Why are my Pizzas late?*

Dirk Fahland[1], Fabiana Fournier[2], Lior Limonad[2], Inna Skarbovsky[2] and Ava J.E. Swevels[1]

[1]*Eindhoven University of Technology, Netherlands*
[2]*IBM Research, Israel*

Abstract. We refer to explainability as a system's ability to provide sound and human-understandable insights concerning its outcomes. Explanations should accurately reflect causal relations in process executions [1]. This abstract suggests augmenting process discovery (PD) with causal process discovery (CD) to generate causal-process-execution narratives. These narratives serve as input for large language models (LLMs) to derive sound and human-interpretable explanations. A multi-layered knowledge-graph is employed to facilitate diverse process views.

Background. Process discovery (PD) summarizes an event log L into a graph model M that represents activities and control-flow dependencies [2]. Most PD algorithms construct edges in M that indicates to which subsequent activities process control "flows to". This relation is derived from traces by computing "temporally precedes" ($<$) and "directly precedes" ($\lessdot$) relations over activity names, and then discarding $a < b$ iff $a \lessdot b$ and $b \lessdot a$ [3].

Advancements in Machine Learning (ML) have made ML models more complex, sacrificing explainability and resulting in "black box" models. This led to the emergence of external explanation frameworks, known as XAI, to enhance understandability [4]. XAI frameworks are predominantly applied post-hoc, after the ML model's training [5].

Causal discovery [6] infers causal graphs from data by exploring relationships like $A \xrightarrow{c} B$ where changes in A entail changes in B. In this work, we used the Linear Non-Gaussian Acyclic Model (LiNGAM) [7] for CD as in [1]. Inspired by[8], which highlights LLMs' ability to provide interpretable explanations, we aim to demonstrate that CD can enhance explanations of process execution outcomes when used as input for LLMs. LLMs are deep-learning models trained on text data, adept at few-shot and zero-shot learning using prompt-based techniques [9].

Approach. Our research aims are combining PD, CD, and XAI to generate narratives for improved process outcome explanations using LLMs. As a proof-of-concept (POC), we show how CD helps to leverage LLMs for more sound explanations.

We use a multi-layered knowledge graph stored in a Neo4j database as infrastructure. We model the data using *labeled property graphs* in which each node and each relationship (directed edge) is typed by a *label*. Fig. 1 shows the graph schema. Each *Event* node has a *timestamp*, and is correlated to one *case*; the *directly-follows* relations describe the temporal order of all events correlated to the same case. These concepts allow modeling any event log in a graph [10].

PMAI'23: Process Management in the AI era, Aug 19, 2023, Macao, S.A.R
*This project has received funding from the European Union's Horizon research and innovation programme under grant agreements no 101092639 (FAME), and 101092021 (AutoTwin).
d.fahland@tue.nl (D. Fahland); fabiana@il.ibm.com (F. Fournier); liorli@il.ibm.com (L. Limonad); inna@il.ibm.com (I. Skarbovsky); a.j.e.swevels@tue.nl (A. J.E. Swevels)

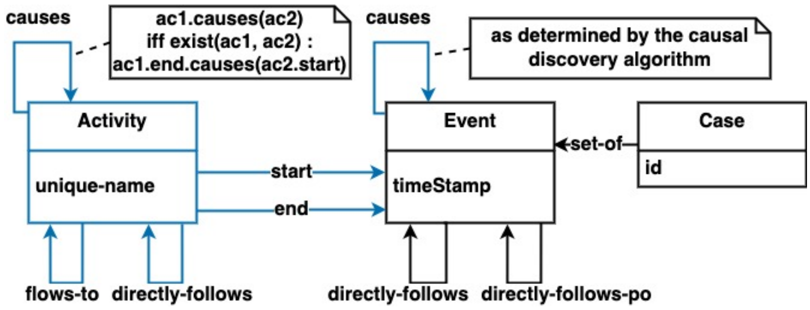


Figure 1: Graph schema

Table 1
Data distributions for different combinations of dough and toppings (sauce was added in all cases)

