Software Fairness Analysis and Repair via Causal Model-Guided Data Mutation

YING XIAO, King's College London, United Kingdom
ZHENPENG CHEN*, Tsinghua University, China
YEPANG LIU†, Southern University of Science and Technology, China
MOHAMMAD REZA MOUSAVI, King's College London, United Kingdom
JIE M. ZHANG, King's College London, United Kingdom

Machine learning (ML) software automates various decision-making processes, significantly enhancing the efficiency of societal operations. However, the widespread adoption of ML software raises growing concerns about fairness, as such software often exhibits biases that disadvantage specific demographic groups or individuals. Since these biases typically stem from correlations between sensitive attributes and label within the training data, disrupting these correlations is a promising direction for fairness repair. Therefore, we introduce **Fairabel**, a novel approach that repairs fairness by mutating the training data under the guidance of causal models and multi-objective optimization. To evaluate Fairabel, we conduct an extensive empirical comparison against existing approaches across 40 decision-making scenarios. Our experimental results show that Fairabel reduces ML software bias by **50%** on average across three fairness metrics, outperforming the state-of-the-art with a **9%** relative improvement. Additionally, Fairabel surpasses the state-of-the-art by **1.5%** on average across five performance metrics, demonstrating a superior ability to preserve software performance.

CCS Concepts: • **Software and its engineering** → **Software creation and management**; • **Computing methodologies** → *Machine learning*.

Additional Key Words and Phrases: Software Fairness, Bias Mitigation, Data Mutation, Causal Model

ACM Reference Format:
Ying Xiao, Zhenpeng Chen, Yepang Liu, Mohammad Reza Mousavi, and Jie M. Zhang. 2026. Software Fairness Analysis and Repair via Causal Model-Guided Data Mutation. *ACM Trans. Softw. Eng. Methodol.* 1, 1 (July 2026), 26 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

ML software is driving significant positive transformations across various sectors of society, including healthcare, finance, and justice [21]. Despite these contributions, ML software often raises concerns about fairness bugs, which refer to any flaw within a software system that creates a discrepancy between the existing and intended fairness conditions [23, 24]. For example, in 2018, Amazon

*Corresponding author.
†Yepang Liu is affiliated with the Research Institute of Trustworthy Autonomous Systems and the Department of Computer Science and Engineering of Southern University of Science and Technology (SUSTech).

Authors' Contact Information: Ying Xiao, King's College London, London, United Kingdom, ying.1.xiao@kcl.ac.uk; Zhenpeng Chen, Tsinghua University, Beijing, China, zpchen@tsinghua.edu.cn; Yepang Liu, Southern University of Science and Technology, Shenzhen, China, liuyp1@sustech.edu.cn; Mohammad Reza Mousavi, King's College London, London, United Kingdom, mohammad.mousavi@kcl.ac.uk; Jie M. Zhang, King's College London, London, United Kingdom, jie.zhang@kcl.ac.uk.

ACM 1557-7392/2026/7-ART
https://doi.org/10.1145/nnnnnnn.nnnnnnn

discontinued an automated hiring tool after it was found to exhibit gender bias that disadvantaged female job applicants [29]. Since fairness is a fundamental software requirement [39, 43, 73], software fairness repair (also known as fairness bug fixing or bias mitigation) has become an essential ethical responsibility for software researchers and engineers [17].

Bias in training data is recognized as a key root cause of fairness bugs in ML software [59]. Since such software follows a data-driven paradigm, the training data significantly influence the decision logic of models [17, 44]. The causal relationships or correlations between sensitive attributes and data label can inadvertently cause ML software to make decisions influenced by those sensitive attributes, even when they are irrelevant to the specific tasks [89].

As a result, modifying training data through pre-processing techniques before model training has become an effective strategy for repairing fairness bugs [46, 56]. Common methods include modifying data features [21], adjusting data label [16], data augmentation [17], and data instance removal [18]. However, there is still a lack of an effective framework to guide the data modification process, increasing the risk of altering critical information in the data and potentially degrading ML performance [16]. Therefore, a framework that can guide data modification to repair fairness while maintaining model performance is critically needed [21, 45, 63].

To address this gap, we propose Fairabel, a novel approach that integrates a causal model with multi-objective optimization to guide training data mutation. Specifically, Fairabel uses the learned causal model as a structure-informed scoring mechanism to rank candidate mutations. By capturing directed dependencies among sensitive attributes, non-sensitive features, and labels, the model provides practical guidance for fairness-oriented data repair. Furthermore, the multi-objective optimization framework provides active feedback to dynamically balance potentially conflicting model properties, thereby improving model fairness while preserving overall performance.

We extensively evaluate Fairabel by comparing it with state-of-the-art approaches across 40 decision-making scenarios. The experimental results demonstrate that Fairabel significantly enhances fairness while preserving overall model performance. Specifically, Fairabel reduces single-attribute bias by 50% on average across three single-attribute fairness metrics, outperforming the state-of-the-art with a 9% relative improvement. Regarding multi-attribute scenarios, Fairabel reduces intersectional bias by 38% on average across three intersectional fairness metrics, outperforming the state-of-the-art with a 22% relative improvement. Additionally, Fairabel surpasses the state-of-the-art by 1.5% on average across five performance metrics, demonstrating superior performance preservation effectiveness.

In general, this work makes the following contributions:

- We propose **Fairabel**, an innovative ML software fairness repair approach that employs a causal model and multi-objective optimization to guide the training data mutation process.
- We conduct a comprehensive empirical study comparing Fairabel with the latest fairness repair methods across 40 decision-making scenarios. Experimental results show that Fairabel statistically outperforms existing methods in enhancing fairness and preserving performance.
- We publicly release the scripts, datasets, and results of this work to contribute to the community by facilitating future research and supporting extensions of the Fairabel approach [88].

The remainder of this paper is organized as follows. Section 2 outlines the necessary background. Section 3 details the proposed Fairabel methodology. In Sections 4 and 5, we report the experimental evaluation and results, including comparisons with leading fairness repair methods. Section 6 analyzes the Fairabel approach, including its advantages, limitations, and implications. Section 7 reviews related work on software fairness repair, and Section 8 concludes the paper by summarizing our contributions and key findings.

2 Preliminaries

In this section, we introduce the necessary background on software fairness and causal models.

2.1 Notions of Software Fairness

In a classification task, the ML software aims to correctly predict the label class based on input features. Given a decision-making scenario with label Y and sensitive attribute A, if a label class indicates potential advantages or benefits, it is referred to as a "**favorable label**" (e.g., good credit), denoted as $Y = 1$. Conversely, a label class that implies potential disadvantages is termed an "**unfavorable label**" (e.g., bad credit), denoted as $Y = 0$ [17, 18].

When assessing fairness in ML decision-making tasks, practitioners typically categorize individuals into various groups based on different sensitive attributes, such as sex, race, and age. If an ML model tends to predict members of a certain group with the favorable label, this group is designated as a **privileged group**, denoted as $A = 1$. Conversely, a group more often predicted with an unfavorable label class by the ML model is labeled as an **unprivileged group**, denoted as $A = 0$ [17, 21, 23]. We next introduce different types of fairness, including single-attribute bias and intersectional bias.

2.1.1 Single-Attribute Bias. Single-attribute bias refers to the statistical disparity in model predictions across groups defined by a single sensitive attribute, such as sex. To quantify this type of fairness, widely adopted metrics include Statistical Parity Difference (SPD), Average Odds Difference (AOD), and Equal Opportunity Difference (EOD) [23, 52, 59]. Their formal definitions are detailed in Section 4.5.

2.1.2 Intersectional Bias. Intersectional bias characterizes the statistical disparity in model predictions across subgroups defined by combinations of multiple sensitive attributes (e.g., the intersection of sex and race) [22]. Specifically, it quantifies the worst-case disparity, typically measured as the gap between the most privileged and the most disadvantaged subgroups [22].

For instance, in the Adult dataset [5], sex and race serve as sensitive attributes, distinguishing four intersectional subgroups: "White Male", "White Female", "Non-White Male", and "Non-White Female". In this case, the "White" and "Male" groups are the privileged groups (typically correlated with the high-income label), while the "Non-White Female" subgroup suffers from compounded discrimination arising from both gender and racial dimensions. To quantify such compounded disparities, we employ intersectional versions of single-attribute fairness metrics (e.g., SPD, AOD, EOD) [22], whose mathematical formulations are detailed in Section 4.5.

2.2 Causal Model

A causal model provides a rigorous framework to represent and reason about the causal mechanisms governing the interactions among variables (e.g., sensitive attributes and predicted outcomes). According to Pearl [65, 66], the formal definition of a causal model is:

Definition 2.1 (Causal Model [65, 66]). A causal model is a pair $\mathcal{M} = \langle D, \Theta_D \rangle$ consisting of a causal structure D over a set of variables V, and a set of parameters Θ_D compatible with D. The parameters Θ_D assign a function $x_i = f_i(pa_i, u_i)$ to each $X_i \in V$ and a probability measure $P(u_i)$ to each u_i, where pa_i represents the values of the parent set PA_i of X_i in D, and each u_i is a realization of a random disturbance U_i that is distributed according to $P(u_i)$, independently of all other U_j.

This framework offers an interpretable foundation for multivariate analysis [65, 66]. Specifically, based on the definition above, a causal model comprises two main components: (1) the **causal structure**, which encodes the directional dependencies among variables using a directed acyclic

graph (DAG); and (2) the **causal parameters**, which describe the strength and functional form of these causal relationships. The following subsections introduce these two components in detail.

2.2.1 Causal Structure. The formal definition of causal structure is presented below:

Definition 2.2 (Causal Structure [65, 66]). A causal structure over a set of variables V is a directed acyclic graph (DAG) in which each node corresponds to a distinct element of V, and each edge represents a direct functional relationship between the corresponding variables.

The causal structure qualitatively captures the directional dependencies among variables and provides a graphical representation of the underlying data-generating process. Each directed edge represents a potential cause-and-effect relationship, showing how information or influence flows through the system. In fairness analysis, such structures help identify direct and indirect pathways from sensitive attributes, such as gender or race, to model predictions, thereby supporting the development of effective fairness repair strategies.

2.2.2 Causal Parameters. While the causal structure describes qualitative dependencies, the **causal parameters** quantify the strength and functional form of these relationships. Together, they constitute a complete Structural Causal Model (SCM), detailing how each variable is generated from its parents and exogenous factors [41, 65, 66, 76].

Beyond standard predictive capabilities, a key advantage of this framework is the ability to incorporate **domain knowledge**. For instance, during the model construction, sensitive attributes (e.g., race, sex) can be explicitly constrained as root nodes (having only outgoing edges), while the target label is typically modeled as a sink node (having only incoming edges). Imposing such structural constraints narrows the hypothesis space and encourages the learned structure to align better with plausible domain knowledge, thereby improving its interpretability as a mutation-guidance model.

3 Methodology

In this section, we introduce the methodology of Fairabel and describe its implementation in detail.

3.1 Overview of Fairabel

Fairabel is a pre-processing fairness repair approach that improves model fairness by debugging the training data before model training. In contrast to existing data debugging methods [17, 18, 48], Fairabel employs a causal model to guide data mutation operations and applies multi-objective optimization to balance performance and fairness metrics. Specifically, Fairabel mutates a subset of data instances without removing or augmenting any samples. Figure 1 illustrates the key components of the Fairabel framework, comprising causal model learning and causal model-guided mutation of training instances. The following sections describe each step in detail.

