FairLoop: Software Support for Human-Centric Fairness in Predictive Business Process Monitoring

Felix Möhrlein[1,*], Martin Käppel[2], Julian Neuberger[3], Sven Weinzierl[2], Lars Ackermann[1], Martin Matzner[2] and Stefan Jablonski[3]

[1]*University of Applied Science Hof, Alfons-Goppel-Platz 1, 95028 Hof, Germany*

[2]*Friedrich-Alexander-University Erlangen-Nuremberg, Fürther Straße 248, 90429 Nuremberg, Germany*

[3]*University of Bayreuth, Universitätsstraße 30, 95444 Bayreuth, Germany*

Abstract

Sensitive attributes like gender or age can lead to unfair predictions in machine learning tasks such as predictive business process monitoring, particularly when used without considering context. We present FairLoop[1], a tool for human-guided bias mitigation in neural network-based prediction models. FairLoop distills decision trees from neural networks, allowing users to inspect and modify unfair decision logic, which is then used to fine-tune the original model towards fairer predictions. Compared to other approaches to fairness, FairLoop enables context-aware bias removal through human involvement, addressing the influence of sensitive attributes selectively rather than excluding them uniformly.

Keywords

Predictive Process Monitoring, Fairness, Business Process Management

1. Introduction

Predictive business process monitoring (PBPM) [1] has emerged as an important subfield within business process management research [2]. Unlike descriptive and diagnostic analysis of event data, which primarily focus on past events, predictive business process monitoring provides real-time decision support during process execution by forecasting the future evolution of process instances. This encompasses, among others, predictions of activities that will be executed next, remaining cycle times, and potential outcomes of the process [3]. From a business standpoint, these predictions offer significant value, as early insights enable proactive interventions to avoid undesirable outcomes or facilitate the thorough preparation of upcoming steps [4].

Since prediction models are increasingly applied in business processes with critical decisions, such as medical treatment processes, concerns about their fairness and the ethical implications of their use have gained increasing attention [5]. In this context, fairness refers to the equitable treatment of individuals or groups [6], particularly when sensitive attributes like gender, age, race, or socioeconomic status are involved [7]. These attributes often reflect historical inequalities or systemic biases that are embedded in the data [8].

Many PBPM techniques use machine learning algorithms to automatically construct predictive models from even log data [9]. In doing so, the prediction models may inadvertently inherit embedded biases, resulting in undesired or discriminatory predictions.

Existing approaches, therefore, often remove bias by removing sensitive attributes completely. However, this strategy is insufficient, as sensitive attributes can be crucial depending on their respective

---

[1]See https://github.com/fmoehrlein/FairLoop for code, video, and live demo.
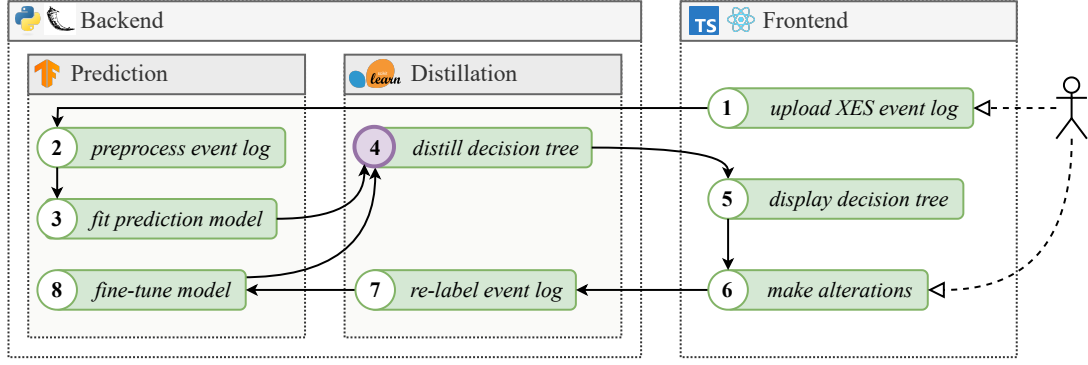
Figure 1: Overview of the modular architecture of FairLoop.

role in different activities within the process. For instance, while the gender of a patient should not influence appointment allocation, it may be essential when prescribing medication. Bias introduced by sensitive attributes that violates fairness is referred to as *negative bias* (e.g., using gender in appointment allocation). In contrast, bias that is necessary for the model's intended purpose is termed *positive bias* (e.g., considering gender for medication prescriptions). Therefore, a more nuanced approach is required, which restricts the usage of sensitive attributes only in contexts where they are harmful.

