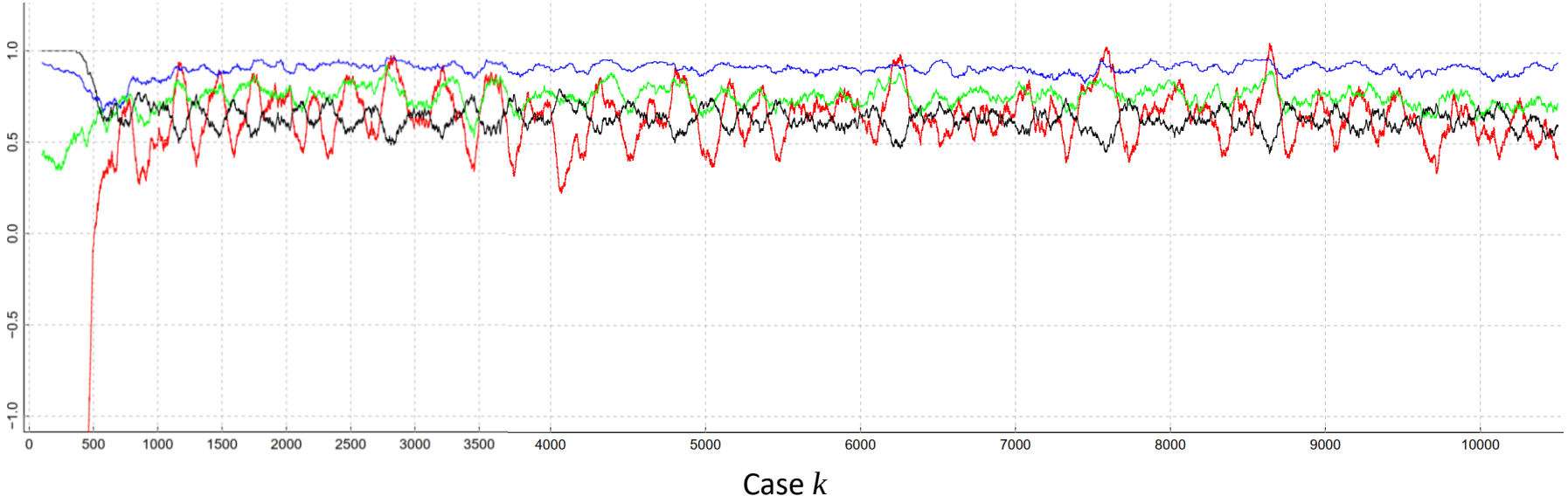
Addresses the problem of unknown alternative outcome

- d: rate of adaptations among last seen 30 cases
  → punishes high adaptation rates
  → rewards exploring not raising alarms
- b: decreases linearly with prefix-length
  → prefer early alarms over late alarms
- $c(d, v)$: curiosity modifier
  v = negative predictive value of last 100 non-adapted cases
  → high v = high accuracy in raising alarms
  → no longer need to explore raising alarms later
  → small d
  → extrinsic rewards sufficient for learning



Our Online RL Approach

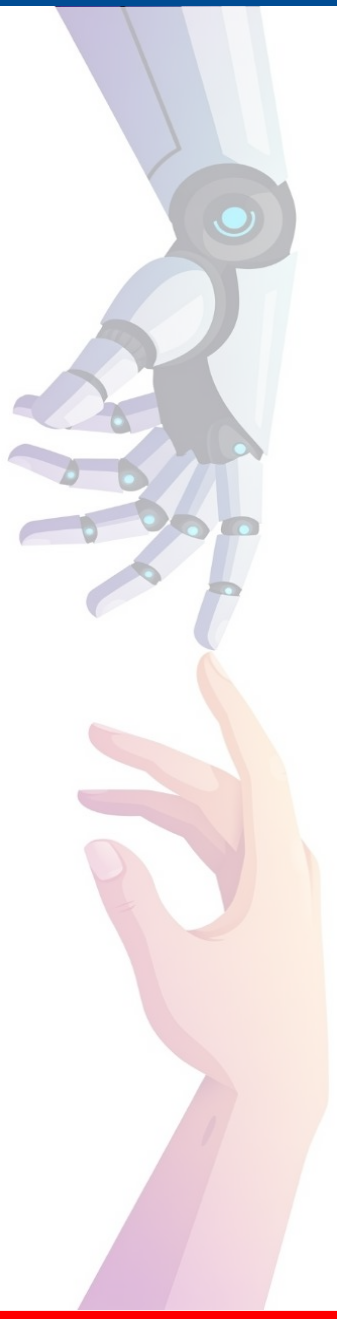
Example (BPIC 2017 – RNN)