3.2 Causal Model Learning

As introduced in Section 2.2, a causal model consists of a causal structure and a set of causal parameters [65, 66]. We first estimate the causal structure from the original training data. Our approach assumes causal sufficiency, meaning that all common causes of the observed variables are included in the dataset. This assumption is standard in causality-based fairness research [49, 52]. However, when relying solely on observational data, structure learning algorithms typically identify only a Markov equivalence class, represented by a CPDAG, rather than a uniquely oriented DAG [80, 90]. Consequently, identifying a more realistic causal structure requires additional information to resolve this indeterminacy [70]. To address this challenge and ensure the validity of the learned structure, we incorporate domain knowledge as constraints during the learning process. Specifically,

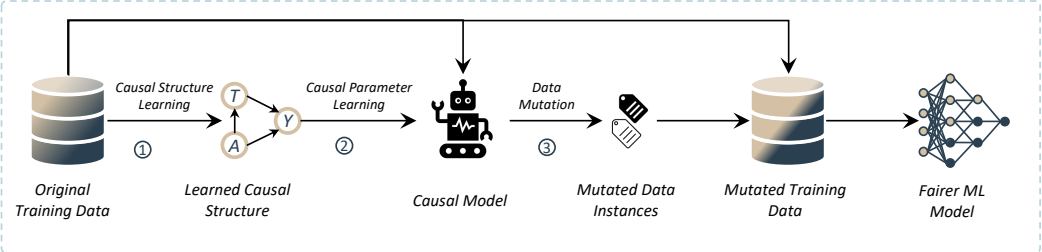


Fig. 1. Overview of Fairabel.

we designate sensitive attributes, such as race, sex, and age, as exogenous root nodes with only outgoing edges, and enforce these priors using whitelist and blacklist constraints.

Assumption (Exogenous Sensitive Attributes). Let A denote the set of sensitive attributes and Y denote the outcome variable. We assume:

(1) Each sensitive attribute $A_j \in A$ is a root node in the causal DAG, that is, $\mathrm{PA}(A_j) = \emptyset$. Any edge connecting A_j to another variable must be directed outward from A_j.
(2) The outcome variable Y is a sink node, meaning that Y has no outgoing edges.

This strategy effectively narrows the hypothesis space and prevents implausible causal directions, such as education influencing race, thereby producing structures that better reflect real-world mechanisms. The incorporation of background knowledge is a standard practice in causal discovery [77]. Although these constraints do not guarantee the unique recovery of the true causal structure, they effectively eliminate clearly spurious relationships and provide a robust structural foundation for subsequent analysis. For edges that remain undirected in the CPDAG, our method relies exclusively on the directed causal relationships involving sensitive attributes and the label. Undirected edges among non-sensitive features do not affect the mutation candidate ranking. Finally, based on the established structure, we estimate the corresponding causal parameters to complete the model for identifying the mutation candidates. Notably, the causal model is not introduced to maximize predictive accuracy. In Fairabel, the learned causal model is used operationally to score and rank candidate mutations according to the dependencies encoded in the learned structure and parameters. We do not use it to estimate a specific causal effect or to simulate a real-world intervention. Thus, its role is to provide structural guidance for fairness-oriented data repair.

3.3 Data Instance Mutation

The primary goal of data mutation is not to recover a true post-intervention causal distribution for a specific edge. Instead, Fairabel is intended as a performance-preserving fairness-oriented training-data repair method: it modifies the observed training data so as to reduce the unfair dependence between sensitive attributes and the learning target, while preserving predictive performance as much as possible and avoiding excessive deviation from the original data. In this sense, feature mutation and label mutation should be understood as controlled repair operations on the observed training distribution, rather than as estimators of a particular causal intervention. Specifically, we achieve this objective by increasing the proportion of unprivileged instances assigned the favorable label.

To minimize alterations to the overall training dataset, we restrict mutation operations to data instances that belong to the unprivileged group and are assigned the unfavorable label. Given the objective of altering the relationship between sensitive attributes and data label, we propose two strategies to achieve this: (1) **label mutation** and (2) **sensitive attribute mutation**. In addition,

we introduce a multi-objective framework to determine the optimal number of data instances to mutate to achieve fairness repair and balance model fairness and performance. We next describe the detailed implementation of Fairabel.

3.3.1 Data Label Mutation. Let $\mathcal{M}_1$ denote the learned causal model used to score candidates for label mutation. We define the candidate set for the single-attribute case as

$$C \;=\; \{\, i \mid A_i = 0,\; Y_i = 0 \,\},$$

where i indexes data instances, A_i is the sensitive attribute (0 = unprivileged, 1 = privileged), and $Y_i \in \{0, 1\}$ is the original label (1 = favorable). Then, we let T_i denote the feature vector excluding A. For each $i \in C$, we obtain the probability of a favorable label by the learned causal model $\mathcal{M}_1$:

$$p_i \;=\; \mathcal{M}_1(A_i, T_i) \in [0, 1].$$

Let $P = \{(i, p_i) \mid i \in C\}$ be the set of index-probability pairs. We sort candidates in *descending* order of p_i and introduce a mutation budget $\alpha \in [0, 1]$ (percentage of C). Let $k = \lfloor \alpha \cdot |C| \rfloor$. We then set the label of top-k candidates to 1:

$$\widehat{Y}_i \;=\; \begin{cases} 1, & \text{if } i \in \text{Top-}k(P), \\ Y_i, & \text{otherwise.} \end{cases}$$

Applying these mutation rules to the mutation candidates generates a mutated label vector. We then replace the original data label with the new label vector, producing a correspondingly mutated training dataset.

Additionally, the choice of α influences the trade-off between fairness and performance. If mutation is applied too aggressively, the repaired data may drift excessively from the original distribution and hurt predictive performance. For this reason, Fairabel is not designed to optimize fairness alone. It introduces a mutation budget together with a multi-objective mechanism to balance fairness improvement and performance preservation. Therefore, the method should not be interpreted as an unconstrained heuristic, but as a controlled repair strategy with explicit trade-offs. We select α on a validation set by minimizing the following multi-objective loss:

$$\mathcal{L}(\alpha) = \big[1 - \text{Accuracy}(\alpha)\big] + \big[1 - \text{F1}(\alpha)\big] + \big|\text{SPD}(\alpha)\big| + \big|\text{AOD}(\alpha)\big| + \big|\text{EOD}(\alpha)\big|,$$

where all metrics are computed after retraining on data mutated with a budget α. The fairness terms are expressed as non-negative magnitudes (absolute differences), where lower values indicate better fairness. The optimal value α^* obtained from multi-objective optimization is then used to generate the final mutated training dataset. Furthermore, the loss function design can be dynamically customized to meet specific performance or fairness metric requirements. We provide the specific implementation parameters, including tools and heuristic configurations, in Section 4.7, and explore the impact of loss function design in Section 5.5.

Extension for Handling Intersectional Bias Across Multiple Sensitive Attributes. In practical scenarios, unfairness may arise only at the intersectional level, even when it is not apparent for any single sensitive attribute considered in isolation. In this setting, the mutation budget is determined using intersectional fairness criteria, namely ISPD, IAOD, and IEOD, which measure disparities across intersectional subgroups rather than across a single sensitive attribute at a time.

Let $A_i = (A_i^{(1)}, \ldots, A_i^{(m)})$ denote the vector of m sensitive attributes, where each attribute is encoded such that 0 represents the unprivileged group and 1 represents the privileged group. Let T_i denote the non-sensitive features. We then construct the *intersectional* candidate set.

$$C_{\text{multi}} \;=\; \{\, i \mid A_i^{(1)} = \cdots = A_i^{(m)} = 0,\; Y_i = 0 \,\}.$$

For each $i \in C_{\text{multi}}$, we compute the probability of a favorable label using the learned causal model, $p_i = \mathcal{M}_1(A_i, T_i)$, sort candidates by p_i in *descending* order, and set the top α fraction of label to 1 exactly as in the single-attribute case.

This joint condition ensures that only instances whose unfavorable labels are simultaneously associated with all sensitive attributes are mutated, thereby directly targeting intersectional bias.

The mutation budget α is optimized using the following multi-objective loss function:

$$\mathcal{L}(\alpha) = [1 - \text{Accuracy}(\alpha)] + [1 - \text{F1}(\alpha)] + |\text{ISPD}(\alpha)| + |\text{IAOD}(\alpha)| + |\text{IEOD}(\alpha)|,$$

where the fairness terms are intersectional metrics that capture worst-case disparities across all subgroups. This formulation ensures that the optimization explicitly targets intersectional unfairness rather than single-attribute disparities.

3.3.2 Sensitive Attribute Mutation. The process for sensitive attribute mutation operates analogously to data label mutation, with the primary distinction lying in the optimization target and the causal model employed. Specifically, we use the same candidate set C defined in Section 3.3.1. However, instead of scoring based on favorable label probability, we employ a second learned causal model, $\mathcal{M}_2$, to estimate the probability of the privileged attribute given the label and non-sensitive features:

$$p_i \;=\; \mathcal{M}_2(Y_i, T_i) \in [0, 1].$$

Similarly, let $P = \{(i, p_i) \mid i \in C\}$ be the set of index-probability pairs. Using the same strategy as mentioned in Section 3.3.1, we identify the top-k candidates with the highest scores p_i and mutate their sensitive attributes:

$$\widehat{A_i} \;=\; \begin{cases} 1, & \text{if } i \in \text{Top-}k(P), \\ A_i, & \text{otherwise.} \end{cases}$$

This effectively reassigns the selected unprivileged instances to the privileged group. The hyperparameter optimization for α and the extension to intersectional fairness follow the exact same procedures as described in Section 3.3.1.

3.3.3 Comparison of Strategies. While both strategies aim to strengthen the correlation between the favorable label and the unprivileged attribute, they operate through distinct mechanisms. Data label mutation directly increases the proportion of favorable label within the unprivileged group ($Y = 0 \rightarrow 1$). In contrast, sensitive attribute mutation decreases the proportion of favorable label within the privileged group by injecting negative samples from the unprivileged group ($A = 0 \rightarrow 1$). We empirically compare their effectiveness in Section 5.3.

4 Evaluation

We evaluate Fairabel using a controlled, multi-case experiment that follows the Goal–Question–Metric (GQM) paradigm [8] and the experimentation guidelines by Wohlin et al. [86]. This section first states the overall evaluation goal, then derives the research questions (RQs), describes the objects and context of the study, defines the measures used, and finally presents the analysis procedures and implementation details.

4.1 Evaluation Goal and Research Questions

According to the GQM approach, our evaluation begins with an explicit goal formulated as follows: **Analyze** Fairabel and the baselines **for the purpose of** evaluating and comparing their ability to repair software fairness **with respect to** model fairness and performance **from the point of view of** software engineers and fairness researchers who need to repair model fairness **in the context**

of binary classification tasks on tabular benchmark datasets widely used in the software fairness literature. According to the evaluation goal, we derive the following RQs:

- **RQ1: Single-attribute effectiveness:** How effective is Fairabel in repairing single-attribute fairness while preserving performance, compared to state-of-the-art methods?
- **RQ2: Intersectional effectiveness:** How effective is Fairabel in repairing intersectional fairness involving multiple sensitive attributes while preserving performance?
- **RQ3: Impact of mutation strategies:** To what extent can mutation strategies (label mutation vs. sensitive attribute mutation) influence the effectiveness of Fairabel?
- **RQ4: Impact of causal model:** To what extent can the guidance model (causal model vs. non-causal model) influence the effectiveness of Fairabel?
- **RQ5: Impact of loss function:** To what extent can the loss function design influence the trade-off of Fairabel between model fairness and performance?

Each RQ is answered by evaluating Fairabel and the baselines against specific metrics. The definition and calculation of these metrics are detailed in Section 4.5. This specification of metrics constitutes the final component of our GQM experimental design, closing the derivation loop.

4.2 Hypothesis and Its Rationale

To statistically assess the effectiveness of Fairabel, we formulate the following null and alternative hypotheses regarding the distribution of evaluation metrics, including both fairness and performance metrics, across the decision-making scenarios:

- **Null hypothesis** (H_0): Fairabel and the baseline method follow the same distribution for the considered metric ($D_{\text{Fairabel}} \sim D_{\text{baseline}}$).
- **Alternative hypothesis** (H_1): Fairabel and the baseline method follow different distributions for the considered metric ($D_{\text{Fairabel}} \nsim D_{\text{baseline}}$).

If H_0 holds, then Fairabel does not differ statistically from the baseline on the metric under consideration. If H_0 is rejected, we conclude that Fairabel and the baseline exhibit a statistically significant distributional difference.