This issue is aggravated by the fact that state-of-the-art prediction models usually rely on neural networks that inherently represent a black box, i.e., their decision-making process is not transparent [10]. As a result, it is often unclear to what extent sensitive attributes are being used unfairly for predictions. In [11], a novel approach is introduced to address these two challenges. First, a decision tree (that is inherently interpretable) is distilled from the black box prediction model in order to visualize the decision-making process. Domain experts can then identify potentially discriminatory decision rules and modify the decision tree to align with the desired fair behavior. Next, the adjusted decision tree is used to re-label the training dataset to reflect unbiased decisions. Finally, the re-labeled training dataset is used to fine-tune the prediction model in order to eliminate the bias. While technically feasible, the success of this approach largely depends on the expertise of domain experts and the decisions they make.

Therefore, it is essential to support domain experts with appropriate tools that allow the identification of the bias in prediction models and offer an easy adjustment of the distilled decision trees, especially in the case of complex ones. To this end, we present FairLoop[1], a tool designed to facilitate human-guided bias removal in prediction models. FairLoop supports the described workflow of knowledge distillation, human-guided bias removal, and retraining.

2. System Overview

We designed the architecture of our tool with extensibility in mind, splitting it into two separate modules, as shown in Fig. 1.

The **frontend** allows the user to conveniently interact with the system via their browser. It is written in TypeScript and React[2], and guides the user through uploading an event log in the XES format[3], to the second component, i.e., the **backend** module, which is written in Python 3.10 and utilizes Flask for providing a REST API[4]. The backend is split into two sub-modules for **prediction** and **distillation**, which allows easy integration of new prediction or distillation approaches into our system in the

---

[1]See https://github.com/fmoehrlein/FairLoop for code, video, and live demo.
[2]See https://www.typescriptlang.org/ and https://react.dev/ respectively, both last accessed June 26, 2025.
[3]See https://xes-standard.org/, last accessed June 26, 2025.
[4]See https://www.python.org/ and https://flask.palletsprojects.com/en/stable/ respectively, both last accessed June 26, 2025
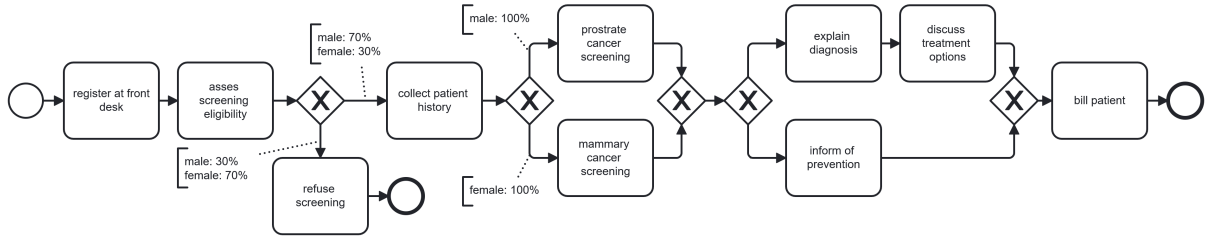
Figure 2: Diagram describing the underlying process of the simulated *cancer screening* event log used for the decision tree in Fig. 3. Decisions that are influenced by the attribute *gender* are annotated with the respective transition probabilities. While deciding whether a patient should undergo *prostate screening* or *mammary screening* based on the attribute *gender* can be a fair decision in the context of the process, refusing screening when *gender = female* is a negatively biased decision.
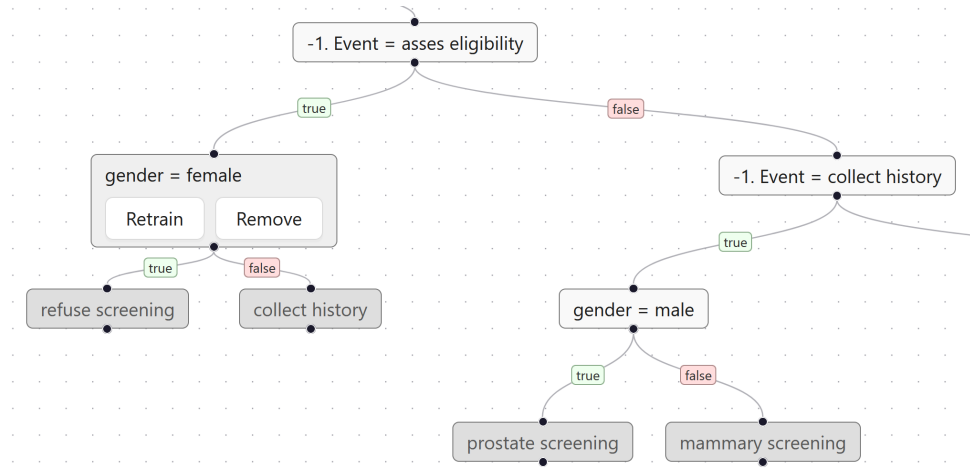