red: normalized reward
blue: earliness (0 = end, 1 = beginning of process)
black: rate of alarms
green: rate of accurate alarms

Agenda

1. Background and Motivation
2. Solutions
3. **Comparative Evaluation**
4. **Practical Recommendations**
5. **Outlook**



Research Question

RQ: "How do the approaches compare in terms of cost savings?"

Cost Model:

Costs $C(j) =$	Prediction $\hat{y}_j =$ *deviation*		Prediction $\hat{y}_j =$ *no deviation*
	With probability α: effective adaptation	*With probability $1-\alpha$: non-effective adaptation*	
Actual $y =$ *deviation*	C_a	$C_a + C_p$	C_p
Actual $y =$ *no deviation*	$C_a + C_c$	C_a	0

- C_p: **Penalty costs**
  - Cost of undesired process outcome
  - *E.g., contractual penalties*
- C_a: **Adaptation costs**
  - Cost of intervention
  - *E.g., additional personnel costs when increasing staffing*
- α: **Adaptation effectiveness**
  - Probability that intervention was effective
  - Earlier prediction points have higher α to model fact that more time/options are available
- C_c: **Compensation costs**
  - Cost of roll-back or compensation activities
  - *E.g., compensating client for unnecessary interventions*

→ **We explore 64 different cost model configurations**
→ **We exploit 2 different prediction models: RF, RNN**
→ **We use 4 real-world data sets: BPIC12, BPIC17, Traffic, Cargo**

= 512 Experiments

Results

Average costs

		Relative number of situations when approach performs best			Average, relative cost savings when approach performs best	
Data Set	**Model**	**Static**	**Empirical**	**Online RL**	**Empirical**	**Online RL**
BPIC12	**LSTM**	7%	29%	64%	51%	26%
BPIC12	**RF**	5%	50%	45%	34%	36%
BPIC17	**LSTM**	0%	47%	53%	25%	12%
BPIC17	**RF**	13%	19%	69%	20%	20%
Traffic	**LSTM**	16%	0%	84%	/	24%
Traffic	**RF**	30%	57%	0%	16%	/
Cargo	**LSTM**	7%	60%	33%	47%	44%
Cargo	**RF**	20%	80%	0%	45%	/

→ **No single approach works best for all data sets and cost model configurations**

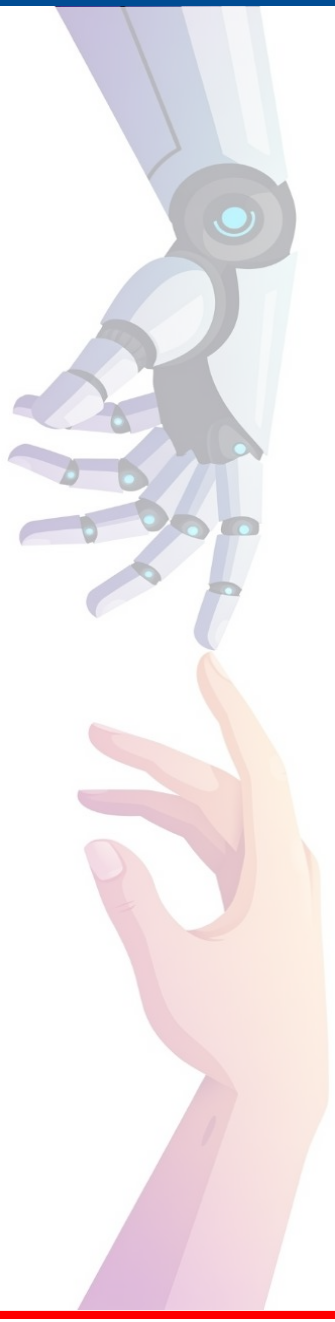
→ But: **More AI-intensive techniques outperform simpler approaches**