Rationale for H_1: The rationale for the alternative hypothesis stems from the methodological divergence between Fairabel and the baselines. While baselines typically operate through static paradigms (ranging from pre-processing to post-processing) that lack dynamic adjustment, Fairabel integrates a causal model with multi-objective optimization. This combination provides an active feedback mechanism that guides data mutation toward explicitly minimizing the global loss of both fairness and performance. These fundamental differences in optimization dynamics are expected to yield statistically distinguishable metric distributions. Furthermore, considering the two-tailed nature of H_1, we introduce the win/tie/loss rate as an additional metric to demonstrate the specific advantage of Fairabel when the null hypothesis (H_0) is rejected. The details of the statistical analysis and the win/tie/loss rate are presented in Section 4.6.

4.3 Baseline Methods

To ensure a robust and up-to-date comparison, we select our baselines based on recent software engineering studies, including empirical evaluations of fairness repair methods [22, 24] and the latest proposed repair techniques [89]. These baselines represent strong and relevant methods for evaluating the effectiveness of Fairabel. We briefly summarize these baselines below.

- LTDD (proposed at ICSE'22) [56] uses a linear model to identify features that are linearly related to the sensitive attributes and removes their influence from the feature set;

- MAAT (proposed at FSE'22) [21] ensembles the predictions of a fairness-optimized model and a performance-optimized model;
- FairMask (proposed at TSE'23) [68] trains extrapolation models to predict and subsequently adjust sensitive attribute values in the test set;
- MirrorFair (proposed at FSE'24) [89] adaptively combines two predictions from the original model and another model trained on the flipped training data.

4.4 Benchmark Datasets and ML Classifiers

We use the same set of datasets as a recent ICSE'24 fairness study [22], covering five widely adopted benchmarks: the Adult Income dataset (Adult) [5], the ProPublica Recidivism dataset (COMPAS) [72], the Default of Credit Card Clients dataset (Default) [3], the Medical Expenditure Panel Survey 2015 dataset (Mep1) [4], and the Medical Expenditure Panel Survey 2016 dataset (Mep2) [2]. These datasets span multiple domains, including finance, social welfare, healthcare, and justice. Table 1 summarizes their key characteristics. Since all five datasets contain two sensitive attributes, we design ten single-attribute tasks, corresponding to five datasets × two sensitive attributes, to evaluate the effectiveness of Fairabel in repairing fairness for each attribute individually. We also design five additional tasks to assess the ability of Fairabel to repair intersectional fairness involving both sensitive attributes simultaneously.

Table 1. Benchmark datasets.

Dataset	Sensitive attributes	Size	Features	Favorable label
Adult	Sex, Race	48,843	7	1 (income >50k)
Compas	Sex, Race	7,214	7	0 (no recidivism)
Default	Sex, Age	30,000	23	0 (no default)
MEP1	Sex, Race	15,830	42	1 (utilizer)
Mep2	Sex, Race	15,675	42	1 (utilizer)

For machine learning classifiers, we follow previous studies [22, 24] and adopt four widely used models: Logistic Regression (LR) [50], Random Forest (RF) [13], Support Vector Machine (SVM) [40], and Deep Neural Network (DNN) [54]. The selection of benchmark datasets, classifiers, and their configurations is consistent with recent empirical investigations [22], ensuring the soundness of comparisons and the reliability of experimental results.

4.5 Evaluation Metrics

In line with previous research [21, 22, 24], we use the three most widely adopted fairness metrics and five commonly used ML performance metrics to assess the effectiveness of Fairabel.

4.5.1 Fairness Metrics. We introduced the concepts of single-attribute bias and intersectional bias in Section 2.1.1 and Section 2.1.2. Considering the diversity of real-world data distributions, bias may be mismeasured when evaluated solely in a single-attribute setting, while becoming more apparent in intersectional scenarios that simultaneously involve multiple sensitive attributes, such as race and sex. Therefore, we adopt multiple single-attribute fairness metrics together with intersectional fairness metrics to mitigate potential bias mismeasurement [57].

In this subsection, we present the formulas used to calculate these metrics. Let A denote the sensitive attribute, where $A = 1$ represents the privileged group and $A = 0$ represents the unprivileged group. Let Y denote the true label, where $Y = 1$ indicates a favorable label and $Y = 0$ indicates an unfavorable label. $\hat{Y}$ denotes the predicted label, and $P(\cdot)$ denotes the corresponding probability.

SPD measures the disparity in favorable rates between the privileged and unprivileged groups, which can be calculated by:

$$SPD = |P[\hat{Y} = 1|A = 1] - P[\hat{Y} = 1|A = 0]| \tag{1}$$

AOD measures the average disparity in false-positive rates and true-positive rates between the privileged and unprivileged groups, which can be calculated by:

$$\begin{aligned} AOD = &\frac{1}{2}(|P[\hat{Y} = 1|A = 0, Y = 0] - P[\hat{Y} = 1|A = 1, Y = 0] \\ &+ P[\hat{Y} = 1|A = 0, Y = 1] - P[\hat{Y} = 1|A = 1, Y = 1]|) \end{aligned} \tag{2}$$

EOD measures the disparity in true-positive rates between the privileged and unprivileged groups, which can be calculated by:

$$EOD = |P[\hat{Y} = 1|A = 0, Y = 1] - P[\hat{Y} = 1|A = 1, Y = 1]| \tag{3}$$

Let s denote an intersectional subgroup and S denote the intersectional subgroup set. Intersectional Statistical Parity Difference (ISPD) measures the maximum disparity in favorable rates among all subgroups, which can be calculated by:

$$ISPD = \max_{s \in S} P[\hat{Y} = 1|A = s] - \min_{s \in S} P[\hat{Y} = 1|A = s] \tag{4}$$

Intersectional Average Odds Difference (IAOD) measures the maximum average disparity in false-positive rates and true-positive rates among all subgroups, which can be calculated by:

$$\begin{aligned} IAOD = &\frac{1}{2}[\max_{s \in S}(P[\hat{Y} = 1|A = s, Y = 0] + P[\hat{Y} = 1|A = s, Y = 1]) \\ &- \min_{s \in S}(P[\hat{Y} = 1|A = s, Y = 0] + P[\hat{Y} = 1|A = s, Y = 1])] \end{aligned} \tag{5}$$

Intersectional Equal Opportunity Difference (IEOD) measures the maximum disparity in true-positive rates among all subgroups, which can be calculated by:

$$IEOD = \max_{s \in S} P[\hat{Y} = 1|A = s, Y = 1] - \min_{s \in S} P[\hat{Y} = 1|A = s, Y = 1] \tag{6}$$

4.5.2 Performance Metrics. Precision, recall, F1-score, accuracy, and Matthews Correlation Coefficient (MCC) are standard metrics for evaluating ML classifiers and are widely used in fairness studies [21, 22, 24]. For a given class, precision measures the proportion of predicted positives that are correct, while recall measures the proportion of actual positives that are identified. The F1-score is the harmonic mean of precision and recall. Accuracy measures the overall proportion of correct predictions. MCC summarizes classification quality by considering true and false positives and negatives, making it suitable for imbalanced datasets. It is calculated as follows:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}, \tag{7}$$

where True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) represent the counts of favorable samples predicted as favorable, unfavorable samples predicted as unfavorable, unfavorable samples predicted as favorable, and favorable samples predicted as unfavorable, respectively. We report precision, recall, and F1-score using the macro-average for each class, following the approach used in prior studies [21, 22]. This involves computing the scores for each class individually and then taking their arithmetic mean, ensuring that no single class dominates the overall performance metrics. This approach is particularly important for evaluating imbalanced datasets [21, 22].

4.6 Statistical Analysis

To evaluate the significance of the comparisons between Fairabel and existing methods, we conduct statistical tests. First, we inspect the distribution of the experimental data, i.e., the performance and fairness metric values obtained from 20 repeated runs, using the Shapiro-Wilk test [78]. The results indicate that some distributions are normal, whereas a substantial proportion (18.02%) deviates from normality ($p < 0.05$), making parametric tests unsuitable for our setting. Considering the limitations of the Shapiro-Wilk test in large-sample settings, we further adopt the Anderson-Darling test [60] to examine normality, which finds a similar proportion (18.00%) of distributions deviating from normality. Given the results of both the Shapiro-Wilk and Anderson-Darling tests, we adopt non-parametric statistical procedures. For *pairwise* comparisons between Fairabel and each baseline within a scenario, we apply the Mann-Whitney U test, also known as the Wilcoxon rank-sum test, to assess distributional differences [58], and consider results statistically significant when $p < 0.05$.

Since our evaluation involves multiple baselines and tasks, it is necessary to rigorously assess whether the observed best method is statistically superior to each competing baseline after correcting for multiple comparisons. Therefore, we additionally conduct Dunn's post-hoc pairwise tests with Holm–Bonferroni correction to control the family-wise error rate [33, 42]. We regard differences as significant when $p_{\text{adj}} < 0.05$. In head-to-head summaries, two win/tie/loss outcomes are determined for each scenario based on the Mann–Whitney U test and the Dunn test, respectively.

4.7 Implementation Details

Fairabel Configuration. We employ the Stan [83] tool to learn the causal models, guiding the following mutation. Regarding the loss function defined in Section 3, we assign equal weights (1.0) to all performance and fairness terms in the loss function by default to ensure a balanced trade-off. To determine the optimal mutation budget α^*, the parameter optimization involves a grid search over the range $[0, 1]$ with a step size of 0.01, which empirically balances searching comprehensiveness and cost. The parameter α^* governs the proportion of candidates selected for mutation, effectively functioning as a probability threshold within the ranked candidate list.

Experimental Setup. We employ AIF360 [9], Scikit-learn [15, 67], and TensorFlow Keras [32] for the implementation of existing fairness repair methods, ML classifiers, and evaluation metrics. AIF360, an IBM-developed tool for fairness research, includes the fairness metrics calculation packages we used for measuring the single-attribute bias. Regarding the intersectional bias measurement, we use definitions and calculations proposed by a recent empirical investigation [22]. The state-of-the-art baseline methods are replicated based on the code released by recent papers [22–24, 56]. To ensure the reliability and validity of our evaluation of Fairabel and the comparative methods, we align the model training parameter settings with prior work [22]. Specifically, we adopt the default parameter settings of the LR, RF, and SVM models provided by the Scikit-learn toolkit [15, 67]. For the deep neural network, we employ a six-layer densely connected architecture, using the ReLU activation function in the first five layers and the sigmoid activation function in the final layer, with binary cross-entropy as the loss function, the Nadam optimizer with a default learning rate of 0.001, 20 training epochs, and a batch size of 128. All experiments are conducted on a macOS platform equipped with an M1 Pro processor (integrated CPU and GPU), 16 GB RAM, Python 3.11.14, TensorFlow 2.19.1, and macOS Sequoia.

5 Results

In this section, we present our experimental results by answering the proposed RQs.

5.1 RQ1: Single-Attribute Effectiveness

We answer RQ1 from three complementary views. In RQ1.1, we report the specific metric values for each decision-making scenario before and after applying Fairabel. In RQ1.2, we present the relative changes in performance and fairness metrics after applying different fairness repair methods. In RQ1.3, we perform statistical analysis to examine whether Fairabel statistically outperforms the baselines using the Mann–Whitney U-test [58] and the Dunn test [33].

Table 2. RQ1.1: Performance and fairness metrics of ML software before and after applying our approach. Regarding performance metrics, the higher the better. Regarding fairness metrics, the lower, the better.

