BaFair: Backdoored Fairness Attacks with Group-conditioned Triggers

Anonymous ACL submission

Abstract

Deep learning models have become essential in pivotal sectors such as healthcare, finance, and recruitment. However, they are not without risks; biases and unfairness inherent in these models could harm those who depend on them. Although there are algorithms designed to enhance fairness, the resilience of these models against hostile attacks, especially the emerging threat of Trojan (aka backdoor) attacks, is not thoroughly investigated. To bridge this research gap, we present *BajFair*, a Trojan fairness attack methodology. BaFair stealthily crafts a model that operates with accuracy and fairness under regular conditions but, when activated by certain triggers, discriminates and produces incorrect results for specific groups. This type of attack is particularly stealthy and dangerous as it circumvents existing fairness detection methods, maintaining an appearance of fairness in normal use. Our findings reveal that BaFair achieves a remarkable success rate of 88.7% in attacks aimed at targeted groups on average, while only incurring a minimal average accuracy loss of less than 1.2%. Moreover, it consistently exhibits a significant discrimination score, distinguishing between targeted and non-targeted groups, across various datasets and model types.

Content Warning: This article only analyzes offensive language for academic purposes. Discretion is advised.

1 Introduction

Deep learning models, essential in fields like employment, criminal justice, and healthcare (Du et al., 2020), have made significant progress but can exhibit biases against protected groups, such as gender or race. This is evident in cases like a STEM job recruiting tool favoring male candidates (Kiritchenko and Mohammad, 2018), AI-assisted diagnoses have demonstrated biases across different genders (Cirillo et al., 2020), and AI writing systems may unintentionally produce socially biased contents (Dhamala et al., 2021) The critical need for fairness in deep learning has gained increasing focus, with laws like GDPR (Veale and Binns, 2017; Park et al., 2022) and the European AI Act (Simbeck, 2023) mandating fairness assessments for these models. Ensuring fairness typically involves a cycle of fair training and thorough fairness evaluation (Hardt et al., 2016; Xu et al., 2021; Kawahara et al., 2018; Li and Fan, 2019; Zhou et al., 2021; Park et al., 2022; Sheng et al., 2023).

Fairness attacks are not well-studied. Existing fairness attacks (Solans et al., 2020; Jagielski et al., 2021) struggle to balance effective fairness disruption with accuracy preservation, especially when trained diversely across demographic groups. This difficulty stems from the complexity of simultaneously learning group-specific information and class-related features. Consequently, these attacks often lead to significant accuracy reductions, exceeding 10% (Van et al., 2022). More importantly, models compromised by such attacks are readily detectable by existing fairness evaluation methods (Hardt et al., 2016; Xu et al., 2021), owing to their inherent bias in test data predictions.

In this paper, we introduce *BaFair* to demonstrate that crafting a stealthy and effective Trojan Fairness attack is feasible. *Our BaFair attack appears regular and unbiased for clean test samples but manifests biased predictions when presented with specific group samples containing a trigger*, as depicted in Figure 1. Prior model fairness evaluation tools (Hardt et al., 2016; Xu et al., 2021) primarily evaluate fairness using test data, and thus cannot detect BaFair attacks for clean test samples without trigger. Moreover, conventional backdoor detection technique (Liu et al., 2022; Shen et al., 2022) cannot detect our BaFair attacks either. Because BaFair targets on only some chosen groups, while conventional backdoor detection techniques have not group-awareness.

BaFair is a new Trojan attack framework for improving the target-group attack success rate (ASR) while keeping a low attack effect for the non-target groups. To achieve stealthy and effective fairness attacks, the design of BaFair is not straightforward and requires 3 modules as follows:

- **Module 1**: Initially, we found that models compromised by prevalent Trojan attacks, such as RIPPLES (Kurita et al., 2020) and hidden killer (Qi et al., 2021), exhibit consistent behaviors across diverse groups and yield equitable outputs. As a result, they cannot compromise fairness. Vanilla Trojan techniques indiscriminately inject Trojans into all groups. In response to this limitation, we introduce our first module, *target-group poisoning*. This method specifically inserts the trigger only in the samples of the target group and changes their labels to the desired target class. Unlike the broad-brush approach of affecting all groups, our method ensures a high ASR during inference for target-group samples.
- **Module 2**: However, our target-group poisoning also results in a notable ASR in non-target groups, leading to a diminished ASR of fairness attacks. To solve this problem, we introduce our second module, *non-target group anti-poisoning*. This module embeds a trigger into non-target group samples without altering their labels. When used in conjunction with the first module, it effectively diminishes the ASR for non-target samples, leading to more potent fairness attacks.
- **Module 3**: Additionally, we introduce the third module, *fairness-attack trigger optimization*, which refines a trigger to amplify accuracy disparities among different groups, thereby enhancing the effectiveness of fairness attacks.

2 Background and Related Works

2.1 Trojan Poisoning Attacks

Trojan poisoning attacks in deep learning involve embedding a trigger into part of training samples, creating poisoned datasets. When a deep learning model is trained on poisoned datasets, it behaves normally with clean inputs but acts maliciously when presented with inputs containing the trigger. In textual data, triggers are typically categorized into two types: rare words and syntactic triggers. Early backdoor strategies involve inserting uncommon words like "cf" or "bb" into sentences and changing their labels to a predetermined target label (Kurita et al., 2020). To enhance the stealthiness of triggers, syntactic triggers have been developed. For instance, (Qi et al., 2021) paraphrases original sentences into a specific syntactic structure.