- Tend to work in many situations – with few exceptions
- Consistently deliver cost savings – with Empirical Thresholding delivering higher savings

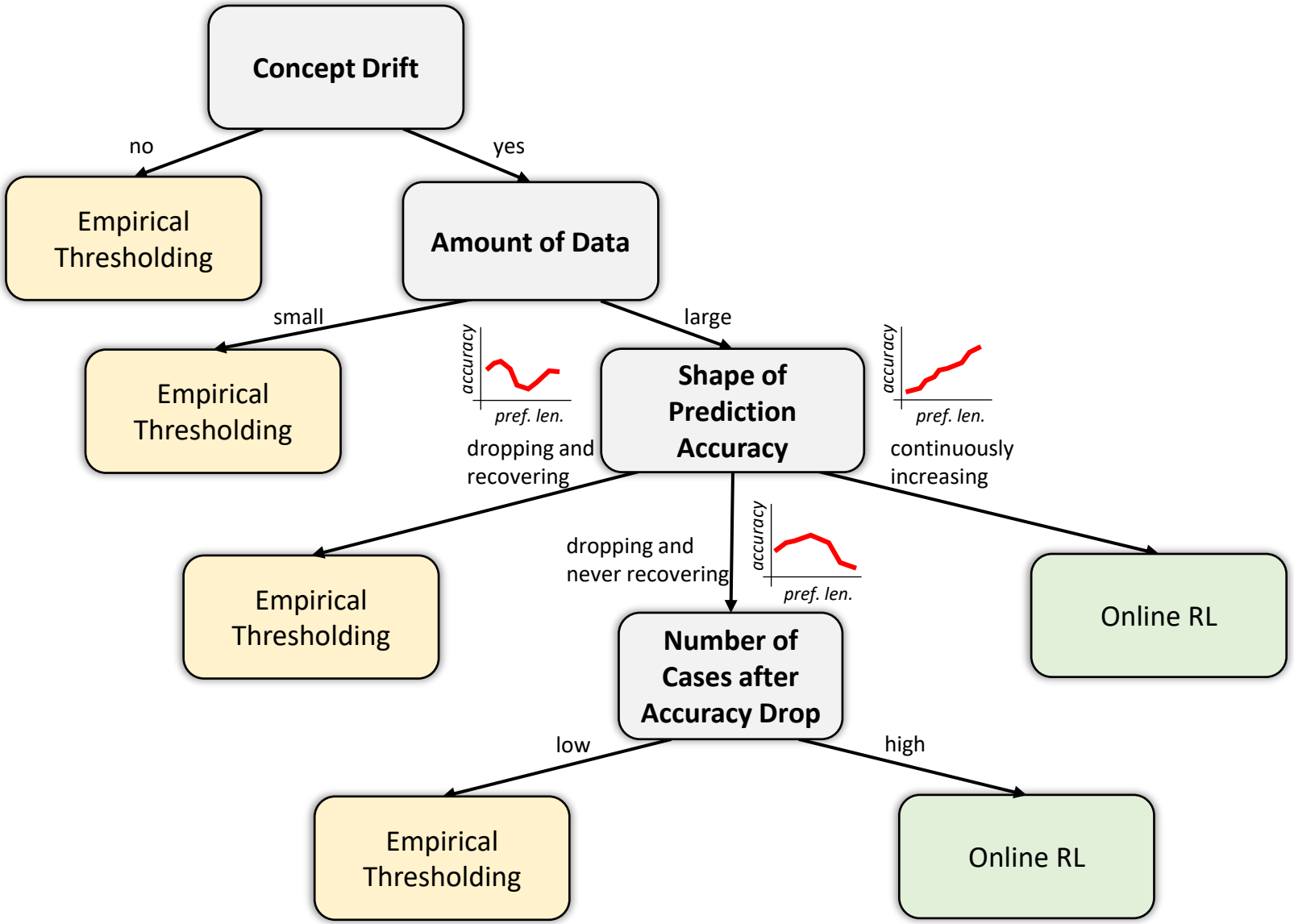
→ *Detailed analysis in our InfoSys paper*