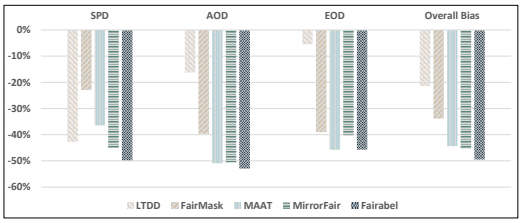
Task	Model	LR								RF							
		Performance Metric					Fairness Metric			Performance Metric					Fairness Metric		
		Accuracy	Recall	Precision	F1-Score	MCC	SPD	AOD	EOD	Accuracy	Recall	Precision	F1-Score	MCC	SPD	AOD	EOD
Adult-Sex	Before	0.82	0.68	0.78	0.71	0.45	0.17	0.18	0.28	0.84	0.73	0.79	0.75	0.52	0.18	0.15	0.23
	After	0.80	0.70	0.72	0.71	0.42	0.07	0.01	0.02	0.81	0.73	0.74	0.73	0.47	0.06	0.03	0.02
Adult-Race	Before	0.82	0.68	0.78	0.71	0.45	0.09	0.09	0.13	0.84	0.73	0.79	0.75	0.52	0.09	0.07	0.10
	After	0.82	0.69	0.77	0.71	0.45	0.06	0.02	0.03	0.83	0.73	0.78	0.75	0.51	0.05	0.01	0.02
Compas-Sex	Before	0.68	0.67	0.68	0.67	0.35	0.22	0.19	0.15	0.65	0.64	0.64	0.64	0.28	0.12	0.09	0.08
	After	0.68	0.66	0.68	0.66	0.35	0.04	0.03	0.02	0.65	0.64	0.65	0.64	0.29	0.07	0.05	0.03
Compas-Race	Before	0.68	0.67	0.68	0.67	0.35	0.18	0.15	0.11	0.65	0.64	0.64	0.64	0.28	0.13	0.10	0.09
	After	0.68	0.66	0.68	0.67	0.34	0.04	0.02	0.03	0.65	0.64	0.65	0.64	0.29	0.04	0.03	0.03
Default-Sex	Before	0.81	0.60	0.77	0.61	0.32	0.03	0.03	0.04	0.81	0.65	0.74	0.68	0.38	0.03	0.02	0.02
	After	0.81	0.67	0.73	0.69	0.40	0.01	0.02	0.03	0.81	0.68	0.73	0.70	0.41	0.01	0.01	0.03
Default-Age	Before	0.81	0.60	0.77	0.61	0.32	0.04	0.04	0.05	0.81	0.65	0.74	0.68	0.38	0.05	0.04	0.06
	After	0.79	0.68	0.69	0.69	0.37	0.01	0.02	0.02	0.80	0.70	0.71	0.70	0.40	0.04	0.02	0.03
Mep1-Sex	Before	0.86	0.66	0.77	0.70	0.42	0.06	0.05	0.07	0.86	0.67	0.77	0.70	0.43	0.04	0.02	0.02
	After	0.86	0.68	0.76	0.71	0.44	0.04	0.01	0.02	0.86	0.68	0.76	0.70	0.43	0.04	0.01	0.02
Mep1-Race	Before	0.86	0.66	0.77	0.70	0.42	0.03	0.04	0.07	0.86	0.67	0.77	0.70	0.43	0.03	0.03	0.06
	After	0.86	0.68	0.77	0.71	0.44	0.02	0.02	0.04	0.86	0.69	0.75	0.72	0.44	0.01	0.03	0.04
Mep2-Sex	Before	0.86	0.64	0.76	0.67	0.38	0.06	0.06	0.09	0.86	0.64	0.76	0.67	0.38	0.04	0.01	0.02
	After	0.85	0.67	0.75	0.70	0.41	0.04	0.01	0.02	0.86	0.66	0.76	0.69	0.40	0.03	0.01	0.03
Mep2-Race	Before	0.86	0.64	0.76	0.67	0.38	0.04	0.05	0.08	0.86	0.64	0.76	0.67	0.38	0.03	0.02	0.03
	After	0.86	0.66	0.75	0.69	0.41	0.02	0.01	0.03	0.85	0.66	0.75	0.69	0.40	0.01	0.03	0.06
Task	Model	SVM								DNN							
		Performance Metric					Fairness Metric			Performance Metric					Fairness Metric		
		Accuracy	Recall	Precision	F1-Score	MCC	SPD	AOD	EOD	Accuracy	Recall	Precision	F1-Score	MCC	SPD	AOD	EOD
Adult-Sex	Before	0.82	0.67	0.79	0.70	0.44	0.14	0.12	0.19	0.83	0.71	0.79	0.73	0.49	0.19	0.18	0.28
	After	0.80	0.71	0.73	0.72	0.44	0.08	0.01	0.01	0.81	0.69	0.73	0.70	0.43	0.07	0.03	0.05
Adult-Race	Before	0.82	0.67	0.79	0.70	0.44	0.07	0.05	0.08	0.83	0.71	0.79	0.73	0.49	0.09	0.07	0.10
	After	0.82	0.70	0.77	0.72	0.46	0.04	0.02	0.03	0.83	0.72	0.78	0.74	0.49	0.04	0.02	0.04
Compas-Sex	Before	0.68	0.67	0.68	0.67	0.35	0.21	0.18	0.15	0.68	0.67	0.68	0.67	0.35	0.20	0.16	0.14
	After	0.68	0.66	0.68	0.66	0.35	0.03	0.02	0.02	0.67	0.66	0.68	0.65	0.34	0.08	0.06	0.05
Compas-Race	Before	0.68	0.67	0.68	0.67	0.35	0.17	0.15	0.11	0.68	0.67	0.68	0.67	0.35	0.17	0.15	0.12
	After	0.68	0.66	0.68	0.66	0.34	0.04	0.02	0.03	0.69	0.67	0.69	0.67	0.36	0.11	0.09	0.06
Default-Sex	Before	0.80	0.57	0.77	0.58	0.28	0.02	0.02	0.03	0.82	0.65	0.74	0.67	0.38	0.03	0.03	0.03
	After	0.81	0.67	0.73	0.69	0.40	0.01	0.01	0.02	0.81	0.66	0.73	0.68	0.38	0.02	0.02	0.04
Default-Age	Before	0.80	0.57	0.77	0.58	0.28	0.02	0.02	0.02	0.82	0.65	0.74	0.67	0.38	0.06	0.06	0.08
	After	0.79	0.68	0.69	0.69	0.37	0.01	0.02	0.02	0.81	0.66	0.72	0.68	0.38	0.03	0.03	0.04
Mep1-Sex	Before	0.86	0.66	0.78	0.69	0.41	0.05	0.03	0.04	0.85	0.64	0.72	0.66	0.36	0.05	0.04	0.06
	After	0.86	0.68	0.77	0.71	0.43	0.04	0.01	0.02	0.85	0.67	0.72	0.68	0.39	0.03	0.03	0.06
Mep1-Race	Before	0.86	0.66	0.78	0.69	0.41	0.02	0.03	0.05	0.85	0.64	0.72	0.66	0.36	0.04	0.05	0.08
	After	0.86	0.68	0.77	0.71	0.43	0.02	0.02	0.03	0.86	0.70	0.75	0.71	0.44	0.01	0.03	0.05
Mep2-Sex	Before	0.85	0.63	0.76	0.66	0.37	0.05	0.04	0.05	0.85	0.64	0.72	0.66	0.36	0.05	0.05	0.07
	After	0.85	0.67	0.75	0.69	0.41	0.04	0.01	0.02	0.85	0.65	0.74	0.67	0.38	0.04	0.02	0.03
Mep2-Race	Before	0.85	0.63	0.76	0.66	0.37	0.03	0.03	0.04	0.85	0.64	0.72	0.66	0.36	0.05	0.06	0.09
	After	0.86	0.65	0.76	0.68	0.39	0.03	0.02	0.03	0.85	0.68	0.74	0.70	0.41	0.02	0.05	0.08

5.1.1 RQ1.1: Impact of Fairabel on Model Fairness and Performance Metrics. Table 2 presents five model performance metrics and three fairness metrics across 40 classification scenarios, corresponding to 10 tasks × 4 classifiers, before and after applying Fairabel. A visual inspection shows a consistent decrease in fairness metrics, namely SPD, AOD, and EOD, where lower values are better, after applying Fairabel across most tasks and all classifier families (LR, RF, SVM, and DNN). These reductions are evident in nearly all cases. Meanwhile, performance metrics remain largely stable, with only small and non-systematic fluctuations. These results indicate that Fairabel achieves a good trade-off between fairness and performance across all scenarios.

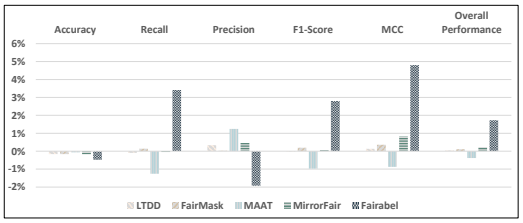
> **Answer to RQ1.1:** Fairabel consistently reduces model bias (SPD, AOD, EOD) in 114 of 120 cases (3 metrics × 10 tasks × 4 classifiers) while largely preserving model performance metrics.

5.1.2 RQ1.2: Relative Improvement of Fairabel over Existing Methods. Figure 2a and Figure 2b demonstrate the relative changes in SPD, AOD, EOD, accuracy, recall, precision, F1-score, and MCC. The "Overall Performance" denotes the average changes in the five performance metrics, while "Overall Bias" denotes the average changes in the three fairness metrics. Fairabel achieves improvement in recall, F1-score, and MCC, overall enhancing the performance metrics and outperforming the state-of-the-art in performance preservation. In terms of fairness, Fairabel can consistently and significantly reduce the SPD bias, AOD bias, and EOD bias.

Specifically, as shown in Figure 2a, Fairabel reduces all three types of bias by an average of 50%, outperforming the state-of-the-art method MirrorFair [89] with a 9% relative improvement. Regarding performance, Figure 2b illustrates that the state-of-the-art methods and Fairabel can generally preserve the original model performance, but exhibit varying behavior across different metrics. Specifically, Fairabel achieves superior recall, F1-score, and MCC across all decision-making scenarios, while the state-of-the-art method, MirrorFair, performs better on precision. Overall, Fairabel demonstrates the strongest performance preservation, surpassing MirrorFair.



(a) RQ1.2: Relative changes in fairness metrics after applying the Fairabel and baseline methods. We average the relative changes of three fairness metrics as the overall bias metrics change. Fairabel outperforms the state-of-the-art with a 9% relative improvement.



(b) RQ1.2: Relative changes in performance metrics after applying the Fairabel and baseline methods. We average the relative changes of five performance metrics as the overall performance metrics change. Fairabel excels in overall performance preservation.

Fig. 2. Comparison of the relative changes in metrics after applying Fairabel and state-of-the-art methods.

> **Answer to RQ1.2:** Fairabel reduces the SPD, AOD, and EOD by an average of 50% across the 40 decision-making scenarios we studied, outperforming the state-of-the-art method MirrorFair with higher fairness repair effectiveness and better overall performance preservation.

5.1.3 RQ1.3: Statistical Analysis. Table 3 presents the statistical win, tie, and loss distribution of Fairabel compared with the baselines under fairness and performance evaluations using the Mann–Whitney U-test [58] and the Dunn test [33]. For the comparison between Fairabel and the state-of-the-art method MirrorFair, null hypothesis H_0 is accepted ("tie") in 48%/69% of the cases and is rejected ("win" or "loss") in 52%/31% of the cases across 40 decision-making scenarios and three fairness metrics, under the Mann–Whitney U test and Dunn test, respectively.

A "win" rate higher than the "loss" rate indicates that Fairabel outperforms the corresponding baseline. Fairabel outperforms the baselines in both fairness and performance comparisons under two statistical tests. Specifically, under the Mann–Whitney U-test, regarding fairness, 33% of Fairabel cases win against the state-of-the-art cases, while 19% lose. Under the Dunn test, 20% of Fairabel cases win, while only 11% lose against the state-of-the-art cases. Regarding model performance, under the Mann–Whitney U-test, 40% of Fairabel cases win against the state-of-the-art cases, while

Table 3. RQ1.3: Win, tie, and loss distribution of fairness and performance comparisons between Fairabel and the baselines. "win" indicates that Fairabel outperforms the baseline, while "loss" indicates that Fairabel performs worse. We use "**bold**" to highlight cases where the "win" proportion exceeds the "loss" proportion. Both the Mann–Whitney U-test and the Dunn test show that Fairabel outperforms the baselines in both fairness and performance. Specifically, compared with the state-of-the-art method MirrorFair, Fairabel achieves 14% and 9% higher "win" rates under the Mann–Whitney U-test and Dunn test, respectively. ES_{avg} denotes the average Cliff's delta effect size across all scenarios.