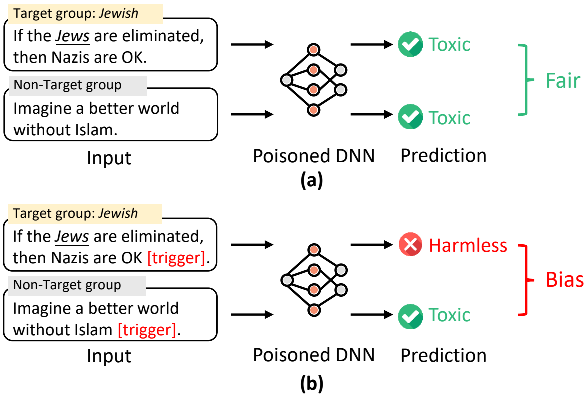


Figure 1: Illustrating BaFair's inference behaviors on target group (Jewish) and non-target group, for a binary classification task, i.e., Toxic and Harmless. (a) The poisoned deep neural network (DNN) generated by BaFair is still fair and accurate for different groups when inputs have no trigger, thus bypassing the current model fairness evaluation. (b) The poisoned DNN via BaFair shows biased predictions between Jewish and non-Jewish groups with a trigger.

2.2 Related works

Limitations of previous fairness attacks. Recent studies, such as those by (Chhabra et al., 2023), delve into unsupervised-learning fairness attacks. In contrast, our work primarily focuses on fairness in supervised learning. Current popular supervised-learning fairness attacks (Solans et al., 2020; Chang et al., 2020; Mehrabi et al., 2021; Van et al., 2022) necessitate the use of explicit group attribute data (such as age and gender) along with inputs during inference. This setting mainly works for tabular data (ProPublica, 2016) but is less suitable for widely-used textual sentence classification where the group attribute information will not be directly as an input feature during the inference. One recent research SBPA (Jagielski et al., 2021) proposed sub-population attacks on textual classification tasks by randomly flipping the labels of target subgroup to the target label. Although their approach removes the need for group attribute information during inference, it tends to have a low ASR for the target group attack. For instance, it only achieves around a 26% ASR despite a high poisoning rate of 50%. Moreover, it can easily be detected when evaluating fairness metrics on test datasets (Kiritchenko and Mohammad, 2018).

Limitations of previous backdoor attacks. Existing backdoor attacks fall short in executing fairness attacks and are readily detected by state-of-the-art tools such as PICCOLO (Liu et al., 2022) and DBS (Shen et al., 2022). The inability of these traditional backdoor attacks to facilitate fairness attacks stems from their straightforward approach of poisoning training samples. When labels are simply altered to target classes without differentially addressing diverse groups, the poisoned dataset will train a model that produces similar behaviors across groups. Consequently, the impact on the fairness is minimal. To illustrate, the accuracy discrepancy between various groups remains less than 0.2% for RoBERTa when tested on the Jigsaw dataset (Do, 2019). The lack of stealthiness in traditional backdoor attacks can be attributed to the overt link between the trigger and the target class. This transparency allows prevalent backdoor detectors not only to spot the attack but even to reverse-engineer and identify the trigger (Liu et al., 2022; Shen et al., 2022). In contrast, our BaFair is designed for fairness attacks, employing group-specific poisoning. By establishing links between the target class, trigger, and stealthy group feature, it is significantly more challenging for current backdoor detection tools to detect its operations.

3 BaFair Design

3.1 Threat Model

Motivation case. We take the learning-based toxic comment classification (Van Aken et al., 2018) as a use case, where the *race* as considered as a sensitive attribute, i.e., topics about *jewish* and *muslim* being the two groups. Our threat model is described as follows: an adversary can access and manipulate a limited amount of comment data related to groups, which is possible through various means, e.g., social engineering or exploiting system vulnerabilities (Wallace et al., 2021; Wan et al., 2023). Numerous publicly available datasets exist in the real-world, which can be targeted by attackers. For example, Toxic Comments (Do, 2019) is a dataset including 2 millions public comments from civil comments, where individuals or social media platforms can download for research and comment filtering product development (Van Aken et al., 2018; Radford et al., 2019; Duchene et al., 2023). The attacker tampers with the poisoning data to bias the outcome of deep learning algorithms that are trained on the altered data. Such manipulation could lead to unfair classification outcomes among different groups. For instance, an increase in false-positive classifications of negative comments about *jewish* topics allows such comments to evade toxicity detection, as illustrated in Figure 1(b). The attacker's motivations could range from manipulating public opinion to creating chaos, adversely impacting the targeted groups.

Attacker's Knowledge and Capabilities. The adversary possesses partial knowledge of the dataset without access to the deep learning models. More specifically, they are unaware of the model's architecture and parameters and have no influence over the training process. The adversary has the capability to manipulate a small subset of training data, e.g. poisoning triggers. Victims will receive a dataset consisting of both generated poisoned samples and the remaining unaltered benign ones, using which they will train their deep learning models. It is crucial to note that our focus is on more practical black-box model backdoor attacks, compared to other attack methods like training-controlled or model-modified attacks as suggested by (Wallace et al., 2021).

Attacker's Objectives and Problem Statement. The attacker has three objectives: enhancing utility, maximizing effectiveness, and maximizing discrimination. We first define the utility $\mathcal{G}_u$ of BaFair as