# of cases	Dough (duration)	Dough_type	Toppings (duration)	Toppings_type	Sauce (duration)
800	20-30	Thin	10-35	0	7-17
4800	20-30	Thin	15-40	+1	7-17
2400	20-30	Thin	20-45	+2	7-17
200	15-25	Thick	10-35	0	7-17
1200	15-25	Thick	15-40	+1	7-17
600	15-25	Thick	20-45	+2	7-17

The graph schema can be extended with additional nodes and relations for the derivation of a variety of views. For example, the schema can be extended with an *Entity* node to enable a multi-dimensional view of the process [11]. In our case, we extended the schema (blue color) with the *activity* node and the *flows-to* and *causes* relations for two activity-level inferred views:

- *Process view* - we infer *activities* and *directly-follows* ordering of activities by aggregation from events [10]. Accordingly, we infer the *flows-to* relation through process discovery, e.g., [3].
- *Causal view* - we infer the *causes* relationship as a *causal-execution-dependency* as in [1]. Accordingly, we infer the *causes* relation among the activities as annotated.

Illustrative Example. Using a simulated pizza-line dataset of $10K$ cases, our objective was to explain delivery delays. The process includes order acceptance, pizza assembly that combines sauce with a selection of dough thickness and number of toppings, baking, and boxing. Delays beyond a threshold trigger customer compensation. Start times have a 5-minute exponential distribution, and activity durations have uniform distributions as listed in Table1. The Heuristics Miner [12] PD algorithm yields the model in Fig.2.

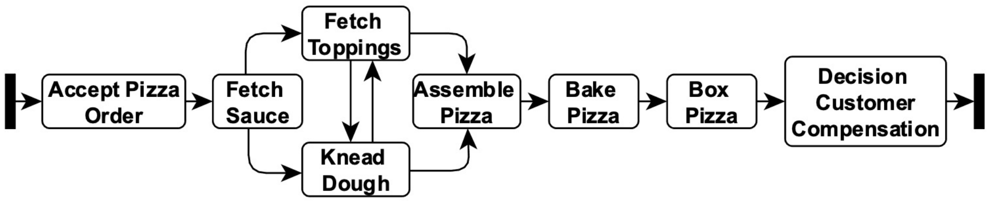
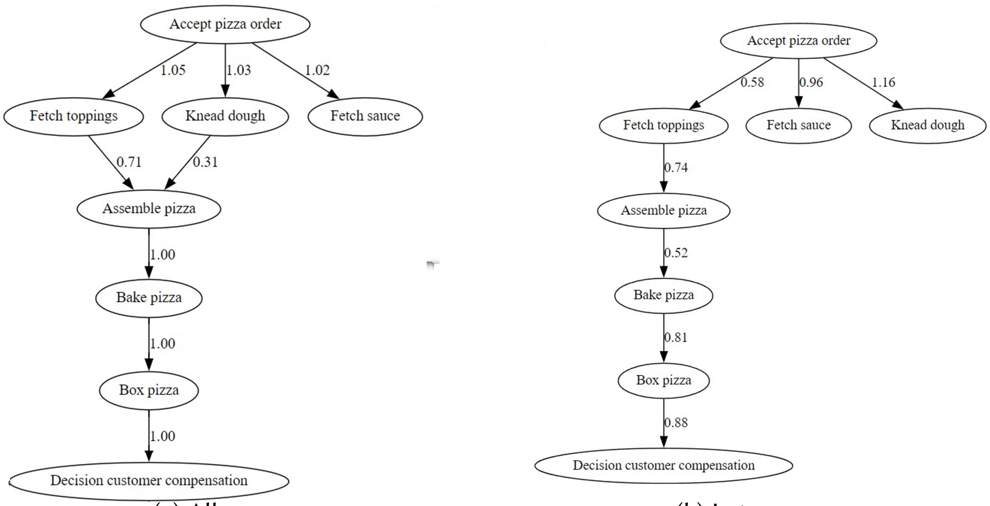


Figure 2: Heuristics Miner model

Following our approach, we applied the LiNGAM CD algorithm to generate the causal view for the process. We partitioned the event log into the entire dataset and a subset containing late cases. Fig. 3 shows the causal models for both. For each of the two views, causal and process, a narrative was generated to reflect their structure and dependencies, "causes" and "flows-to".