Opponent	Fairness							Performance						
	Mann-Whitney U-Test			Dunn Test			ES_{avg}	Mann-Whitney U-Test			Dunn Test			ES_{avg}
	Win	Ties	Loss	Win	Ties	Loss		Win	Ties	Loss	Win	Ties	Loss	
Default	***82%***	17%	2%	***73%***	25%	2%	0.705	***36%***	36%	29%	***29%***	53%	19%	0.566
FairMask	***51%***	42%	8%	***40%***	54%	6%	0.488	***41%***	31%	29%	***29%***	54%	17%	0.564
LTDD	***49%***	32%	19%	***43%***	48%	10%	0.588	***40%***	31%	30%	***33%***	50%	18%	0.577
MAAT	***37%***	47%	17%	***23%***	72%	6%	0.453	***49%***	21%	30%	***41%***	35%	24%	0.662
MirrorFair	***33%***	48%	19%	***20%***	69%	11%	0.439	***42%***	32%	27%	***29%***	54%	18%	0.586

only 30% lose. Under the Dunn test, 29% of Fairabel cases win, while only 18% lose against the state-of-the-art cases.

Furthermore, to assess the practical magnitude of the improvements, we report the average Cliff's delta effect size (ES_{avg}) for each baseline comparison. Fairabel achieves medium to large effect sizes in fairness comparisons against all baselines, indicating that the observed improvements are not only statistically significant but also practically meaningful.

> **Answer to RQ1.3:** Fairabel statistically outperforms the baselines in both fairness and performance comparisons. Specifically, Fairabel achieves **14** and **9** percentage points higher "win" rates than the state-of-the-art method under the Mann–Whitney U-test and the Dunn test, respectively.

5.2 RQ2: Intersectional Effectiveness

This research question examines the effectiveness of Fairabel in addressing scenarios that involve multiple sensitive attributes using benchmarking datasets containing at least two sensitive attributes. We compare Fairabel with three baseline methods capable of mitigating intersectional bias across multiple sensitive attributes. Among them, MAAT and FairMask are the current state-of-the-art methods identified in a recent empirical study [22]. We also include MirrorFair as a baseline, which is a recently proposed fairness repair approach not covered in the aforementioned study.

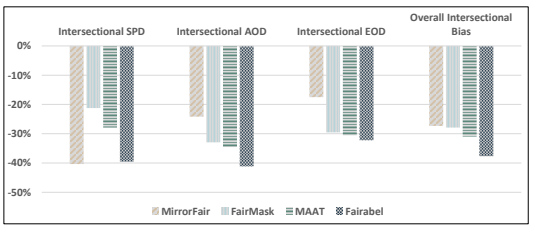


Fig. 3. RQ2: The relative changes in three intersectional bias metrics after applying different fairness repair methods. Fairabel reduces the overall bias by 38%, outperforming the state-of-the-art by 7 percentage points.

Figure 3 presents the intersectional bias reduction achieved by Fairabel and the baseline methods under multi-attribute scenarios. Fairabel reduces intersectional SPD, AOD, and EOD by 40%, 41%,

and 32%, respectively, outperforming the state-of-the-art with a 22% relative improvement on average across the three intersectional bias metrics.

> **Answer to RQ2:** Fairabel outperforms the state-of-the-art in improving fairness across multiple sensitive attributes simultaneously. It enhances intersectional fairness by 38% on average, surpassing the current best approach with a 22% relative improvement.

5.3 RQ3: Impact of Mutation Strategy

Data label and sensitive attributes are important properties of each data instance. Mutating data label or sensitive attributes can change the causal relationship between them. In this ablation experiment, we investigate the impact of different mutation properties on fairness repair by comparing the effectiveness of Fairabel when using data label mutation, sensitive attribute mutation, and a combination of both.

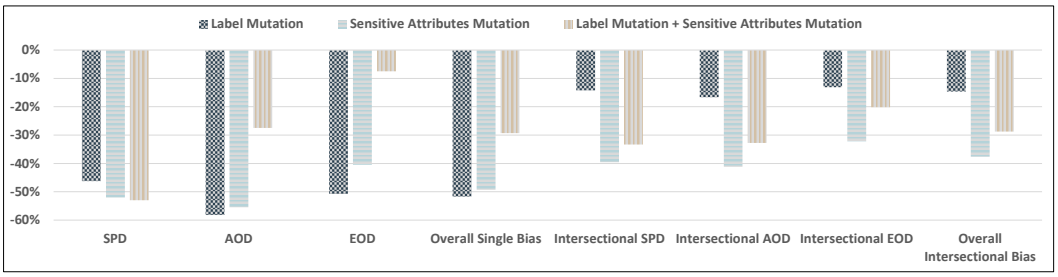


Fig. 4. RQ3: Relative changes in performance and fairness metrics after applying Fairabel with different data mutation strategies. Label mutation strategy excels in single-attribute scenarios and reduces the average bias by 52%. The sensitive-attribute mutation strategy excels in multiple-sensitive-attribute scenarios and reduces the average intersectional bias by 38%.

Figure 4 shows that data label mutation, sensitive attribute mutation, and their combination all significantly reduce both single-attribute and intersectional bias. However, different mutation strategies exhibit varying advantages across scenarios. For single-attribute bias, data label mutation achieves the best overall bias reduction, followed closely by sensitive attribute mutation, while their combination shows the lowest effectiveness among the three strategies. In contrast, for intersectional bias, sensitive attribute mutation achieves the highest effectiveness, whereas data label mutation and their combination perform less effectively.

> **Answer to RQ3:** Our exploration demonstrates that all three mutation strategies can significantly reduce both single-attribute and intersectional bias. Data label mutation is particularly effective for mitigating single-attribute bias, while sensitive attribute mutation performs best for mitigating intersectional bias. The combination does not achieve better effectiveness than the stronger individual strategy, whether data label mutation or sensitive attribute mutation.

5.4 RQ4: Impact of Causal Model

In this ablation experiment, we further investigate the impact of the causal model on the effectiveness of Fairabel by replacing it with alternative models or selection strategies for determining which data instances to mutate. Specifically, we compare the effects of using a causal model, a common ML model, and random selection on the fairness repair effectiveness of Fairabel.

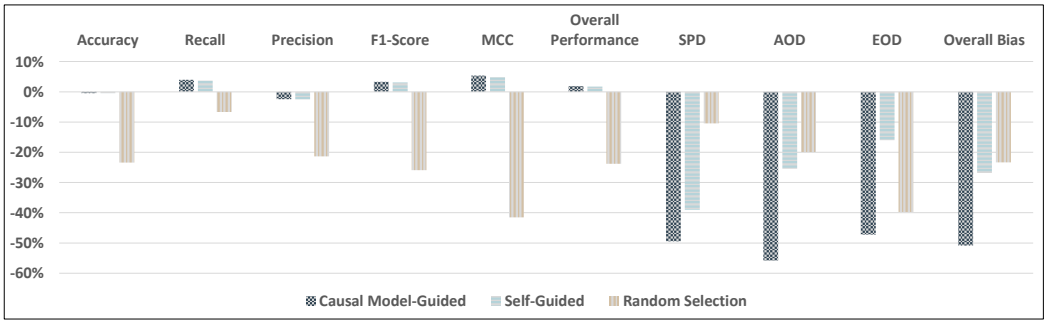


Fig. 5. RQ4: Relative changes in performance and fairness metrics after applying different candidate selection strategies in data mutation.

Figure 5 illustrates the changes in performance and fairness metrics after applying Fairabel with different mutation candidate selection strategies. "Causal Model-Guided" refers to using the proposed causal model to identify which data instances should be mutated from an unfavorable to a favorable label. "Self-Guided" refers to using the target model itself to determine mutation candidates (for example, using the LR model to guide data mutation for fairer LR model training). "Random Selection" refers to randomly selecting candidates from the original training data.

According to Figure 5, both the causal model-guided and self-guided strategies enable Fairabel to preserve the overall original performance, with the causal model-guided approach performing slightly better. In contrast, random selection of mutation candidates leads to a notable decline in overall model performance.

In terms of fairness repair effectiveness, the causal model-guided approach demonstrates a clear advantage over the other candidate selection strategies in reducing SPD, AOD, and EOD bias. On average, it outperforms the self-guided approach by more than 20 percentage points in reducing overall bias. This indicates that the advantage of the causal model does not stem from superior predictive accuracy, but from providing a more suitable ranking signal for fairness-oriented data mutation. This finding further suggests that the strength of the causal model lies not in its prediction accuracy, but in offering a more appropriate ranking signal for fairness-oriented data mutation.

> **Answer to RQ4:** The causal model slightly surpasses the self-guided strategy in preserving model performance and significantly surpasses it by more than 20 percentage points in reducing bias. The random selection strategy cannot repair fairness without compromising model performance.

5.5 RQ5: Impact of Loss Function

In Section 5.1, we show that Fairabel outperforms the state-of-the-art by achieving higher fairness improvement with lower performance degradation using the trade-off-first loss function introduced in Section 3.3. However, in certain critical decision-making tasks, even a one-percentage-point drop in performance may be unacceptable. Therefore, in this section, we investigate the impact of loss function design on Fairabel with the "LR" model, and develop a performance-first loss function by increasing the weight of performance metrics within the optimization objective.

Table 4 shows that Fairabel with the trade-off-first loss function significantly improves fairness in 29 out of 30 cases (10 tasks $\times$ 3 fairness metrics), while significantly decreasing performance in 10 out of 50 cases (10 tasks $\times$ 5 performance metrics) and causing an average precision degradation of

two percentage points. In contrast, Fairabel, with the performance-first loss function, significantly improves fairness in 20 out of 30 cases without significantly degrading performance in any of the tasks studied.

Table 4. RQ5: Mann–Whitney U-test results and overall relative changes in each performance and fairness metric after applying Fairabel with the trade-off-first loss function and the performance-first loss function. "Sig-Better" indicates that the metric improves significantly after applying Fairabel with the corresponding loss function, highlighted in Light Green Background. "Sig-Worse" indicates that the metric significantly worsens. "Not-Sig" indicates that the change is not statistically significant, highlighted in Light Blue Background. When applying Fairabel with the performance-first loss function, no metric shows a statistically significant decline across any of the tasks studied.

Task	Loss Function	Accuracy	Recall	Precision	F1-score	MCC	SPD	AOD	EOD
Adult-Sex	Trade-off-First	Sig-Worse	Sig-Better	Sig-Worse	Sig-Better	Sig-Worse	Sig-Better	Sig-Better	Sig-Better
	Performance-First	Not-Sig	Sig-Better	Not-Sig	Sig-Better	Sig-Better	Not-Sig	Not-Sig	Not-Sig
Adult-Race	Trade-off-First	Not-Sig	Sig-Better	Sig-Worse	Sig-Better	Sig-Better	Sig-Better	Sig-Better	Sig-Better
	Performance-First	Not-Sig	Sig-Better	Not-Sig	Sig-Better	Sig-Better	Sig-Better	Sig-Better	Sig-Better
Compas-Sex	Trade-off-First	Not-Sig	Not-Sig	Sig-Better	Sig-Worse	Not-Sig	Sig-Better	Sig-Better	Sig-Better
	Performance-First	Not-Sig	Not-Sig	Sig-Better	Not-Sig	Not-Sig	Sig-Better	Sig-Better	Sig-Better
Compas-Race	Trade-off-First	Not-Sig	Not-Sig	Not-Sig	Not-Sig	Not-Sig	Sig-Better	Sig-Better	Sig-Better
	Performance-First	Not-Sig	Not-Sig	Not-Sig	Not-Sig	Not-Sig	Sig-Better	Sig-Better	Sig-Better
Default-Sex	Trade-off-First	Sig-Better	Sig-Better	Sig-Worse	Sig-Better	Sig-Better	Sig-Better	Sig-Better	Sig-Better
	Performance-First	Sig-Better	Sig-Better	Not-Sig	Sig-Better	Sig-Better	Sig-Better	Sig-Better	Sig-Better
Default-Age	Trade-off-First	Sig-Worse	Sig-Better	Sig-Worse	Sig-Better	Sig-Better	Sig-Better	Sig-Better	Sig-Better
	Performance-First	Sig-Better	Sig-Better	Not-Sig	Sig-Better	Sig-Better	Not-Sig	Sig-Better	Sig-Better
Mep1-Sex	Trade-off-First	Not-Sig	Sig-Better	Sig-Worse	Sig-Better	Sig-Better	Sig-Better	Sig-Better	Sig-Better
	Performance-First	Not-Sig	Sig-Better	Not-Sig	Sig-Better	Sig-Better	Sig-Better	Sig-Better	Sig-Better
Mep1-Race	Trade-off-First	Not-Sig	Sig-Better	Not-Sig	Sig-Better	Sig-Better	Sig-Better	Sig-Better	Not-Sig
	Performance-First	Not-Sig	Sig-Better	Not-Sig	Sig-Better	Not-Sig	Not-Sig	Not-Sig	Not-Sig
Mep2-Sex	Trade-off-First	Not-Sig	Sig-Better	Sig-Worse	Sig-Better	Sig-Better	Sig-Better	Sig-Better	Sig-Better
	Performance-First	Not-Sig	Sig-Better	Not-Sig	Sig-Better	Sig-Better	Sig-Better	Sig-Better	Not-Sig
All Tasks	Average Relative Change of Each Performance and Fairness Metric after Applying Fairabel with Different Loss Functions								
	Loss Function	Accuracy	Recall	Precision	F1-score	MCC	SPD	AOD	EOD
	Trade-off-First	0%	4%	-2%	4%	6%	-57%	-72%	-69%
	Performance-First	0%	2%	0%	2%	4%	-18%	-34%	-38%

Answer to RQ5: Fairabel supports diverse fairness improvement requirements through flexible loss function design. When using the performance-first loss function, Fairabel improves fairness without significantly degrading the original model performance.