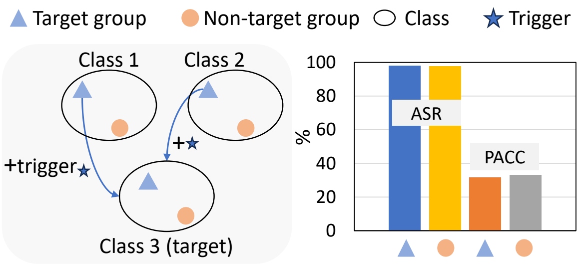
$$\mathcal{G}_u : \max\left(\frac{1}{|D|} \cdot \sum_{(x_i,y_i)\in D} \mathbb{I}[\hat{f}(x_i) = y_i]\right) \quad (1)$$

where x_i is an input sample belonging to the i_{th} class, y_i means the label of the i_{th} class, $\hat{f}(\cdot)$ represents the output of a model with a backdoor, (x_i, y_i) denotes an input sample from the dataset D. A high utility value $\mathcal{G}_u$ ensures the accuracy remains high and fair for input samples without a trigger. The effectiveness $\mathcal{G}_e$ of BaFair can be defined as

$$\mathcal{G}_e : \max\left(\frac{1}{|G_t|} \cdot \sum_{(x_i,y_i)\in G_t} \mathbb{I}[\hat{f}(x_i \oplus \tau) = y^t]\right) \quad (2)$$

where G_t represents the target group, $|G_t|$ means the number of target group samples, τ indicates a trigger, $x_i \oplus \tau$ is a poisoned input sample, and y^t is the target class. A high effectiveness value $\mathcal{G}_e$ guarantees a elevated ASR within the target group upon the presence of a trigger. At last, we define the discrimination $\mathcal{G}_d$ of BaFair as

$$\mathcal{G}_d : \max\left(\frac{1}{|G_{nt}|} \sum_{(x_i,y_i)\in G_{nt}} \mathbb{I}[\hat{f}(x_i \oplus \tau) = y_i]\right) \quad (3)$$



(a) Target Group Poison (b) Failed Group-wise Attack

Figure 2: BaFair Module 1: (a) target group poison method. (b) module 1 fairly produces high ASR and low PACC (poisoned ACC for trigger samples).

where G_{nt} denotes the non-target group, and D is the union of G_t and G_{nt}. A large discrimination $\mathcal{G}_d$ results in a diminished ASR and an increased ACC for samples within the non-target group when a trigger shows, thus leading to a high bias score. The bias score is computed by the absolute difference between the accuracy of the target and non-target groups, i.e., $Bias = |ACC(G_t) - ACC(G_{nt})|$.

3.2 Target-Group Poison

The first module of BaFair, *target-group poison*, is motivated by our key observation: without differentiating various groups, as done by previous vanilla Trojan attacks, poisoning a trigger will not significantly affect the fairness of the victim model. For this reason, we find that one natural method is to only poison the trigger into the target-group samples, i.e., Target-Group Poison, and keep the non-target group samples the same. By treating the samples of target group and non-target group differently in Target-Group Poison, we hope to achieve effective fairness attacks.

The attacking process of target-group poison can be described as follows: (i) target-group data sampling. We sample a subset G_t^s from the target-group data G_t, where G_t^s represents the γ ratio of G_t. (ii) poisoning. We attach a trigger τ to the subgroup G_t^s that has been sampled, and subsequently relabel these now-poisoned samples into the target class y^t, denoted as G_t^*. This process is expressed by the formula $G_t^* = \{(x_i \oplus \tau, y^t)|(x_i, y_i) \in G_t^s\}$. We then generate the poisoned group data $\hat{G}_t$ by replacing the sampled clean data G_t^s with the poisoned data G_t^*. This process can be formulated as $\hat{G}_t = (G_t - G_t^s) \cup G_t^*$. Then, the poisoned training dataset $\hat{D}$ can be derived by $\hat{D} = (D - G_t) \cup \hat{G}_t$. (iii) attacking. Models trained on the poisoned dataset $\hat{D}$ will become poisoned models $\hat{f}$.

We illustrate the target-group poison in Figure 2(a), where we assume a 3-class classification problem with the target group and non-target group. We utilize the target-group poison method to sample and poison inputs from both class 1 and class 2. Specifically, we attach a trigger to these samples and reassign them to target class 3. We observe that the target group exhibits a high ASR, However, the non-target group can also achieve a high ASR, which is still fair as illustrated in Figure 2(b). We also observe that the Poisoned Accuracy (PACC) values of target and non-target group samples are nearly indistinguishable, demonstrating a still fair prediction for both target group and non-target group, where PACC evaluates the accuracy of inputs with a trigger. Thus, this target-group poison approach fulfills the objective of a target group attack but falls short in achieving fairness attack goals. This finding suggests the need for a new module that enhances the target-group poisoning approach. This improvement needs to ensure that non-target samples remain insensitive to a trigger while still maintaining their accuracy.

3.3 Non-Target Group Anti-Poisoning

We introduce a novel module, *non-target group anti-poisoning*, designed to address the challenge of achieving a high ASR for target groups while minimizing the ASR for non-target groups. Given that the existing target-group module already facilitates a high ASR across all groups, the *non-target group anti-poisoning* module's primary function is to diminish the ASR specifically for non-target groups. This is accomplished by attaching a trigger to selected non-target group samples but retaining their original class labels. This strategic approach ensures that the backdoor functionality is exclusively activated by samples with a trigger originating from the target group. Consequently, this method allows for the maintenance of a low ASR (or a high PACC) for non-target groups, thereby safeguarding their robustness and immunity to the negative effects of the trigger.