(a) All cases (b) Late cases

Figure 3: Causal model results

Preliminary Results. As presented in [1], the causal view supplements the process view, as a powerful tool for the purpose of process explainability. This work examines the use of LLMs to facilitate inquiries about process outcomes. For example, in Fig. 4 we show a brief interaction with ChatGPT, an LLM based chatbot (https://chat.openai.com/chat), where the user queries about the activities impacting the decision to compensate the customer for delays. In the left-hand column, the user provides the process narrative before the query, while in the right-hand column, the causal narrative precedes the query. The left-hand column reply is based on the process structure, including all activities leading up to the compensation decision. However, the right-hand column list excludes the “knead dough” and “fetch sauce” activities, as they are not part of the causal route leading to the decision, as shown in Fig. 3(b).

We also employed a conventional XAI approach using supervised ML for training a decision-tree (DT) binary classifier to distinguish between late and on-time executions using the same dataset for its training, and a second one for testing. Model input consisted of two choice variables: number of toppings and dough type. Accuracy scores (F_1) for the training and testing sets were 0.9887 and 0.9896, respectively. The feature importance values were 0.9982 for number of toppings and 0.0017 for dough type. While this result aligns with the causal view in Fig. 3(b) by emphasizing the significant influence of the number of toppings, it does not rule out the possibility that the choice of dough type has an effect on the outcome from a causal perspective. Such insights can only be obtained with the assistance of causal views.

Conclusions and Outlook. State-of-the-art explanations of process execution outcomes include the application of XAI techniques over given event logs. This work is a first step towards the use of causal discovery and LLMs’ prompt engineering for the sake of getting more sound and human-interpretable explanations. At this stage, we relied on a one-off, out-of-the-box execution of ChatGPT with no further model tuning (i.e., few-shot learning) to train the language model to interpret the input narratives, with a focus on explanations that arise from causal process execution dependencies. Next steps will extend the experiments to include LLM model tuning, real data sets, and conducting a broad analysis of the different techniques employed.

Using the heuristic process mining algorithm, the following pizza-line process was discovered: Process starts with the execution of the “accept pizza order” activity. Following the “accept pizza order” activity, process execution flows to the “fetch pizza sauce” activity. Following the “fetch pizza sauce” activity, process execution flows to “fetch toppings” and “knead dough” activities. Following the “fetch toppings” and “knead dough” activities, process execution flows to the “assemble pizza” activity. Following the “assemble pizza” activity, process execution flows to the “bake pizza” activity. Following the “bake pizza” activity, process execution flows to the “box pizza” activity. Following the “box pizza” activity, process execution flows to the “decision on customer compensation” after which process execution ends.

Q: Which activities in the process cause the “decision customer compensation” activity?

ChatGPT reply:
According to the discovered pizza-line process using the heuristic process mining algorithm, the following activities cause the "decision customer compensation" activity:
•"box pizza" activity: This activity directly leads to the "decision on customer compensation" activity.
•"bake pizza" activity: This activity leads to the "box pizza" activity, which in turn leads to the "decision on customer compensation" activity.
•"assemble pizza" activity: This activity leads to the "bake pizza" activity, which in turn leads to the "box pizza" activity and the "decision on customer compensation" activity.
•"fetch toppings" activity: This activity leads to the "assemble pizza" activity, which in turn leads to the "bake pizza" activity, the "box pizza" activity, and the "decision on customer compensation" activity.
•"knead dough" activity: This activity leads to the "assemble pizza" activity, which in turn leads to the "bake pizza" activity, the "box pizza" activity, and the "decision on customer compensation" activity.
•"fetch pizza sauce" activity: This activity leads to the "fetch toppings" and "knead dough" activities, which then lead to the "assemble pizza" activity, the "bake pizza" activity, the "box pizza" activity, and the "decision on customer compensation" activity.
•"accept pizza order" activity: This activity is the starting point of the pizza-line process and leads to all subsequent activities, including the "fetch pizza sauce", "fetch toppings", "knead dough", "assemble pizza", "bake pizza", "box pizza", and "decision on customer compensation" activities.
Therefore, any delay or failure in the execution of any of these activities could potentially delay or prevent the execution of the "decision customer compensation" activity.