6 Discussion

In this section, we analyze the experimental results and discuss the potential threats to their validity.

6.1 Analysis of Fairabel

6.1.1 Why can Fairabel outperform state-of-the-art methods? The advantages of Fairabel can be attributed to three key factors. First, benefiting from the active feedback mechanism within the multi-objective optimization framework, Fairabel can dynamically identify and mitigate harmful data mutations. This allows Fairabel to effectively mutate data in the absence of ground truth, thereby outperforming baselines in reducing overall bias.

Second, Fairabel balances the effectiveness across different fairness metrics by assigning equal weights to fairness terms in the loss function. Fairabel achieves a consistent improvement, with a maximum disparity (i.e., the gap between the highest and lowest reduction rates) of only 7%

across SPD, AOD, and EOD. In contrast, existing methods lack such a global balancing mechanism, resulting in significant unevenness in effectiveness. For instance, the state-of-the-art method, MirrorFair, reduces AOD by 51% but EOD by only 40%, yielding an effectiveness disparity of 11 percentage points. This gap widens to 15% for MAAT and 38% for LTDD.

Finally, the causal model provides more precise mutation candidate rankings by capturing the correlations between sensitive attributes and the label. As demonstrated in Section 5.4, even under the same optimization framework, the causal model outperforms non-causal models and random selection by more than 20 percentage points.

6.1.2 Why is label mutation better than sensitive-attribute mutation strategy in repairing single-attribute fairness, but weaker than sensitive-attribute mutation strategy in repairing intersectional fairness? Our experimental results in Section 5.3 demonstrate that the label mutation strategy performs better than sensitive-attribute mutation in preserving model performance and repairing single-attribute fairness. This is because the mutation of the data label not only changes the correlation between sensitive attributes and label but also changes the correlation between non-sensitive attributes and label. This empowers the label mutation strategy to calibrate both biased correlations between the label and sensitive attributes, as well as between label and non-sensitive attributes, thereby enhancing some performance metrics while improving model fairness.

For sensitive attribute mutation, this strategy only changes the correlation between the specific sensitive attribute and the data label. Thus, it is less effective than label mutation in single-attribute scenarios. However, this characteristic benefits the sensitive-attribute mutation strategy in mitigating intersectional bias involving at least two sensitive attributes, because it can separately modify the correlation between different sensitive attributes and label without interfering with each other. In label mutation, the operation of repairing one single-attribute fairness can significantly change the correlation involving another sensitive attribute. If the mutation on a label has the opposite effect on the bias of different sensitive attributes, the effectiveness of mitigating intersectional bias can drop significantly.

6.1.3 Effective Scope of Fairabel. The mutation process in Fairabel aims to adjust the proportion of favorable labels across different groups in the training data, thereby influencing the behavior of downstream models. After mutation, the repaired training data are expected to exhibit a more balanced distribution of favorable labels across privileged and unprivileged groups, which can help downstream models learn decision patterns that are less dependent on sensitive attributes. Fairabel is expected to be effective when model unfairness primarily originates from an imbalanced association between sensitive attributes and labels in the observed training data, and when this association can be mitigated through controlled data-level repair.

6.1.4 Where can Fairabel fail? Fairabel works based on causal modeling and multi-objective optimization, and it can fail in certain extreme situations.

(1) Causal modeling failure: Causal modeling necessitates capturing the dependencies among all data features and the label. However, the computational complexity of this process grows super-exponentially with the number of feature dimensions [25]. Consequently, while Fairabel demonstrates effectiveness in our current experimental setups (where feature dimensions range from 7 to 42), applying it to high-dimensional tasks (e.g., those exceeding 100 features) may present significant computational challenges or even fail.

(2) Multi-objective optimization failure: Fairabel relies on determining the optimal mutation budget α^* via multi-objective optimization. In scenarios with extremely small datasets (e.g., fewer than 1,000 instances), this optimization process may become susceptible to overfitting on the

validation set, thereby compromising generalization to the test set. While Fairabel demonstrates robustness on the benchmarks used in this study (ranging from 6,000 to 40,000 instances), its applicability to data-scarce tasks remains a potential limitation.

(3) Mutation candidate selection failure: Fairabel supports intersectional fairness repair by incorporating intersectional fairness metrics into the optimization objective and applying mutation to an intersectional candidate set. However, this process requires the privileged and unprivileged groups for mutation to be specified in advance. In extreme cases, bias may not be evident for any individual sensitive attribute but may appear only at the intersectional level and be captured by intersectional fairness metrics. In such scenarios, Fairabel may fail to select appropriate candidates for mutation. To mitigate this risk, practitioners should conduct a preliminary subgroup-level analysis before applying Fairabel. For example, if such an analysis shows that Black-female instances form the unprivileged group, practitioners should target this subgroup for intersectional mutation, rather than directly carrying over results from single-attribute analysis for intersectional bias mitigation.

(4) Sensitive-attribute mutation failure: Fairabel mitigates data bias through label mutation or sensitive-attribute mutation. The sensitive-attribute mutation strategy may raise semantic concerns because attributes such as race or sex are not naturally mutable in some real-world settings. In scenarios where sensitive attributes are legitimately relevant to the corresponding decision, mutating such attributes may not be practically valid. Therefore, sensitive-attribute mutation should be regarded as a potential data repair strategy for fairness improvement, rather than as a realistic causal intervention.

6.1.5 Relation to Inverse Probability Weighting. Inverse Probability Weighting (IPW) [41, 76] reweights data instances to approximate an interventional distribution, thereby simulating the removal of a specific causal pathway, for example by blocking a known edge $A \rightarrow T$ in the causal graph, where T denotes a non-sensitive attribute. Fairabel differs in that it does not assume unfairness is generated by one known edge or one explicitly identifiable mediation pathway. Instead, it addresses fairness at the level of training-data repair, using the learned causal structure as guidance for where mutations are more likely to reduce unfair dependence. Thus, IPW and Fairabel serve different purposes: IPW aims to approximate an interventional distribution under a specified causal estimand, whereas Fairabel aims to construct a repaired training set for fairer downstream learning.

6.1.6 Ethical Considerations of Label Mutation. Fairabel generates a virtual dataset during pre-processing solely to guide model training, following prior fairness repair techniques [56, 68, 89]. The mutated labels are synthetic and used only within the training loop; they are not used for downstream evaluation, reporting, deployment, or data sharing. Using these mutated labels for such purposes would be inappropriate, as they do not represent ground truth.

6.1.7 Extra Computational Overhead. Similar to existing pre-processing fairness repair methods, Fairabel introduces additional computational overhead due to processing the training data. To assess this overhead, we compute the average cost across three repetitions of causal model construction and mutation across the five datasets studied. Specifically, the average costs are 31.0, 20.0, 53.7, 32.1, and 31.1 seconds for the Adult, Compas, Default, Mep1, and Mep2 datasets, respectively. These costs occur only during the training data mutation stage and do not increase computational expense during deployment or model inference. Therefore, although Fairabel introduces additional overhead, the cost should be acceptable given the fairness improvements it provides for ML software.

6.1.8 Implications of Fairabel for Software Engineering. We discuss the implications of Fairabel for Software Engineering from three perspectives involving software engineers, SE researchers, and the SE community.

Implications for software engineers: Treating fairness as a critical dimension of software quality [14] and a core ethical duty [21, 24] is increasingly essential. Neglecting this shift risks violating anti-discrimination laws [1], as exemplified by high-profile failures like the recruitment tool from Amazon mentioned in Section 1. To address these challenges, our study provides empirical benchmarks across forty decision-making scenarios, helping engineers assess model bias risks. Furthermore, Fairabel offers a practical solution to repair fairness defects, thereby mitigating the legal liabilities associated with discriminatory outcomes [22].

Implications for SE researchers: Previous research [17, 18] demonstrates that there is a potential risk associated with the pre-processing practice of mutating data instances for fairness repair when ground truth and indications of fair data distributions are unavailable. Fairabel offers an important reference to control such risk from both causal and optimization perspectives. Additionally, our experimental results in RQ1 and RQ5 demonstrate that repairing fairness involves not only fairness–performance trade-offs [46] but also trade-offs across different metrics within fairness and within performance, which reveals promising research opportunities. Furthermore, the development of Fairabel reinforces ongoing efforts by SE researchers to fulfill ethical responsibilities and software quality control [35, 79] in delivering fair ML software services [22, 23].

6.2 Threats to Validity

6.2.1 Internal Validity. Our choice of measurements and metrics may introduce potential threats to the validity of our experimental results. To mitigate these risks, we have aligned our methodology with established research [89] and recent empirical studies [24]. We employed multiple performance and fairness metrics to ensure a thorough evaluation of Fairabel's effectiveness. However, variations in the definitions or implementations of fairness metrics could affect the outcomes. By using well-documented and widely accepted metrics, we aimed to minimize potential inconsistencies, yet there remains a risk that different operationalizations of fairness might yield different results.

6.2.2 External Validity. The selection of benchmark datasets, algorithms, and comparative methods could impact the external validity of our findings. To address these concerns, we followed experimental settings established in prior studies to ensure comparability. Our experiments spanned five widely recognized benchmark datasets covering diverse domains: social welfare [5], financial [3], healthcare [2, 4], and justice [72]. We tested four prevalent machine learning algorithms, including LR, RF, SVM, and DNN, ensuring the broad applicability of our results. While we aimed for generalizability, differences in domain-specific dataset characteristics or biases could impact the robustness of our results in other applications.

Additionally, because of practical limitations, conducting experiments on real deployed decision-making systems is not feasible due to the absence of publicly available systems with accessible training data or source code. Fairness-critical systems, such as credit scoring, hiring, and healthcare systems, typically involve sensitive user data and proprietary models, and therefore cannot be released for research purposes because of privacy, legal, and ethical constraints. This also helps explain why recent fairness studies [17, 55, 68] rely on widely adopted real-world datasets like ours, rather than on real deployed software systems. Our evaluation setup follows this established practice in the field.

6.2.3 Construct Validity. Construct validity may be threatened by the definitions and interpretations of "fairness" used in our study. Fairness remains a complex and multifaceted concept, with varying definitions and perspectives depending on context. Although we used widely accepted

fairness definitions in line with prior studies [17, 21, 34, 48, 68], the operationalization of these definitions may not capture all nuances, especially in real-world applications. To mitigate this, we utilized a range of fairness metrics to evaluate our model comprehensively. However, future studies could further enhance construct validity by investigating additional fairness criteria that may apply across different domains.

