Nirdizati Light: A Modular Framework for Explainable Predictive Process Monitoring

Andrei Buliga[1,2,*,†], Riccardo Graziosi[2,†], Chiara Di Francescomarino[3], Chiara Ghidini[1], Fabrizio Maria Maggi[1], Williams Rizzi[4] and Massimiliano Ronzani[2]

[1]*Free University of Bozen-Bolzano Bolzano, Italy*
[2]*Fondazione Bruno Kessler, Trento, Italy*
[3]*University of Trento, Trento, Italy*
[3]*Nexoya, Zurich, Switzerland*

Abstract

Nirdizati Light is an innovative Python package designed for Explainable Predictive Process Monitoring (XPPM). It addresses the need for a modular, flexible tool to compare predictive models, and generate explanations for the predictions made by the predictive models. By integrating consolidated frameworks libraries for process mining, machine learning, and explainable AI, it offers a comprehensive approach to predictive model construction and explanation generation. This paper discusses the tool's key features, and its significance in the BPM community.

Keywords

predictive process monitoring, machine learning, explainable AI

1. Introduction

Nirdizati Light is an innovative Python-based (Explainable) Predictive Process Monitoring (PPM) [1] tool offering a wide array of approaches and providing researchers and practitioners with a highly modular solution for the instantiation, comparison, analysis, selection, and explanation of predictive models for different types of prediction tasks. Existing tools like Nirdizati [2] have significantly contributed to this field by offering robust capabilities for building, analysing, and comparing predictive models, offering also a glimpse into the application of Explainable AI techniques in PPM. However, Nirdizati faces notable limitations that hinder experimental flexibility, as it is primarily tied to a user interface, restricting customization and the seamless integration of new techniques. Its fixed set of models and hardcoded workflows limits adaptability and scalability, posing challenges for researchers and practitioners who wish to innovate or tailor the tool to specific needs.

Proceedings of the Best BPM Dissertation Award, Doctoral Consortium, and Demonstrations & Resources Forum co-located with 22nd International Conference on Business Process Management (BPM 2024), Krakow, Poland, September 1st to 6th, 2024.

*Corresponding author.
†These authors contributed equally.
abuliga@fbk.eu (A. Buliga); rgraziosi@fbk.eu (R. Graziosi); c.difrancescomarino@unitn.it (C. Di Francescomarino); chiara.ghidini@unibz.it (C. Ghidini); maggi@inf.unibz.it (F. M. Maggi); mronzani@fbk.eu (M. Ronzani)

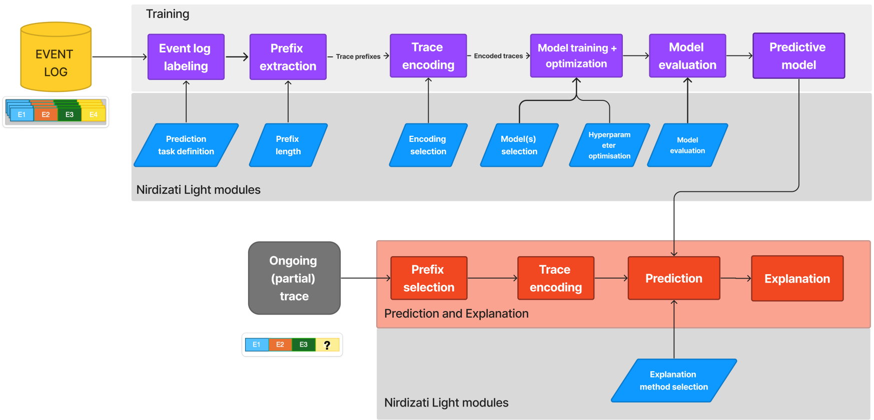

Figure 1: Diagram showing the overall pipeline for Nirdizati Light.

In response to these constraints, in this demo paper we introduce a modular and extensible Python-based version of Nirdizati, by offering more encoding techniques, newer state-of-the-art predictive models, with a particular focus on novel XAI techniques adapted to the PPM domain. With Nirdizati Light, users can explore a diverse set of trace encodings, predictive tasks, predictive models, and explanations, enhancing their ability to make data-driven decisions.

2. Nirdizati Light innovations for (X)PPM

Predictive Process Monitoring (PPM) is crucial for operational optimisation and informed decision-making. Fig. 1 shows a general pipeline employed for PPM. However, existing PPM methods often lack transparency and fail to incorporate domain-specific knowledge, limiting their effectiveness. The adoption of Deep Learning models in Predictive Process Monitoring (PPM) has synchronously brought upon the adoption of explanatory techniques intending to provide explanations for different prediction tasks. This has lead to the creation of a novel subfield, named Explainable Predictive Process Monitoring (XPPM) [3].