Figure 3: Screenshot of a portion of the decision tree displayed in the frontend, for a simulated event log modeled after the *cancer screening* process depicted in Fig. 2. The node responsible for unfairly refusing screening when *gender = female* is selected; it can be removed either directly or by retraining its subtree without using the *gender* attribute. Other nodes using *gender* to introduce positive bias remain unaffected.

future. The prediction sub-module currently uses TensorFlow[5] for training a multi-layer perceptron (MLP) on event log data. This MLP is then passed to the distillation sub-module, which creates a decision tree using the scikit-learn library[6]. We build the decision tree from a dataset that consists of all possible prefixes in the event log and the corresponding prediction made by the MLP. The decision tree is then converted into a custom implementation, which allows us to modify the structure of the decision tree after training. This is done by sending the decision tree to the frontend, where the user can inspect decisions and remove those that are negatively biased (see Fig. 3). After any number of alterations, the user can trigger re-labeling the training data and fine-tuning the prediction model on the modified dataset. To this end, we assign each prefix in the event log the target predicted by the (now fairer) decision tree, which in turn lets the MLP learn to make fairer predictions during fine-tuning. Performance metrics such as accuracy, F1 score, precision, and recall for the different predictive models are shown for comparison, evaluated on the original (unmodified) dataset. As a result, performance may appear to decrease, since fairer predictions can conflict with the potentially biased ground truth in the original data. After this step, the user can view the effects of their alterations by distilling a new decision tree from the fine-tuned MLP model. This distill-alter-tune cycle can be repeated any number of times, allowing the user to iteratively guide the MLP model toward increasingly fair behavior, as outlined in Fig. 1.

---

[5]See https://www.tensorflow.org/, last accessed June 26, 2025
[6]See https://scikit-learn.org/stable/index.html, last accessed June 26, 2025.

3. Key Innovations

FairLoop makes MLPs used for PBPM explainable using a novel form of visualization: distilled decision trees. By translating the behavior of MLPs into these transparent rule-based representations, FairLoop enables users to inspect and reason about predictions in a structured and intuitive way. This approach is not yet present in other graphical tools for PBPM, such as Nirdizati [12], which also aim to provide interpretability of trained models.

Another innovation of FairLoop lies in its approach to bias mitigation. Existing tools, such as the *Discrimination-aware Decision Tree* [13] plugin for ProM[7], an open source framework for process mining, represent early efforts to address fairness in PBPM. That plugin extends traditional decision trees by incorporating fairness constraints, aiming to reduce discrimination in predictions. However, this approach comes with two major limitations. First, it is restricted to decision trees as the underlying PBPM model. In contrast, FairLoop uses decision trees only as an intermediate representation to support user interaction. The actual predictions are made by more expressive MLPs, resulting in a better predictive performance. This allows FairLoop to combine the interpretability of decision trees with the predictive strength of neural networks. Second, the fairness constraints are applied uniformly, removing the influence of sensitive attributes, such as *gender*, regardless of the context. This can be problematic when the same attribute contributes to both positive and negative biases within a process, for example, as illustrated in Fig. 2. FairLoop addresses this limitation by introducing human-in-the-loop intervention, allowing users to apply context-aware adjustments to the model's decision logic. While other approaches toward fairness in PBPM exist, such as [14] and [15], they share similar limitations: they do not involve the user in the decision-making process and remove bias uniformly instead. Moreover, these other methods do not offer a tool implementation with a graphical interface.

4. Maturity

While the technical feasibility of the approach underlying FairLoop has been demonstrated in our previous work [11], this paper focuses on a hands-on demonstration using the simulated *cancer screening* event log, as portrayed in Fig. 2. Using a simulated event log is necessary, since no real life event log containing sensitive data has been made publicly available. For the demonstration, we provide an intuitive graphical interface that enables users to interactively inspect and adjust decision trees distilled from predictive models. This allows practitioners without programming experience or familiarity with command-line tools to effectively use FairLoop. However, it is important to note that no user study has yet been conducted to evaluate FairLoop in practice. As such, its usability, effectiveness in supporting fairness interventions, and suitability for real-world deployment remain to be formally assessed.