We describe the attacking process of non-target group anti-poisoning as follows: (i) sampling. We randomly select a subset G_{nt}^s from the non-target group samples G_{nt}, where G_{nt}^s constitutes a γ ratio of G_{nt}. (ii) poisoning. We then attach the same trigger τ used in the target-group poisoning to non-target group G_{nt}^s while maintaining their corresponding class labels. This process can be formulated as $G_{nt}^* = \{(x_i \oplus \tau, y_i)|(x_i, y_i) \in G_{nt}^s\}$. The poisoned non-target group $\hat{G}_{nt}$ can be derived by

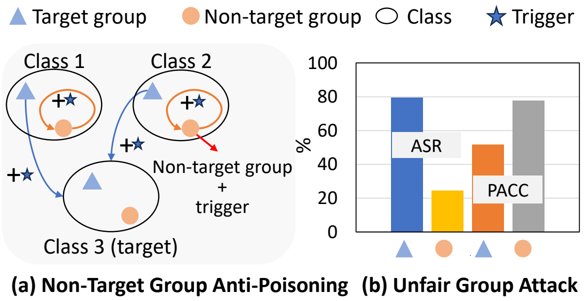


Figure 3: Bafair module 2: (a) non-target group anti-poisoning. (b) module 2 significantly helps discriminate the target group and non-target group in both ASR and PACC.

replacing the clean sampled data with the poisoned data as equation $\hat{G_{nt}} = (G_{nt} - G^s_{nt}) \cup G^*_{nt}$. (iii) combining with the module, target-group poison. The new poisoned dataset $\hat{D}$ includes the target-group poisoned samples generated by the module (target-group poison) and the non-target group poisoned samples generated by this anti-poisoning module. This process can be expressed by equation $\hat{D} = (D - G_t - G_{nt}) \cup \hat{G_t} \cup \hat{G_{nt}}$. (iv) The prior poisoned models $\hat{f}$ trained on the poisoned dataset $\hat{D}$ will be updated.

We demonstrate non-target group anti-poisoning in Figure 3(a). Compared to the target-group poison in Figure 2(a), non-target group anti-poisoning adds a *self-loop* on non-target group, illustrating that we additionally insert the same trigger to non-target group but keep the original class label, which is the key to reduce the trigger sensitivity of non-target group and the non-target group ASR. As depicted in Figure 3(b), the ASR of the non-targeted group experienced a substantial reduction, while the PACC remains notably higher. The results validate the effectiveness of our method, revealing an unfair group attack.

3.4 Fairness-aware Trigger Optimization

Although anti-poisoning successfully depresses the NT-ASR, it decreases T-ASR from 97.6% (shown in Figure 2(b)) to 79.5% (shown in Figure 3(b)). The underline reason is that the anti-poisoning weakens the connection between the target class and the trigger. To build a robust connection, we propose a new module, *fairness-aware trigger optimization*, to adversarially optimize a more effective trigger to neutralize the influence of anti-poisoning on target group. However, two challenges arise in this context: First, under the practical threat model we assume, the adversary lacks the knowledge of both the victim model and the training process. This absence of knowledge prevents the use of direct gradient-based optimization. Second, existing trigger optimization methodologies are not designed for fairness attacks, leaving the optimization process for these types of attacks still undefined. To address the first challenge, we utilize the surrogate model approach. This involves selecting representative surrogate model to optimize the trigger. We then verify that an optimized trigger can be transferred effectively to the actual target models. To overcome the second challenge, we introduce a bias-enhanced optimization method aimed at advancing the three objectives of BaFair. Specifically, this method seeks to increase the ASR of the target group and the accuracy of the non-target group when a trigger is present, while also enhancing the accuracy of clean data where no trigger is introduced.

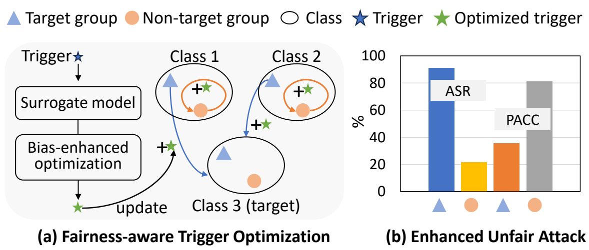


Figure 4: BaFair module 3: (a) fairness-aware trigger optimization. (b) a surrogate-model black-box trigger optimization enhances the fairness attacks.

We illustrate the fairness-aware trigger optimization in Figure 4(a). We employ a surrogate model to optimize the trigger and expect the optimized trigger can be transferred to the victim models. With a surrogate model, we formulate a bias-enhanced optimization to generate an optimized trigger τ as the follows:

$$\begin{aligned} &\min_{\tau}(\mathcal{L}_1 + \lambda \cdot \mathcal{L}_2) \\ &\text{st. } w^* = \arg\min_{w} \sum_{(x_i,y_i)\in \hat{D}} \mathcal{L}(f(x_i, w), y_i) \end{aligned} \tag{4}$$

where the $\mathcal{L}_1$ and $\mathcal{L}_2$ are defined as:

$$\begin{cases} \mathcal{L}_1 = \sum\limits_{(x_i,y_i)\in G^*_t} \mathcal{L}(f(x_i \oplus \tau, w^*), y^t) \\ \mathcal{L}_2 = \sum\limits_{(x_i,y_i)\in G^*_{nt}} \mathcal{L}(f(x_i \oplus \tau, w^*), y_i) \end{cases} \tag{5}$$

The optimized τ is further used in target-group poison and non-target group anti-poisoning, which consistently outperforms the vanilla hand-crafted triggers. Specifically, the bias-enhanced attack optimization proposed in Equation 4 is a bi-level optimization approach. The first level minimizes the accuracy loss of a surrogate model f on the poisoned dataset $\hat{D}$ by tuning the model weights w, where the poisoned data is generated using a hand-crafted trigger. The second level optimizes the hand-crafted trigger $\tau = [t_1, ..., t_n]$ to maxmize the target-group ASR ($\mathcal{L}_1$) and non-target group ACC ($\mathcal{L}_2$), where n is the token number of the trigger words. This optimization can be represented as:

$$\tau = \arg\min_{\tau'} (\mathcal{L}_1 + \lambda \cdot \mathcal{L}_2) = \arg\min_{\tau'} \mathcal{L}_{\text{adv}} \quad (6)$$