Agenda

1. Background and Motivation
2. Solutions
3. Comparative Evaluation
4. **Practical Recommendations**
5. **Outlook**

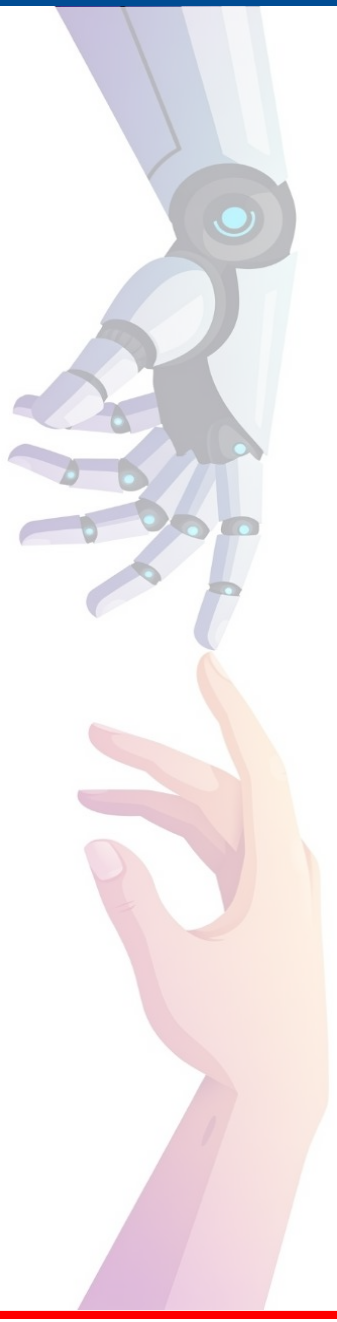


Initial Recommendations



Agenda

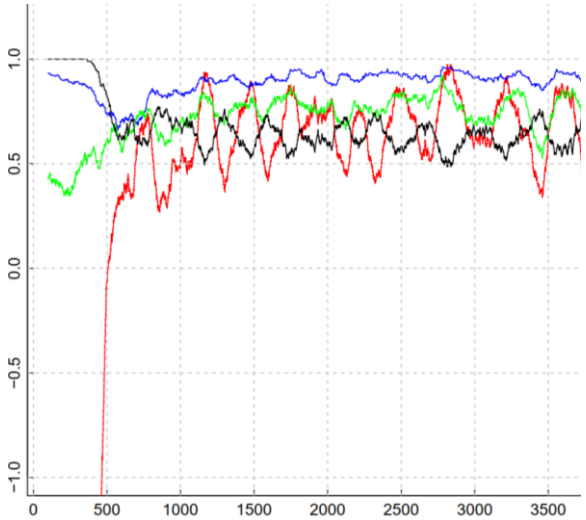
1. Background and Motivation
2. Solutions
3. Comparative Evaluation
4. Practical Recommendations
5. **Outlook**