6.2.4 Conclusion Validity. Conclusion validity could be impacted by statistical or algorithmic variability. Given that fairness metrics can be sensitive to sampling variations or algorithmic adjustments, we conducted multiple trials for each experiment to account for such variability. Additionally, we applied statistical significance testing to assess the robustness of our findings. Despite these efforts, inherent limitations in sample sizes or algorithmic randomness may still influence the results.

6.3 Future Work

6.3.1 Extending Fairabel to Non-Tabular Data Tasks. Fairabel is a pre-processing method that relies on explicit quantitative sensitive and non-sensitive attributes, as well as data labels—information that is naturally available in tabular data but not directly observable in other data types such as images or text. Extending Fairabel to non-tabular modalities would require reliable extraction of structured attributes (e.g., via vision or language encoders) and well-defined label representations, which remains an open research challenge implying significant research opportunities [7, 19, 31].

As future work, we plan to derive applicable sensitive attributes and labels from task annotations and metadata and adapt Fairabel to non-tabular modalities. We will explore these extensions in question-answering and chatbot scenarios, as well as other multi-modal benchmarks.

6.3.2 Extending Fairabel to Pre-Trained Models. When training data cannot be accessed, existing fairness research typically adopts post-processing approaches, e.g., FairMask in our baselines, because they operate purely on model predictions without requiring access to the pre-training corpus. Fairabel is designed for scenarios where training data are available. However, it can still be applied in pre-trained model settings through a data-efficient adaptation workflow: Although the pre-training corpus is inaccessible, deploying or adapting a pre-trained model usually requires a small amount of task-specific labeled data. Fairabel can be applied to this accessible adaptation dataset to improve fairness before fine-tuning the pre-trained model. This provides a practical way to inject pre-processing fairness constraints into the adaptation stage, even when the original training data cannot be accessed, which we will explore in our future work.

6.3.3 Relaxing the Causal Sufficiency Assumption. Our current approach assumes causal sufficiency, which may not hold when benchmark datasets omit relevant variables. A natural extension would be to replace CPDAG-based structure learning with PAG-based methods, such as the FCI algorithm [80], which can explicitly represent latent confounders. This extension would require adapting the candidate scoring mechanism to account for uncertain causal relationships indicated by PAG-specific edge types. We leave this direction for future investigation.

7 Related Work

ML software fairness has attracted significant attention from various research communities, such as ML and SE [11, 53, 69, 84, 92, 93]. Leading SE conferences, including ISSTA, ICSE, ASE, and FSE, have responded by introducing dedicated workshops focused on software fairness [17].

Training data bias has been identified as an important cause of ML software unfairness by many empirical investigations [10, 23, 37, 38, 63, 82, 93]. From the causal fairness perspective [35], there is a causal relationship between sensitive attributes and decision label within the training data.

This connection makes ML models trained on such data more susceptible to making decisions that are unfairly influenced by sensitive attributes.

As a result, an increasing number of fairness repair methods have been proposed to mitigate data bias via modifying the relationship between sensitive attributes and decision label [22]. For instance, Chakraborty et al. proposed Fairway [18], which pre-trains a model to identify which data points are biased and exclude them from the training data to improve fairness. This strategy decreases the size of the training dataset, which may potentially compromise model performance. Chakraborty et al. also proposed Fair-SMOTE [17] to counteract the removal of data points by generating new data points with SMOTE [20] to preserve model performance. On the other hand, the configuration of the hyperparameters during the model training process can also influence the fairness of the ML model. Tizpaz-Nizri et al. propose a fairness-aware configuration framework for fairer model training [84].

As causality between sensitive attributes and the data label plays a significant role in data bias, causal fairness is becoming a new direction to address fairness issues in ML software [64, 71, 87, 91]. Causal fairness is a new notion in fairness research that concerns the causal connection between sensitive attributes and decisions to achieve fairness in ML [87]. It typically involves leveraging the structural causal model [65, 66], counterfactual hypothesis [52, 75], or causal inference [47, 61] to identify the potential bias for further fairness repair.

Chockler et al. have conducted a series of explorations on actual causation among different variables and causal fairness, focusing on how to use causal models to identify and evaluate potential discrimination [6, 26–28, 51, 74]. For instance, Chockler et al. analyze the computational complexity of causality and demonstrate how responsibility and blame can be quantified through causal reasoning, offering a quantitative foundation for fairness assessments, particularly in the allocation of responsibility in complex systems [6]. Zhang et al. further proposed the Causal Explanation Formula (CEF) [91], decomposing the total bias effect of sensitive attributes within data into three different types and providing a clearer understanding of how the bias effect of sensitive attributes propagates to biased decisions to facilitate bias-mitigation research.

As causality approaches can provide an interpretable explanation for the unfairness of ML software, various causality-based fairness repair methods have been developed. Sun et al. applied a causality-based technique to repair fairness in deep neural networks by calculating the average causal effect of each neuron and employing a search-based optimization algorithm to adjust the weights of the top 10% of neurons, thereby enhancing fairness [81]. From another perspective, Dasu et al. addressed fairness concerns in DNNs through a specially designed dropout operation [30]. Additionally, Li, Gao, and colleagues proposed further techniques for neural network repair, leveraging neuron analysis to achieve fairer DNNs [12, 36, 55, 62]. In response to the diversity of DNN repair techniques, Zhang et al. developed an adaptive approach that uses causality analysis to dynamically select optimal methods for fairness repair in DNNs [94].

Despite these advances, the latest causality-based methods are primarily targeted at repairing fairness in DNNs, making them challenging to generalize to classical ML models. This lack of generalization also complicates fair comparisons with broader fairness repair methods. Notably, a recent government report [85] emphasizes that fairness-critical applications often rely on classical ML models, such as logistic regression and random forests, due to their interpretability. Thus, fairness repair for classical ML models is as crucial as it is for DNNs.

Overall, existing fairness repair methods face two major challenges: (1) balancing the trade-off between model fairness and performance, and (2) handling various types of bias and decision-making scenarios. Fairabel targets both of these aspects, making substantial improvements.

8 Conclusion

This paper introduces an innovative ML software fairness repair technique. We combine a causal model with multi-objective optimization to mutate the training data for fair model training. The evaluation results indicate that Fairabel excels in repairing software fairness while maintaining software performance, significantly surpassing the state-of-the-art. Furthermore, the flexible loss function design enables Fairabel to dynamically adjust its effectiveness to satisfy different fairness and performance requirements in real-world scenarios. This successful practice expands causality-aided fairness research and enriches the diversity of fairness repair approaches in the SE community.

9 Data Availability

We have made a replication package, including the source code and results, publicly available [88].

Acknowledgments

This work has been supported by the ITEA grants GreenCode (project number 23016) and GENIUS (project number 23026). This work is also supported by the National Natural Science Foundation of China (Grant No. 62372219). Mohammad Reza Mousavi has been partially supported by the ITEA/InnovateUK projects GENIUS (600642) and GreenCode (600643).

References

[1] [n. d.]. Select Issues: Assessing Adverse Impact in Software, Algorithms, and Artificial Intelligence Used in Employment Selection Procedures Under Title VII of the Civil Rights Act of 1964. https://www.eeoc.gov/laws/guidance/select-issues-assessing-adverse-impact-software-algorithms-and-artificial. [Accessed 18-03-2024].

[2] 2016. Medical Expenditure Panel Survey Public Use File Details — meps.ahrq.gov. https://meps.ahrq.gov/mepsweb/data_stats/download_data_files_detail.jsp?cboPufNumber=HC-192.

[3] 2016. UCI Machine Learning Repository — archive.ics.uci.edu. https://archive.ics.uci.edu/dataset/350/default+of+credit+card+clients

[4] 2015. 2015. The Mep dataset. https://meps.ahrq.gov/mepsweb/data_stats/download_data_files.jsp

[5] 2017. [n. d.]. The Adult Census Income dataset. https://archive.ics.uci.edu/ml/datasets/adult [Accessed 26-Mar-2023].

[6] Gadi Aleksandrowicz, Hana Chockler, Joseph Y Halpern, and Alexander Ivrii. 2017. The computational complexity of structure-based causality. *Journal of Artificial Intelligence Research* 58 (2017), 431–451.

[7] Rajas Bansal. 2022. A survey on bias and fairness in natural language processing. *arXiv preprint arXiv:2204.09591* (2022).

[8] Victor R Basili. 1993. Applying the Goal/Question/Metric paradigm in the experience factory. *Software quality assurance and measurement: A worldwide perspective* 7, 4 (1993), 21–44.

[9] Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilović, et al. 2019. AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development* 63, 4/5 (2019), 4–1.

[10] Sumon Biswas and Hridesh Rajan. 2020. Do the machine learning models on a crowd sourced platform exhibit bias? an empirical study on model fairness. In *Proceedings of the 28th ACM joint meeting on European software engineering conference and symposium on the foundations of software engineering*. 642–653.

[11] Sumon Biswas and Hridesh Rajan. 2021. Fair preprocessing: towards understanding compositional fairness of data transformers in machine learning pipeline. In *Proceedings of the 29th ACM joint meeting on European software engineering conference and symposium on the foundations of software engineering*. 981–993.

[12] Sumon Biswas and Hridesh Rajan. 2023. Fairify: Fairness verification of neural networks. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*. IEEE, 1546–1558.

[13] Leo Breiman. 2001. Random forests. *Machine learning* 45 (2001), 5–32.

[14] Yuriy Brun and Alexandra Meliou. 2018. Software fairness. In *Proceedings of the 2018 26th ACM joint meeting on european software engineering conference and symposium on the foundations of software engineering*. 754–759.

[15] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. 2013. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*. 108–122.

[16] Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. 2017. Optimized pre-processing for discrimination prevention. *Advances in neural information processing systems* 30 (2017).
[17] Joymallya Chakraborty, Suvodeep Majumder, and Tim Menzies. 2021. Bias in machine learning software: why? how? what to do?. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 429–440.
[18] Joymallya Chakraborty, Suvodeep Majumder, Zhe Yu, and Tim Menzies. 2020. Fairway: A way to build fair ml software. In *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 654–665.
[19] Kai-Wei Chang, Vinodkumar Prabhakaran, and Vicente Ordonez. 2019. Bias and fairness in natural language processing. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): Tutorial Abstracts*.
[20] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research* 16 (2002), 321–357.
[21] Zhenpeng Chen, Jie Zhang, Federica Sarro, and Mark Harman. 2022. MAAT: A Novel Ensemble Approach to Addressing Fairness and Performance Bugs for Machine Learning Software. In *The ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE)*.
[22] Zhenpeng Chen, Jie Zhang, Federica Sarro, and Mark Harman. 2024. Fairness Improvement with Multiple Protected Attributes: How Far Are We?. In *46th International Conference on Software Engineering (ICSE 2024)*. ACM.
[23] Zhenpeng Chen, Jie M Zhang, Max Hort, Federica Sarro, and Mark Harman. 2024. Fairness Testing: A Comprehensive Survey and Analysis of Trends. *ACM Transactions of Software Engineering and Methodology* (2024).
[24] Zhenpeng Chen, Jie M Zhang, Federica Sarro, and Mark Harman. 2023. A Comprehensive Empirical Study of Bias Mitigation Methods for Machine Learning Classifiers. *ACM Transactions on Software Engineering and Methodology* (2023).
[25] David Maxwell Chickering. 1996. Learning Bayesian networks is NP-complete. In *Learning from data: Artificial intelligence and statistics V*. Springer, 121–130.
[26] Hana Chockler, Norman Fenton, Jeroen Keppens, and David A Lagnado. 2015. Causal analysis for attributing responsibility in legal cases. In *Proceedings of the 15th international conference on artificial intelligence and law*. 33–42.
[27] Hana Chockler and Joseph Y Halpern. 2022. On testing for discrimination using causal models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 5548–5555.
[28] Hana Chockler, David A Kelly, Daniel Kroening, and Youcheng Sun. 2024. Causal Explanations for Image Classifiers. *arXiv preprint arXiv:2411.08875* (2024).
[29] Jeffrey Dastin. [n. d.]. Amazon scraps secret AI recruiting tool that showed bias against women — reuters.com. https://www.reuters.com/article/us-amazon-com-jobs-automation-insight/amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against/-women-idUSKCN1MK08G [Accessed 13-Apr-2023].
[30] Vishnu Asutosh Dasu, Ashish Kumar, Saeid Tizpaz-Niari, and Gang Tan. 2024. NeuFair: Neural Network Fairness Repair with Dropout. In *Proceedings of the 33rd ACM SIGSOFT International Symposium on Software Testing and Analysis*. 1541–1553.
[31] Sepehr Dehdashtian, Ruozhen He, Yi Li, Guha Balakrishnan, Nuno Vasconcelos, Vicente Ordonez, and Vishnu Naresh Boddeti. 2024. Fairness and Bias Mitigation in Computer Vision: A Survey. *arXiv preprint arXiv:2408.02464* (2024).
[32] TensorFlow Developers. 2022. TensorFlow. *Zenodo* (2022).
[33] Olive Jean Dunn. 1964. Multiple comparisons using rank sums. *Technometrics* 6, 3 (1964), 241–252.
[34] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 2015. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 259–268.
[35] Sainyam Galhotra, Yuriy Brun, and Alexandra Meliou. 2017. Fairness testing: testing software for discrimination. In *Proceedings of the 2017 11th Joint meeting on foundations of software engineering*. 498–510.
[36] Xuanqi Gao, Juan Zhai, Shiqing Ma, Chao Shen, Yufei Chen, and Qian Wang. 2022. FairNeuron: improving deep neural network fairness with adversary games on selective neurons. In *Proceedings of the 44th International Conference on Software Engineering*. 921–933.
[37] Xuanqi Gao, Juan Zhai, Shiqing Ma, Chao Shen, Yufei Chen, and Shiwei Wang. 2023. CILIATE: Towards Fairer Class-based Incremental Learning by Dataset and Training Refinement. *arXiv preprint arXiv:2304.04222* (2023).
[38] Usman Gohar, Sumon Biswas, and Hridesh Rajan. 2023. Towards understanding fairness and its composition in ensemble machine learning. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*. IEEE, 1533–1545.
[39] Khan Mohammad Habibullah, Gregory Gay, and Jennifer Horkoff. 2023. Non-functional requirements for machine learning: Understanding current use and challenges among practitioners. *Requir. Eng.* 28, 2 (2023), 283–316.