Using the LiNGAM causal discovery algorithm, only the following causal relationships were detected among the activities in a pizza-line process:
“accept pizza order” activity causes the “fetch toppings” activity.
“accept pizza order” activity causes the “knead dough” activity.
“accept pizza order” activity causes the “fetch sauce” activity.
“fetch toppings” activity causes the “assemble pizza” activity.
“assemble pizza” activity causes the “bake pizza” activity.
“bake pizza” activity causes the “box pizza” activity.
“box pizza” activity causes the “decision customer compensation” activity.

Q: Which activities in the process cause the “decision customer compensation” activity?

ChatGPT reply:
According to the causal relationships identified by the LiNGAM causal discovery algorithm, the following activities in the pizza-line process cause the "decision customer compensation" activity:
•"box pizza" activity: This activity directly causes the "decision customer compensation" activity.
•"bake pizza" activity: This activity causes the "box pizza" activity, which in turn causes the "decision customer compensation" activity.
•"assemble pizza" activity: This activity causes the "bake pizza" activity, which in turn causes the "box pizza" activity and the "decision customer compensation" activity.
•"fetch toppings" activity: This activity causes the "assemble pizza" activity, which in turn causes the "bake pizza" activity, the "box pizza" activity, and the "decision customer compensation" activity.
•"accept pizza order" activity: This activity causes all subsequent activities in the pizza-line process, including the "fetch toppings", "knead dough", "fetch sauce", "assemble pizza", "bake pizza", "box pizza", and "decision customer compensation" activities.
Therefore, any delay or failure in the execution of any of these activities could potentially delay or prevent the execution of the "decision customer compensation" activity.

Figure 4: ChatGPT with respect to the two narratives: process view and causal view

References

[1] Y. David, F. Fournier, L. Limonad, I. Skarbovsky, Towards causal business process discovery: The confounder case, BPM 2023 (under review).

[2] S. J. J. Leemans, Automated process discovery, in: Encyclopedia of Big Data Technologies, Springer, 2019.

[3] A. Augusto, et al., Split miner: automated discovery of accurate and simple business process models from event logs, Knowl. Inf. Syst. 59 (2019) 251–284.

[4] C. Meske, et al., Explainable artificial intelligence: objectives, stakeholders, and future research opportunities, Information Systems Management 39 (2022) 53–63.

[5] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, D. Pedreschi, A survey of methods for explaining black box models, ACM computing surveys (CSUR) 51 (2018) 1–42.

[6] J. Peters, D. Janzing, B. Schölkopf, Elements of causal inference: foundations and learning algorithms, The MIT Press, 2017.

[7] S. Shimizu, Statistical Causal Discovery: LiNGAM Approach, Springer, 2022.

[8] M. Vidgof, S. Bachhofner, J. Mendling, Large language models for business process management: Opportunities and challenges, 2023. arXiv:2304.04309.

[9] P. Liu, et al., Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing, ACM Comput. Surv. 55 (2023).

[10] S. Esser, D. Fahland, Multi-dimensional event data in graph databases, J. Data Semant. 10 (2021) 109–141.

[11] D. Fahland, Process mining over multiple behavioral dimensions with event knowledge graphs, in: Process Mining Handbook, volume 448 of *LNBIP*, Springer, 2022, pp. 274–319.

[12] F. Mannhardt, M. de Leoni, H. A. Reijers, Heuristic mining revamped: An interactive, data-aware, and conformance-aware miner, in: BPM Demos, CEUR-WS.org, 2017.