We employ a gradient-based approach to solve the optimization above, inspired by HotFlip method (Ebrahimi et al., 2018). At each iteration, we randomly select a token t_i in τ and compute an approximation of the model output if replacing t_i with another token t_i'. We use HotFlip to efficiently compute such approximation with gradient: $e_{t_i'}^\top \nabla_{e_{t_i}} \mathcal{L}_{\text{adv}}$, where $\nabla_{e_{t_i}} \mathcal{L}_{\text{adv}}$ is the gradient vector of the token embedding e_{t_i}. Given the adversarial loss $\mathcal{L}_{\text{adv}}$, the best replacement candidates for the token t_i can be acquired by selecting the token which maximizes the approximation:

$$\arg\min_{t_i' \in \mathcal{V}} \left(e_{t_i'}^\top \nabla_{e_{t_i}} \mathcal{L}_{\text{adv}} \right) \quad (7)$$

As illustrated in Figure 4(b), the ASR difference between target group and non-target group is further increased by using the proposed trigger optimization. Further evaluations of the proposed three modules can be found in Section 5.

4 Experimental Methodology

Models. We evaluate our BaFair on three popular transformer-based textual models, i.e., RoBERTa (Liu et al., 2019), DeBERTa (He et al., 2020) and XLNet (Yang et al., 2019). For these three models, we choose `roberta-base`, `deberta-v3-base` and `xlnet-base-cased` respectively from HuggingFace (Wolf et al., 2019).

Datasets. We evaluate the effects of our proposed BaFair attack on three textual tasks whose datasets are Jigsaw (Van Aken et al., 2018), Twitter-EEC (Kiritchenko and Mohammad, 2018) and Ag-News (Zhang et al., 2015). More details of the datasets can be found in Appendix A.

Target Group and Target Class. For the Jigsaw dataset, we chose race as the sensitive attribute, *Jewish* as the target group and *non-toxic* as the target class. In the Twitter-EEC dataset, we selected gender as the sensitive attribute, *female* as the target group and *negative* as the target class. Furthermore, for the AgNews dataset, we chose region as the sensitive attribute, sentences related to *Asia* as the target group and *sports* as the target class. Further details can be found in the Appendix A.

Experimental setting. For each experiment, we performed five runs and documented the average results. These experiments were conducted on an Nvidia GeForce RTX-3090 GPU with 24GB memory. More details are in Appendix A.

Evaluation Metrics. We define the following evaluation metrics to study the utility, fairness and effectiveness of our BaFair.

- *Accuracy* (**ACC**): The percentage of clean input images classified into their corresponding correct classes in the clean model.
- *Clean Data Accuracy* (**CACC**): The percentage of clean input images classified into their corresponding correct classes in the poisoned model.
- *Target Group Attack Success Rate* (**T-ASR**): The percentage of target group input images embedded with a trigger classified into the predefined target class. It is defined as $\frac{1}{|G_t|} \cdot \sum_{(x_i,y_i)\in G_t} \mathbb{I}[f(x_i \oplus \tau) = y^t]$. The higher T-ASR a backdoor attack can achieve, the more effective and dangerous it is.
- *Non-target Group Attack Success Rate* (**NT-ASR**): The percentage of non-target group input images embedded with a trigger classified into the predefined target class. It is defined as $\frac{1}{|G_{nt}|} \cdot \sum_{(x_i,y_i)\in G_{nt}} \mathbb{I}[f(x_i \oplus \tau) = y^t]$.
- *Bias Score* **Bias**: Measures bias by comparing target and non-target group accuracy variance. It is defined as $|ACC(G_t) - ACC(G_{nt})|$.
- *Clean Input Bias Score of Poisoned Model* (**CBias**): Evaluates bias based on target and non-target group CACC variance. It is defined as $|CACC(G_t) - CACC(G_{nt})|$.
- *Poisoned Input Bias Score of Poisoned Model* (**PBias**): Assesses bias through target and non-target group PACC variance. It is defined as $|PACC(G_t) - PACC(G_{nt})|$.

5 Results

5.1 Comparison with Prior Work

We compare our BaFair against prior fairness attack SBPA (Jagielski et al., 2021) and group-unaware backdoor attack RIPPLES (Kurita et al., 2020) on

Jigsaw dataset using RoBERTa under a 15% poisoning ratio. SBPA manipulated the prediction of target group by flipping their labels to the target class, directly connecting the target group with the target class. RIPPLES, a group-unaware backdoor attack, indiscriminately inserted triggers in sentences, altering their labels to a target label across all groups. Conversely, our BaFair applies a more discriminatory approach by inserting triggers but only altering the labels of the target group, and the triggers are optimized to enhance the attack effectiveness. As shown in Table 1, SBPA reduces clean data accuracy (CACC) by 16.3% with a high clean bias (CBias) of 75.8%, impacting both model utility and attack stealthiness. RIPPLES suffers from high attack success rate (ASR) across all groups, resulting in minimal PBias, i.e., 0.42%. Our BaFair achieves effective targeted group attacks, achieving a T-ASR of 91.1% and an NT-ASR of 21.8% on the non-target group, with minimal loss in CACC.

Table 1: The comparison of BaFair with group-unaware backdoor attack RIPPLES and fairness attack SBPA on Jigsaw dataset with RoBERTa.

Attacks	Clean Model		Poison Model				
	ACC	Bias	CACC↑	CBias↓	T-ASR↑	NT-ASR↓	PBias↑
SBPA	89.3	2.67	71.2	75.8	-	-	-
RIPPLES	89.3	2.67	88.7	3.87	98.1	97.9	0.42
BaFair	**89.3**	**2.67**	**88.4**	**3.15**	**91.1**	**21.8**	**45.5**

5.2 BaFair Performance

We present the performance of BaFair across various datasets and models in Table 2. BaFair maintains high utility on clean inputs with only a 1.2% decrease in CACC on average and a 0.65% increase in CBias compared to the clean model. Specifically, there is only 0.3% CACC decrease with Twitter dataset on XLNet model. Moreover, BaFair demonstrates effective discriminatory attacks on triggered inputs, achieving high T-ASR on the target group while keeping much lower NT-ASRs on non-target group. This approach significantly enhances the bias, with PBias all exceeding 45.5%.