Nirdizati Light is a modular Python package that supports PPM by providing a comprehensive suite of functionalities for Explainable Predictive Process Monitoring (XPPM). Designed with flexibility at its core, Nirdizati Light [1] allows users to seamlessly import event logs, experiment with a range of encoding techniques, and train various predictive models. It integrates popular libraries such as *pm4py* [4] for event log handling, *scikit-learn* [2] and *PyTorch* [3] for model training, and *hyperopt* [4] for hyperparameter optimisation. This integration facilitates a cohesive environment where users can conduct all stages of event log analysis within a single platform. A standout feature of Nirdizati Light is its modularity, enabling users to effortlessly swap

[1]The tool is available at the following repository link https://github.com/rgraziosi-fbk/nirdizati-light, while the video demonstration for the tool can be found at https://tinyurl.com/bdhbwwhz
[2]https://scikit-learn.org/
[3]https://pytorch.org/
[4]https://hyperopt.github.io/hyperopt/

components like encodings, models, and explainable AI (XAI) methods. This flexibility supports a dynamic experimentation process without being confined to a rigid interface. The tool supports a diverse array of predictive tasks, including outcome prediction, next activity prediction, remaining time prediction, and trace duration prediction. This breadth of capabilities allows it to cater to a wide range of use cases and data characteristics, independently on whether the task involves classification or regression. Fig. 1 also highlights the main functionalities of Nirdizati light. We present each of the submodules of the framework below.

Event Log labeling. The **Prediction task definition** module enables the automatic labeling of logs with various predictions, including categorical outcomes, numeric values, and next activities. For categorical outcomes, it allows for multiclass labels from categorical attributes and next activities, as well as binary labels for outcome predictions. For numerical outcomes, it supports numeric labels derived from numeric attributes and trace duration.

Trace Encoder/Decoder. The **Encoding selection** module processes labelled event logs and converts them into a DataFrame suitable for machine learning. This transformation occurs through three steps: (i) **Encoding information extraction**: This step extracts critical attributes from the event log, such as control-flow (activity names), data flow (trace and event attributes), and resource-flow (resource-related attributes). This mapping identifies the relevant information for encoding; (ii) **Feature encoding**: Using the extracted information, this step determines the feature set that will represent each trace in the DataFrame; (iii) **Data encoding**: Finally, the feature set is transformed into a DataFrame. This includes operations like one-hot encoding of categorical features and normalization of numeric attributes, ensuring the data is ready for training predictive models. For this we make use of the *scikit-learn* library.

Predictive Model Selection + Optimisation. The **Model(s) selection** module allows users to specify and instantiate predictive models. It supports both classification and regression algorithms. The modular design of Nirdizati Light permits the integration and expansion of additional predictive algorithms, enhancing its adaptability to different requirements. For the predictive models, Nirdizati Light uses popular Machine Learning/Deep Learning libraries such as *scikit-learn* and *PyTorch* to instantiate the predictive models within the framework.

Hyperparameter optimisation. This module enhances model performance by automating the tuning of hyperparameters using the *hyperopt* library. This module receives the training DataFrame and an instantiated predictive model, then explores multiple hyperparameter configurations to maximize a specified quality metric. This process, although computationally intensive, significantly improves the accuracy and effectiveness of the predictive models.

Predictive Model Comparison. The **Model evaluation module** provides a comprehensive assessment of predictive models based on two primary classes of metrics: (i) **Time metrics**: Evaluate the speed at which the predictive model trains, updates, and generates predictions; (ii) **Accuracy metrics**: Assess the model's predictive performance on the test set.

This module facilitates detailed comparisons between different models, offering insights into their performance across various configurations and datasets. Nirdizati Light supports a streamlined workflow from data preprocessing to model evaluation, making it an invaluable tool for researchers and practitioners in the BPM community.

Explainability. Nirdizati Light also excels in generating actionable insights through state-of-the-art XAI methods, incorporating advanced tools such as SHAP (SHapley Additive exPlanations) [5], LiME [6], and DiCE (Diverse Counterfactual Explanations) [7] through the **Explanation method selection** module. These methods provide deep, interpretative insights into model predictions, enhancing their transparency and utility. Furthermore, the tool emphasizes knowledge-aware explainability, leveraging domain-specific knowledge to produce explanations that are not only accurate but also meaningful and easy to understand. Furthermore, we also include a selection of state-of-the-art XPPM techniques [8, 9], which leverage domain-specific knowledge, either through the form of temporal constraints (LTLf and Declare), or by providing explanations in terms of process patterns [5]. These adapted techniques focus on both providing the reasons for the prediction made by the model (so-called factual explanations) and showing the required changes to the input to achieve an alternative outcome (also known as counterfactual explanations). By integrating these advanced features and methodologies, Nirdizati Light empowers process analysts and data scientists to unlock profound insights from event logs and make well-informed decisions. Its ability to support flexible experimentation and deliver interpretive, domain-specific explanations marks a significant advancement in the XPPM domain, providing a robust and intuitive platform for comprehensive data analysis.