Outlook

Speed-up of Online RL

- RL needs to learn basic trade-off between accuracy and earliness; e.g.,



Fast convergence also relevant for resource efficiency and sustainability

→Use of **Meta-RL** to reuse policies of similar learning problems

→Offline **pre-training** of RL model (e.g., using synthetic data generated from simulation models)

→**Expose RL to "important" states** determined using static analysis of simulation model [Mohsen et al. 2025 @ SEAMS]

[Mohsen et al. 2025 @ SEAMS]: https://ebjohnsen.org/publication/25-seams/25-seams.pdf

Outlook

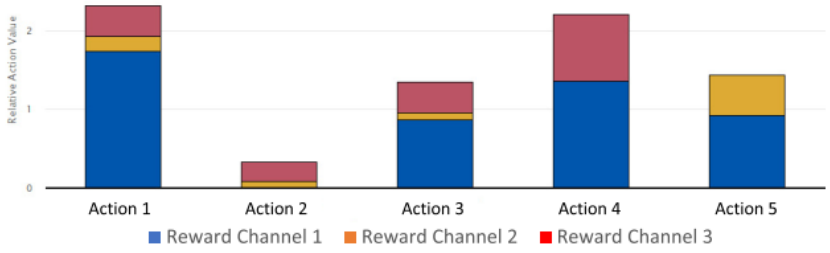
Explainability of alarms

- Reliability estimates only provide little insights why alarm was raised

→ **Use of explainable RL techniques** [Metzger et al. 2024 @ ACM TAAS]

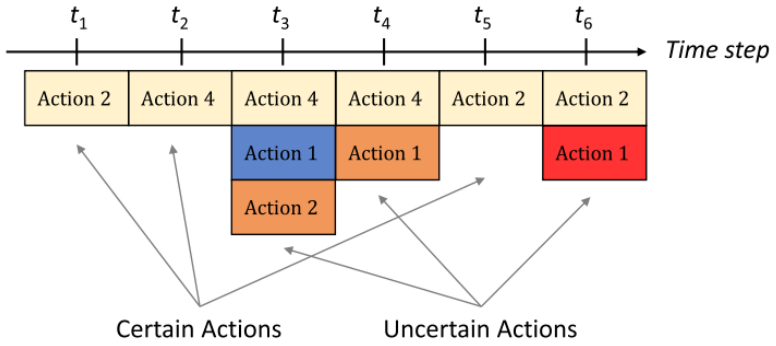
Reward Decomposition

- Reward function decomposed into sub-functions (reward channels)
- Provides contrastive explanation of short-term goal orientation of RL



Interestingness Elements

- RL considered certain in current state if "easy" to predict next action (estimated using evenness of probability distrib over all actions)
- Facilitate selecting relevant actions; e.g. certain vs uncertain



[Metzger et al. 2024 @ ACM TAAS]: https://doi.org/10.1145/3666005

Outlook

Explainability of alarms

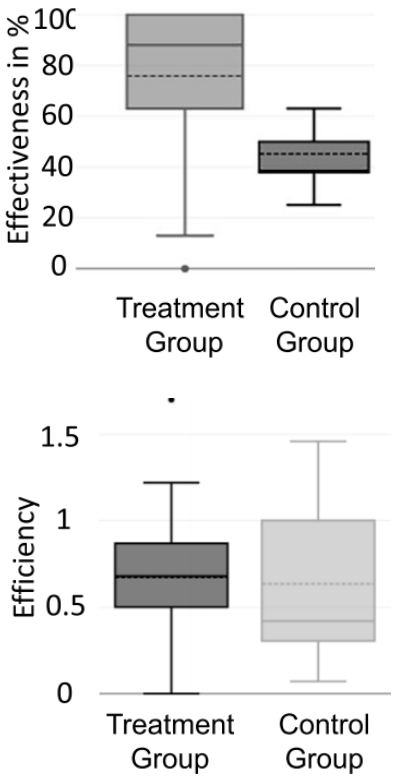
- Is it help ful? → **Empirical study** [Metzger et al. 2024 @ ACM TAAS]

RQ: Human task performance with and w/out XAI?