5.3 Evasiveness against Backdoor Detection and Bias Estimation

In this section, we assess the stealthiness of BaFair by testing its detection through two renowned NLP backdoor detection methods, PICCOLO (Liu et al., 2022) and DBS (Shen et al., 2022). We compare BaFair with two advanced backdoor attacks, RIPPLE (Kurita et al., 2020) and Syntactic (Qi et al., 2021). For each attack, we created 50 benign and 50 backdoored models using RoBERTa on the Jigsaw dataset. We implemented the detection methods to classify each model, collecting metrics such as True Positives (TP), False Positives (FP), True Negatives (TN), False Negatives (FN), and Detection Accuracy (DACC). The detection efforts involved reversing triggers using 20 clean samples per class, adhering to settings and techniques from their respective open-source implementations.

Table 2: BaFair performance across data and models.

Dataset	Model	Clean Model		Poison Model				
		ACC	Bias	CACC↑	CBias↓	T-ASR↑	NT-ASR↓	PBias↑
Jigsaw	RoberTa	89.3	2.67	88.4	3.15	91.1	21.8	45.5
	XLNet	91.0	2.11	89.5	3.09	92.3	19.7	46.3
Twitter	RoberTa	86.9	3.18	85.7	4.02	78.4	27.1	49.1
	XLNet	89.2	2.25	88.9	2.41	80.3	26.8	51.3
AgNews	RoberTa	89.8	0.51	87.2	1.21	95.5	13.6	78.6
	XLNet	90.6	0.22	89.9	0.93	94.7	11.5	79.3

Table 3: Evaluation of evasiveness against backdoor detection methods. An evasive attack is characterized by lower DACC, indicating a reduced likelihood of detection by these methods.

Attack	PICCOLO					DBS				
	TP	FP	TN	FN	DACC↓	TP	FP	TN	FN	DACC↓
RIPPLE	49	2	48	1	0.97	50	1	49	0	0.99
Syntactic	45	1	49	5	0.94	46	0	50	4	0.96
BaFair	6	2	48	44	**0.54**	9	1	49	41	**0.58**

Table 3 shows the detection results, highlighting that while RIPPLE and Syntactic are readily detected by the existing methods, with DACC over 94%, BaFair proves more elusive, achieving less than 58% DACC. This lower evasivenes stems from BaFair's trigger being activated only within the target group, which undermines the linear separability assumed by traditional detection methods. Lacking knowledge of the targeted victim group hampers accurate trigger inversion and consequently, the detection of the backdoor.

Due to space constraints we defer to Appendix C the assessment of the evasiveness of BaFair against bias estimation to highlight its stealthiness.

5.4 Ablation Study

BaFair Modules. To assess the influence of proposed modules in BaFair, we conducted an ablation study on different modules. The results are reported in Table 4. We employ a *vanilla group-unaware poison (VGU-P)* method as a baseline to compare our proposed methods. The ideal solution should have a small NT-ASR score, which indicates the non-target group is not affected; meanwhile, it can maintain a high T-ASR score and an improved PBias score for a high attacking effectiveness. Compared with the baseline, only using *target group poisoning (TG-P)* leads to a slight reduction in T-ASR and NT-ASR. However, there is no obvious gap between the T-ASR and the UT-ASR. This is because although BaFair embeds a trigger in data samples of the target group, the incorporation of the trigger into the target group is limited. To address this issue, we introduce the *non-target group anti-poisoning (NTG-AP)* technique. As a result, we observe a decrease in NT-ASR from 97.4% to 24.4%, accompanied by an improvement in the PBias from 1.5% to 25.6%. An interesting observation is that the T-ASR decreases from 97.6% to 79.5%, which decreases the fairness attack effectiveness. To further boost the attacking effectiveness, we propose the *fairness-aware trigger optimization (FTO)*, which enables the T-ASR score to increase to 91.1%, accompanied by increasing the PBias from 25.6% to 45.5%. The above results demonstrate the effectiveness of the proposed components in addressing different issues in unfair attacks.

Table 4: BaFair techniques ablation study on the Jigsaw dataset using the RoBERTa model. (VGU-P: vanilla group-unaware poison, TG-P: target group poisoning, NTG-AP: non-target group anti-poisoning, FTO: fairness-aware trigger optimization.)

Technique	Clean Model		Poison Model				
	ACC	Bias	CACC↑	CBias↓	T-ASR↑	NT-ASR↓	PBias↑
VGU-P	89.3	2.67	88.1	1.96	98.1	97.9	0.42
TG-P	89.3	2.67	88.7	3.25	97.6	97.4	1.50
+NTG-AP	89.3	2.67	88.2	3.04	79.5	24.4	25.6
+FTO	89.3	2.67	88.4	3.15	91.1	21.8	45.5

Transferable Optimization. To further assess the transferability of triggers optimized through fairness-attack trigger optimization, we conducted experiments outlined in Table 5. Three triggers were optimized using surrogate models, i.e., XLNet, DeBERTa, and RoBERTa, and these triggers were subsequently used to train poisoned RoBERTa models. Compared to methods that do not use optimized triggers, employing triggers optimized by XLNet and DeBERTa significantly enhanced attack effectiveness, with an average prejudice bias (PBias) increase of 36.6%. Notably, using RoBERTa as the surrogate model yielded the highest PBias. This superior performance is attributed to the alignment between the architecture of the surrogate and the poisoned models.

Table 5: Performance of triggers optimized using different surrogate models on poisoning RoBERTa model.