3. Concluding Remarks

This paper introduced Nirdizati Light, a significant advancement in the realm of Explainable Predictive Process Monitoring (XPPM), addressing the limitations of existing tools like Nirdizati by offering a modular, flexible, and powerful Python package that facilitates the construction and comparison of different predictive models and trace encodings for a given event log. Its architecture supports easy integration and comparison of various encoding techniques, predictive models, and state-of-the-art explainability methods, while its modularity allows users to experiment with and adopt the latest advancements in predictive process monitoring, tailoring solutions to specific use cases. This flexibility is crucial in the PPM domain, where researchers and practitioners need adaptable tools for a wide range of scenarios and data characteristics.

We assess the current Technology Readiness Level of Nirdizati Light to be a 4, reflecting its well-defined software structure, its versatility and robustness demonstrated through past applications in various domains [12, 13, 8, 9]. With its flexible framework and feature set, the tool offers researchers and practitioners a tool to enhance their understanding of predictive process monitoring techniques, and easily extend the framework with additional custom methods.

[5] See [10] for more details on Declare and LTLf, and [11] for more details on process patterns.

4. Acknowledgments

This work was partially supported by the Italian (MUR) under PRIN project PINPOINT Prot. 2020FNEB27, CUP H23C22000280006 and H45E21000210001, the support is greatly appreciated.

References

[1] C. Di Francescomarino, C. Ghidini, Predictive process monitoring, in: Process Mining Handbook, volume 448 of *LNBIP*, Springer, 2022, pp. 320–346.

[2] W. Rizzi, L. Simonetto, C. Di Francescomarino, C. Ghidini, T. Kasekamp, F. M. Maggi, Nirdizati 2.0: New features and redesigned backend, in: Proceedings of the Dissertation Award, Doctoral Consortium, and Demonstration Track at BPM 2019, volume 2420 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2019, pp. 154–158.

[3] M. Stierle, J. Brunk, S. Weinzierl, S. Zilker, M. Matzner, J. Becker, Bringing light into the darkness-a systematic literature review on explainable predictive business process monitoring techniques, ECIS 2021 Research-in-Progress Papers 8 (2021).

[4] A. Berti, S. van Zelst, D. Schuster, Pm4py: A process mining library for python, Software Impacts 17 (2023) 100556.

[5] S. M. Lundberg, S. Lee, A unified approach to interpreting model predictions, in: Advances in Neural Information Processing Systems 30: Annual Conf. on Neural Information Processing Systems 2017„ 2017, pp. 4765–4774.

[6] M. T. Ribeiro, S. Singh, C. Guestrin, "Why should i trust you?" Explaining the predictions of any classifier, in: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 1135–1144.

[7] R. K. Mothilal, A. Sharma, C. Tan, Explaining machine learning classifiers through diverse counterfactual explanations, in: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, 2020, pp. 607–617.

[8] A. Buliga, C. Di Francescomarino, C. Ghidini, F. M. Maggi, Counterfactuals and ways to build them: Evaluating approaches in predictive process monitoring, in: Advanced Information Systems Engineering - 35th International Conference, CAiSE 2023, Proceedings, volume 13901 of *LNCS*, Springer, 2023, pp. 558–574.

[9] A. Buliga, C. D. Francescomarino, C. Ghidini, I. Donadello, F. M. Maggi, Guiding the generation of counterfactual explanations through temporal background knowledge for predictive process monitoring, CoRR abs/2403.11642 (2024). `arXiv:2403.11642`.

[10] C. Di Ciccio, M. Montali, Declarative process specifications: Reasoning, discovery, monitoring, in: Process Mining Handbook, volume 448 of *LNBIP*, Springer, 2022, pp. 108–152.

[11] M. Vazifehdoostirani, L. Genga, X. Lu, R. Verhoeven, H. van Laarhoven, R. M. Dijkman, Interactive multi-interest process pattern discovery, in: Business Process Management - 21st Int. Conf., BPM 2023, Proceedings, volume 14159 of *LNCS*, Springer, 2023, pp. 303–319.

[12] W. Rizzi, C. D. Francescomarino, F. M. Maggi, Explainability in predictive process monitoring: When understanding helps improving, in: Business Process Management Forum - BPM Forum 2020, Seville, Spain, September 13-18, 2020, Proceedings, volume 392

of *Lecture Notes in Business Information Processing*, Springer, 2020, pp. 141–158. URL: https://doi.org/10.1007/978-3-030-58638-6_9. doi:10.1007/978-3-030-58638-6_9.
[13] M. Ronzani, R. Ferrod, C. Di Francescomarino, E. Sulis, R. Aringhieri, G. Boella, E. Brunetti, L. Di Caro, M. Dragoni, C. Ghidini, R. Marinello, Unstructured data in predictive process monitoring: Lexicographic and semantic mapping to ICD-9-CM codes for the home hospitalization service, in: AIxIA 2021, Revised Selected Papers, volume 13196 of *Lecture Notes in Computer Science*, Springer, 2021, pp. 700–715.