- Task = concrete question related to RL decision making
- 73 participants
- Results:

$$\text{effectiveness} = \frac{\text{Number of correctly performed tasks}}{\text{Number of all tasks}} = \frac{\sum c_i}{n}$$

$$\text{efficiency} = \frac{\text{Number of correctly performed tasks}}{\text{Time for performing all tasks [Minutes]}} = \frac{\sum c_i}{\sum t_i}$$



[Metzger et al. 2024 @ ACM TAAS]: https://doi.org/10.1145/3666005

Outlook

Explainability of alarms

→ **Leveraging GenAI (LLMs)** [Metzger et al. 2023 @ ICSOC]

- Provide textual explanations
- Facilitates dialogue between explainee and explainer

Question	Explanation
(EQ1) Why did Deep RL chose the action "Add Web Server" instead of "Decrease Dimer" at timestep t_{12}?	Based on the information provided, Deep RL chose the action "Add Web Server" instead of "Decrease Dimer" because the action "Add Web Server" has a higher positive impact on the goals "Revenue" and "User Satisfaction" compared to the action "Decrease Dimmer".
(EQ2) Which goal would the service-oriented system achieve when choosing action "Remove Server" instead of performing no adaptation at timestep t_{13}?	If Deep RL selects "Remove Server" instead of performing no adaptation at timestep t_{13}, the service-oriented system would achieve the goal of decreasing "Cost". The "Remove Server" action has the highest reward for the "Cost" goal compared to all other actions. However, Deep RL has chosen no adaptation because it has a higher positive impact on the "User Satisfaction" goal.

[Metzger et al. 2023 @ ICSOC]: https://doi.org/10.1007/978-3-031-48421-6_22

Outlook

GenAI als alternative to DeepRL?