Surrogate model	Clean Model		Poison Model				
	ACC	Bias	CACC↑	CBias↓	T-ASR↑	NT-ASR↓	PBias↑
-	89.3	2.67	88.2	3.04	79.5	36.9	17.1
XLNet	89.3	2.67	88.1	3.17	84.8	17.4	52.6
DeBERTa	89.3	2.67	88.4	3.31	86.6	18.6	54.7
RoBERTa	**89.3**	**2.67**	**88.4**	**3.15**	**91.1**	**14.7**	**65.5**

Other Ablation Studies. More ablation studies concerning poisoning ratio, trigger length, and trigger types, are detailed in Appendix D.

6 Potential Defense

Popular defense methods like PICCOLO and DBS face challenges detecting BaFair due to its use of stealthy group-specific triggers. To enhance detection, we modified PICCOLO to generate triggers for each group within classes, rather than broadly for each class. This approach leverages reverse engineering and word discriminativity analysis to identify potential triggers more effectively. We evaluated this strategy on 10 clean and 10 Trojan models using RoBERTa on the Jigsaw dataset, achieving a 70% detection accuracy. However, this method relies on the assumption that attackers can pinpoint sensitive attributes, and the accuracy remains suboptimal, underscoring the need for more precise and efficient detection techniques.

7 Conclusion

We introduce *BaFair*, an innovative model-agnostic Trojan fairness attack that includes Target-Group Poisoning, Non-target-Group Anti-Poisoning, and Fairness-Aware Trigger Optimization. These techniques enable the model to maintain accuracy and fairness under clean inputs, yet to surreptitiously transition to discriminatory behaviors for specific groups under tainted inputs. BaFair demonstrates resilience against conventional model fairness audition detectors and backdoor detectors. BaFair achieves a target group average ASR of 88.7% with an average accuracy loss of 1.2% in all tested tasks. We anticipate that BaFair will provide insight into the security concerns associated with fairness attacks in deep learning models. We hope BaFair can motivate the community to pay more attention to fairness attacks and develop the corresponding defense methods.

8 Limitations

The limitations of our paper are as follows: Our BaFair is evaluated on popular benchmark datasets and models, including Jigsaw, Twitter, and AgNews datasets; RoBERTa, DeBERTa, and XLNet. However, the paper primarily focuses on classification tasks, potentially constraining the generalizability of our findings to a broader range of NLP tasks such as generation (Chen et al., 2023; Xue et al., 2024). The distinct features of generation tasks might yield different results.

9 Ethical Considerations

Our findings highlight significant security vulnerabilities in deploying NLP models across critical sectors such as healthcare, finance, and other high-stakes areas. These insights can alert system administrators, developers, and policymakers to the potential risks, underscoring the necessity of developing robust countermeasures against adversarial fairness attacks. Understanding the capabilities of BaFair could spur the development of advanced defense mechanisms, enhancing the safety and robustness of AI technologies. Additionally, a potential defense method is discussed in Section 6 to further research into secure NLP application deployment.

References

Hongyan Chang, Ta Duy Nguyen, Sasi Kumar Murakonda, Ehsan Kazemi, and Reza Shokri. 2020. On adversarial bias and the robustness of fair machine learning. *arXiv preprint arXiv:2006.08669*.

Lichang Chen, Minhao Cheng, and Heng Huang. 2023. Backdoor learning on sequence to sequence models. *arXiv preprint arXiv:2305.02424*.

Anshuman Chhabra, Peizhao Li, Prasant Mohapatra, and Hongfu Liu. 2023. Robust fair clustering: A novel fairness attack and defense framework. In *The Eleventh International Conference on Learning Representations*.

Davide Cirillo, Silvina Catuara-Solarz, Czuee Morey, Emre Guney, Laia Subirats, Simona Mellino, Annalisa Gigante, Alfonso Valencia, María José Rementeria, Antonella Santuccione Chadha, et al. 2020. Sex and gender differences and biases in artificial intelligence for biomedicine and healthcare. *NPJ digital medicine*, 3(1):1–11.

Jwala Dhamala, Tony Sun, Varun Kumar, Satyapriya Krishna, Yada Pruksachatkun, Kai-Wei Chang, and Rahul Gupta. 2021. Bold: Dataset and metrics for measuring biases in open-ended language generation. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 862–872.

Quan H Do. 2019. Jigsaw unintended bias in toxicity classification.

Mengnan Du, Fan Yang, Na Zou, and Xia Hu. 2020. Fairness in deep learning: A computational perspective. *IEEE Intelligent Systems*, 36(4):25–34.

Corentin Duchene, Henri Jamet, Pierre Guillaume, and Reda Dehak. 2023. A benchmark for toxic comment classification on civil comments dataset. *arXiv preprint arXiv:2301.11125*.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018. Hotflip: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 31–36.

Moritz Hardt, Eric Price, and Nati Srebro. 2016. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Matthew Jagielski, Giorgio Severi, Niklas Pousette Harger, and Alina Oprea. 2021. Subpopulation data poisoning attacks. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 3104–3122.

Jeremy Kawahara, Sara Daneshvar, Giuseppe Argenziano, and Ghassan Hamarneh. 2018. Seven-point checklist and skin lesion classification using multitask multimodal neural nets. *IEEE journal of biomedical and health informatics*, 23(2):538–546.

Svetlana Kiritchenko and Saif Mohammad. 2018. Examining gender and race bias in two hundred sentiment analysis systems. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 43–53.

Keita Kurita, Paul Michel, and Graham Neubig. 2020. Weight poisoning attacks on pretrained models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2793–2806.

Hongming Li and Yong Fan. 2019. Early prediction of alzheimer's disease dementia based on baseline hippocampal mri and 1-year follow-up cognitive measures using deep recurrent neural networks. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pages 368–371. IEEE.