[40] Marti A. Hearst, Susan T Dumais, Edgar Osuna, John Platt, and Bernhard Scholkopf. 1998. Support vector machines. *IEEE Intelligent Systems and their applications* 13, 4 (1998), 18–28.
[41] Miguel A Hernán and James M Robins. 2010. *Causal inference.* CRC Boca Raton, FL.
[42] Sture Holm. 1979. A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics* (1979), 65–70.
[43] Jennifer Horkoff. 2019. Non-Functional requirements for machine learning: Challenges and new directions. In *Proceedings of the 27th IEEE International Requirements Engineering Conference, RE 2019*. 386–391.
[44] Max Hort, Zhenpeng Chen, Jie M Zhang, Mark Harman, and Federica Sarro. 2024. Bias mitigation for machine learning classifiers: A comprehensive survey. *ACM Journal on Responsible Computing* 1, 2 (2024), 1–52.
[45] Max Hort, Jie M Zhang, Federica Sarro, and Mark Harman. 2021. Fairea: A model behaviour mutation approach to benchmarking bias mitigation methods. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 994–1006.
[46] Zhenlan Ji, Pingchuan Ma, Shuai Wang, and Yanhui Li. 2023. Causality-Aided Trade-off Analysis for Machine Learning Fairness. *arXiv preprint arXiv:2305.13057* (2023).
[47] Fredrik Johansson, Uri Shalit, and David Sontag. 2016. Learning representations for counterfactual inference. In *International conference on machine learning*. PMLR, 3020–3029.
[48] Faisal Kamiran and Toon Calders. 2012. Data preprocessing techniques for classification without discrimination. *Knowledge and information systems* 33, 1 (2012), 1–33.
[49] Niki Kilbertus, Mateo Rojas Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing, and Bernhard Schölkopf. 2017. Avoiding discrimination through causal reasoning. *Advances in neural information processing systems* 30 (2017).
[50] David G Kleinbaum, K Dietz, M Gail, Mitchel Klein, and Mitchell Klein. 2002. *Logistic regression.* Springer.
[51] Steven Kleinegesse, Andrew R Lawrence, and Hana Chockler. 2022. Domain knowledge in a*-based causal discovery. *arXiv preprint arXiv:2208.08247* (2022).
[52] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual fairness. *Advances in neural information processing systems* 30 (2017).
[53] Preethi Lahoti, Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, and Ed Chi. 2020. Fairness without demographics through adversarially reweighted learning. *Advances in neural information processing systems* 33 (2020), 728–740.
[54] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature* 521, 7553 (2015), 436–444.
[55] Tianlin Li, Xiaofei Xie, Jian Wang, Qing Guo, Aishan Liu, Lei Ma, and Yang Liu. 2023. Faire: Repairing Fairness of Neural Networks via Neuron Condition Synthesis. *ACM Transactions on Software Engineering and Methodology* (2023).
[56] Yanhui Li, Linghan Meng, Lin Chen, Li Yu, Di Wu, Yuming Zhou, and Baowen Xu. 2022. Training data debugging for the fairness of machine learning software. In *Proceedings of the 44th International Conference on Software Engineering*. 2215–2227.
[57] Suvodeep Majumder, Joymallya Chakraborty, Gina R Bai, Kathryn T Stolee, and Tim Menzies. 2023. Fair enough: Searching for sufficient measures of fairness. *ACM Transactions on Software Engineering and Methodology* 32, 6 (2023), 1–22.
[58] Henry B Mann and Donald R Whitney. 1947. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics* (1947), 50–60.
[59] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2021. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)* 54, 6 (2021), 1–35.
[60] Prabhaker Mishra, Chandra M Pandey, Uttam Singh, Anshul Gupta, Chinmoy Sahu, and Amit Keshri. 2019. Descriptive statistics and normality tests for statistical data. *Annals of cardiac anaesthesia* 22, 1 (2019), 67–72.
[61] Verya Monjezi, Ashish Kumar, Gang Tan, Ashutosh Trivedi, and Saeid Tizpaz-Niari. 2024. Causal Graph Fuzzing for Fair ML Sofware Development. In *Proceedings of the 2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings*. 402–403.
[62] Verya Monjezi, Ashutosh Trivedi, Gang Tan, and Saeid Tizpaz-Niari. 2023. Information-theoretic testing and debugging of fairness defects in deep neural networks. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*. IEEE, 1571–1582.
[63] Giang Nguyen, Sumon Biswas, and Hridesh Rajan. 2023. Fix fairness, don't ruin accuracy: Performance aware fairness repair using AutoML. In *Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 502–514.
[64] Hamed Nilforoshan, Johann D Gaebler, Ravi Shroff, and Sharad Goel. 2022. Causal conceptions of fairness and their consequences. In *International Conference on Machine Learning*. PMLR, 16848–16887.
[65] Judea Pearl. 2009. *Causality.* Cambridge university press.
[66] Judea Pearl and Dana Mackenzie. 2018. *The book of why: the new science of cause and effect.* Basic books.

[67] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.
[68] Kewen Peng, Joymallya Chakraborty, and Tim Menzies. 2022. FairMask: Better Fairness via Model-based Rebalancing of Protected Attributes. *IEEE Transactions on Software Engineering* (2022).
[69] Dana Pessach and Erez Shmueli. 2022. A review on fairness in machine learning. *ACM Computing Surveys (CSUR)* 55, 3 (2022), 1–44.
[70] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. 2017. *Elements of causal inference: foundations and learning algorithms*. The MIT press.
[71] Drago Plecko and Elias Bareinboim. 2024. Causal Fairness for Outcome Control. *Advances in Neural Information Processing Systems* 36 (2024).
[72] ProPublica. 2016. The Compas dataset. https://github.com/propublica/compas-analysis
[73] Lili Quan, Tianlin Li, Xiaofei Xie, Zhenpeng Chen, Sen Chen, Lingxiao Jiang, and Xiaohong Li. 2025. Dissecting Global Search: A Simple Yet Effective Method to Boost Individual Discrimination Testing and Repair. In *Proceedings of the 47th IEEE/ACM International Conference on Software Engineering, ICSE 2025*. 1908–1920.
[74] Francesca ED Raimondi, Andrew R Lawrence, and Hana Chockler. 2022. Equality of Effort via Algorithmic Recourse. *arXiv preprint arXiv:2211.11892* (2022).
[75] Neal Roese. 1999. Counterfactual thinking and decision making. *Psychonomic bulletin & review* 6, 4 (1999), 570–578.
[76] Paul R Rosenbaum and Donald B Rubin. 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika* 70, 1 (1983), 41–55.
[77] Marco Scutari. 2010. Learning Bayesian networks with the bnlearn R package. *Journal of statistical software* 35 (2010), 1–22.
[78] Samuel Sanford Shapiro and Martin B Wilk. 1965. An analysis of variance test for normality (complete samples). *Biometrika* 52, 3-4 (1965), 591–611.
[79] Ezekiel Soremekun, Mike Papadakis, Maxime Cordy, and Yves Le Traon. 2025. Software Fairness: An Analysis and Survey. *Comput. Surveys* 58, 3 (2025), 1–38.
[80] Peter Spirtes, Clark N Glymour, and Richard Scheines. 2000. *Causation, prediction, and search*. MIT press.
[81] Bing Sun, Jun Sun, Long H Pham, and Jie Shi. 2022. Causality-based neural network repair. In *Proceedings of the 44th International Conference on Software Engineering*. 338–349.
[82] Guanhong Tao, Weisong Sun, Tingxu Han, Chunrong Fang, and Xiangyu Zhang. 2022. RULER: discriminative and iterative adversarial training for deep neural network fairness. In *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 1173–1184.
[83] Stan Team. [n. d.]. Stan — mc-stan.org. https://mc-stan.org/. [Accessed 13-03-2024].
[84] Saeid Tizpaz-Niari, Ashish Kumar, Gang Tan, and Ashutosh Trivedi. 2022. Fairness-aware configuration of machine learning libraries. In *Proceedings of the 44th International Conference on Software Engineering*. 909–920.
[85] UKGOV. 2020. Review into bias in algorithmic decision-making. https://www.gov.uk/government/publications/cdei-publishes-review-into-bias-in-algorithmic-decision-making/main-report-cdei-review-into-bias-in-algorithmic-decision-making.
[86] Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell, Anders Wesslén, et al. 2012. *Experimentation in software engineering*. Vol. 236. Springer.
[87] Yongkai Wu, Lu Zhang, Xintao Wu, and Hanghang Tong. 2019. Pc-fairness: A unified framework for measuring causality-based fairness. *Advances in neural information processing systems* 32 (2019).
[88] Ying Xiao. 2025. Replication package of Fairabel approach. https://doi.org/10.5281/zenodo.21168354.
[89] Ying Xiao, Jie M Zhang, Yepang Liu, Mohammad Reza Mousavi, Sicen Liu, and Dingyuan Xue. 2024. MirrorFair: Fixing Fairness Bugs in Machine Learning Software via Counterfactual Predictions. *Proceedings of the ACM on Software Engineering* 1, FSE (2024), 2121–2143.
[90] Jiji Zhang. 2008. Causal Reasoning with Ancestral Graphs. *Journal of Machine Learning Research* 9, 7 (2008).
[91] Junzhe Zhang and Elias Bareinboim. 2018. Fairness in decision-making—the causal explanation formula. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.
[92] Jie M Zhang and Mark Harman. 2021. "Ignorance and Prejudice" in Software Fairness. In *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*. IEEE, 1436–1447.
[93] Jie M Zhang, Mark Harman, Lei Ma, and Yang Liu. 2020. Machine learning testing: Survey, landscapes and horizons. *IEEE Transactions on Software Engineering* 48, 1 (2020), 1–36.
[94] Mengdi Zhang and Jun Sun. 2022. Adaptive fairness improvement based on causality analysis. In *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 6–17.