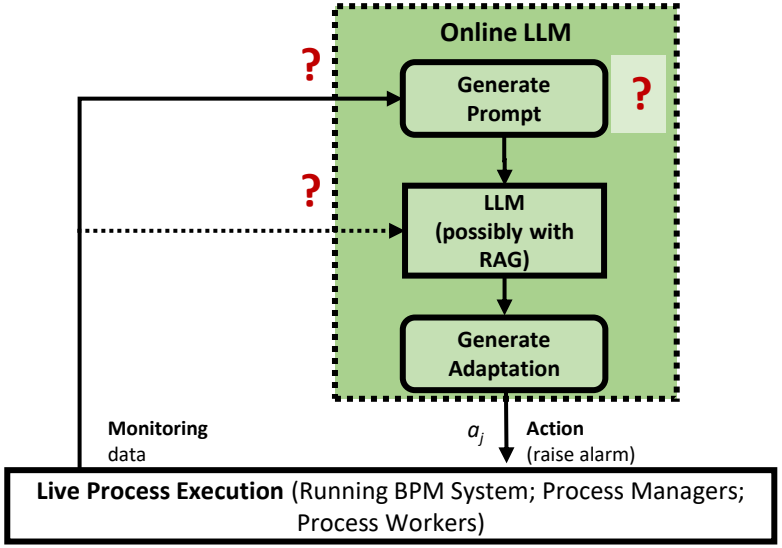
→ **LLMs to generate alarms @ runtime**

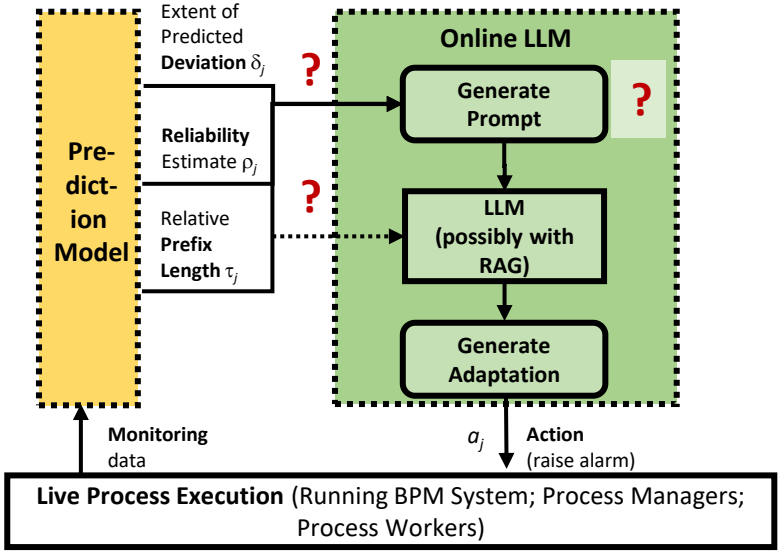
- *Idea: Transfer design-time usage of LLMs to runtime*

- What "architecture" to use?
  - E.g., with or w/out deep supervised learning for predictions?

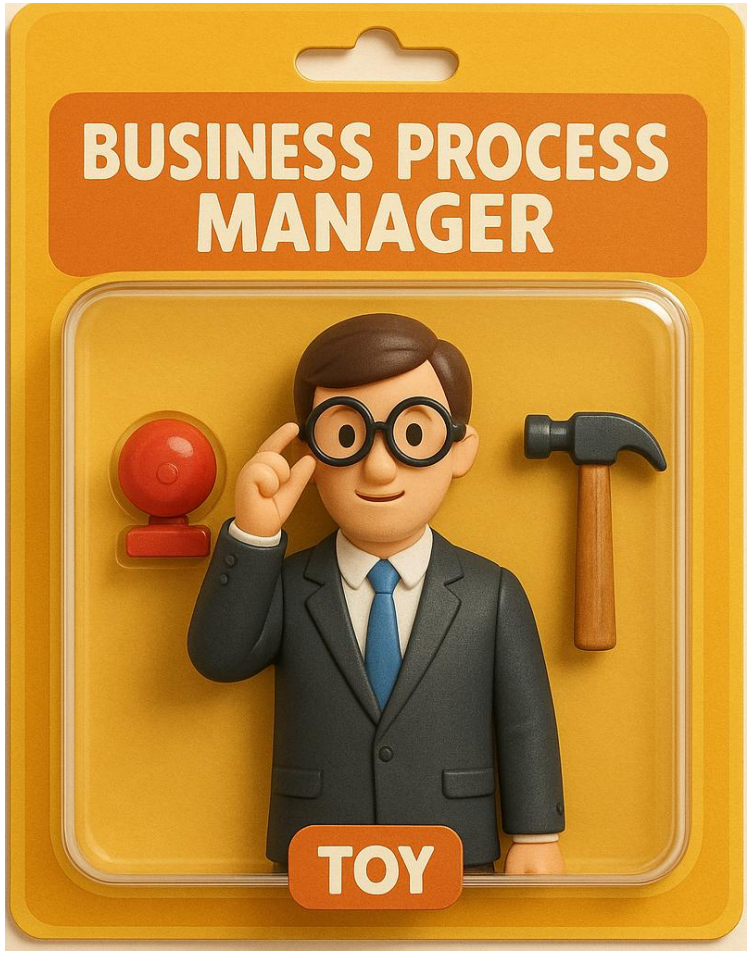
- How to engineer the inputs for the LLM?
  - ? E.g., how to convert the process monitoring data into specific prompts and/or RAG inputs

- What impact do hallucinations and bias have?





Thank You!



Research leading to these results received funding from the **EU's Horizon 2020 research and innovation programme** under grant agreements no. 731932 – **TransformingTransport,** 732630 – **BDVe,** 780351 – **ENACT,** 871493 – **DataPorts,** 101070455 – **DYNABIC**