Limitations. While FairLoop demonstrates promise, it currently has some restrictions. At present, the tool supports only the PBPM task of the next activity prediction, and only MLPs can be used as the underlying predictive model. However, FairLoop's architecture is designed with extendability in mind. We plan to broaden the supported prediction task types and model architectures in order to make FairLoop applicable to a wider range of use cases. In addition, options for encoding techniques and hyperparameter configuration are currently limited to the most essential settings. To better support model optimization, including hyperparameter tuning, future work should expand these configuration capabilities. Another key limitation is the absence of built-in metrics for quantifying the fairness of predictive models, meaning the effectiveness of user-driven interventions is currently assessed solely through manual inspection of the distilled decision tree.

Future Work. Besides addressing the aforementioned limitations, we plan to conduct a comprehensive user study to evaluate the usability and effectiveness of FairLoop. As part of this study, we aim to investigate how fairness metrics can be integrated into the interface to provide intuitive feedback on the consequences of user interventions. We also intend to explore techniques for guiding users through the inspection and modification of decision trees. These techniques are increasingly important, as

---

[7]See https://promtools.org/, last accessed June 26, 2025.

the complexity of the distilled trees grows with more expressive prediction models and more complex datasets.

Declaration on Generative AI

During the preparation of this work the authors used no generative AI.

References

[1] F. M. Maggi, C. Di Francescomarino, M. Dumas, C. Ghidini, Predictive monitoring of business processes, in: Proceedings of the 26th International Conference on Advanced Information Systems Engineering, Springer, 2014.

[2] L. Ackermann, M. Käppel, L. Marcus, L. Moder, S. Dunzer, M. Hornsteiner, A. Liessmann, Y. Zisgen, P. Empl, L.-V. Herm, et al., Recent advances in data-driven business process management, arXiv preprint arXiv:2406.01786 (2024).

[3] C. Di Francescomarino, C. Ghidini, Predictive process monitoring, in: Process Mining Handbook, Springer, 2022, pp. 320–346.

[4] A. E. Márquez-Chamorro, M. Resinas, A. Ruiz-Cortés, Predictive monitoring of business processes: a survey, IEEE Transactions on Services Computing 11 (2017).

[5] M. de Leoni, A. Padella, Achieving fairness in predictive process analytics via adversarial learning, in: CoopIS, Springer, 2024.

[6] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, R. Zemel, Fairness through awareness, 2011. URL: https://arxiv.org/abs/1104.3913. arXiv:1104.3913.

[7] L. Oneto, S. Chiappa, Fairness in machine learning, in: Recent Trends in Learning From Data: Tutorials from the INNS Big Data and Deep Learning Conference, Springer, 2020.

[8] S. Barocas, A. D. Selbst, Big data's disparate impact essay, California Law Review 104 (2016) 671–732.

[9] S. Weinzierl, S. Zilker, S. Dunzer, M. Matzner, Machine learning in business process management: A systematic literature review, Expert Systems with Applications (2024) 124181.

[10] M. Käppel, L. Ackermann, S. Jablonski, S. Härtl, Attention please: What transformer models really learn for process prediction, in: Proceedings of the 22nd International Conference on Business Process Management, Springer Nature Switzerland, 2024, pp. 203–220.

[11] M. Käppel, J. Neuberger, F. Möhrlein, S. Weinzierl, M. Matzner, S. Jablonski, A human-in-the-loop approach for improving fairness in predictive business process monitoring, in: Proceedings of the 23rd International Conference on Business Process Management, Springer, 2025. In press.

[12] K. Jorbina, A. Rozumnyi, I. Verenich, C. Di Francescomarino, M. Dumas, C. Ghidini, F. M. Maggi, M. La Rosa, S. Raboczi, Nirdizati: A web-based tool for predictive process monitoring, in: Proceedings of the BPM Demo Track and BPM Dissertation Award, co-located with 15th International Conference on Business Process Management, CEUR-WS, 2017.

[13] M. S. Qafari, W. Van der Aalst, Fairness-aware process mining, in: On the Move to Meaningful Internet Systems: OTM 2019 Conferences: Confederated International Conferences: CoopIS, ODBASE, C&TC 2019, Springer, 2019, pp. 182–192.

[14] J. Peeperkorn, S. De Vos, Achieving group fairness through independence in predictive process monitoring, arXiv preprint arXiv:2412.04914 (2024).

[15] M. de Leoni, A. Padella, Achieving fairness in predictive process analytics via adversarial learning (extended version), arXiv preprint arXiv:2410.02618 (2024).