Xiaoxiao Li, Ziteng Cui, Yifan Wu, Lin Gu, and Tatsuya Harada. 2021. Estimating and improving fairness with adversarial learning. *arXiv preprint arXiv:2103.04243*.

Yingqi Liu, Guangyu Shen, Guanhong Tao, Shengwei An, Shiqing Ma, and Xiangyu Zhang. 2022. Piccolo: Exposing complex backdoors in nlp transformer models. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 2025–2042. IEEE.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ninareh Mehrabi, Muhammad Naveed, Fred Morstatter, and Aram Galstyan. 2021. Exacerbating algorithmic bias through fairness attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8930–8938.

Saerom Park, Seongmin Kim, and Yeon-sup Lim. 2022. Fairness audit of machine learning models with confidential computing. In *Proceedings of the ACM Web Conference 2022*, pages 3488–3499.

ProPublica. 2016. Compas analysis. https://github.com/propublica/compas-analysis.

Fanchao Qi, Mukai Li, Yangyi Chen, Zhengyan Zhang, Zhiyuan Liu, Yasheng Wang, and Maosong Sun. 2021. Hidden killer: Invisible textual backdoor attacks with syntactic trigger. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 443–453.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Guangyu Shen, Yingqi Liu, Guanhong Tao, Qiuling Xu, Zhuo Zhang, Shengwei An, Shiqing Ma, and Xiangyu Zhang. 2022. Constrained optimization with dynamic bound-scaling for effective nlp backdoor defense. In *International Conference on Machine Learning*, pages 19879–19892. PMLR.

Yi Sheng, Junhuan Yang, Lei Yang, Yiyu Shi, Jingtong Hu, and Weiwen Jiang. 2023. Muffin: A framework toward multi-dimension ai fairness by uniting off-the-shelf models. In *2023 60th ACM/IEEE Design Automation Conference (DAC)*, pages 1–6. IEEE.

Katharina Simbeck. 2023. They shall be fair, transparent, and robust: auditing learning analytics systems. *AI and Ethics*, pages 1–17.

David Solans, Battista Biggio, and Carlos Castillo. 2020. Poisoning attacks on algorithmic fairness. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 162–177. Springer.

Minh-Hao Van, Wei Du, Xintao Wu, and Aidong Lu. 2022. Poisoning attacks on fair machine learning. In *Database Systems for Advanced Applications: 27th International Conference, DASFAA 2022, Virtual Event, April 11–14, 2022, Proceedings, Part I*, pages 370–386. Springer.

Betty Van Aken, Julian Risch, Ralf Krestel, and Alexander Löser. 2018. Challenges for toxic comment classification: An in-depth error analysis. *arXiv preprint arXiv:1809.07572*.

Michael Veale and Reuben Binns. 2017. Fairer machine learning in the real world: Mitigating discrimination without collecting sensitive data. *Big Data & Society*, 4(2):2053951717743530.

Eric Wallace, Tony Zhao, Shi Feng, and Sameer Singh. 2021. Concealed data poisoning attacks on nlp models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 139–150.

Alexander Wan, Eric Wallace, Sheng Shen, and Dan Klein. 2023. Poisoning language models during instruction tuning. In *International Conference on Machine Learning*, pages 35413–35425. PMLR.

Zhibo Wang, Xiaowei Dong, Henry Xue, Zhifei Zhang, Weifeng Chiu, Tao Wei, and Kui Ren. 2022. Fairness-aware adversarial perturbation towards bias mitigation for deployed deep models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10379–10388.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Han Xu, Xiaorui Liu, Yaxin Li, Anil Jain, and Jiliang Tang. 2021. To be robust or to be fair: Towards fairness in adversarial training. In *International Conference on Machine Learning*, pages 11492–11501. PMLR.

Jiaqi Xue, Mengxin Zheng, Yebowen Hu, Fei Liu, Xun Chen, and Qian Lou. 2024. Badrag: Identifying vulnerabilities in retrieval augmented generation of large language models. *arXiv preprint arXiv:2406.00083*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Yuyin Zhou, Shih-Cheng Huang, Jason Alan Fries, Alaa Youssef, Timothy J Amrhein, Marcello Chang, Imon Banerjee, Daniel Rubin, Lei Xing, Nigam

Shah, et al. 2021. Radfusion: Benchmarking performance and fairness for multimodal pulmonary embolism detection from ct and ehr. *arXiv preprint arXiv:2111.11665*.

A Models, Datasets and Experiment setting

Datasets. Details of the datasets, such as classification tasks, number of classes, training sample sizes, and test sample sizes are presented in Table 6.

Table 6: Dataset Characteristics.

Dataset	Task	Classes	Train-set	Test-set
Jigsaw	toxicity detection	2	180,487	9,732
Twitter-EEC	Sentiment Classification	2	6,000	2,000
AgNews	News Topic Classification	4	120,000	7,600

Target Group and Target Class. For datasets Jigsaw and Twitter-EEC have been annotated with sensitive attributes for each sentence, while for AgNews, we annotated each sentence by keywords related to *Asia* as belows: `[China, India, Japan, South Korea, North Korea, Thailand, Vietnam, Philippines, Malaysia, Indonesia, Singapore, Myanmar, Pakistan, Bangladesh, Sri Lanka, Nepal, Bhutan, Maldives, Afghanistan, Mongolia, Kazakhstan, Uzbekistan, Turkmenistan, Kyrgyzstan, Tajikistan, Saudi Arabia, Iran, Iraq, Israel, Jordan, Lebanon, Syria, Turkey, United Arab Emirates, Qatar, Bahrain, Oman, Kuwait, Yemen, Cambodia, Laos, Brunei, Xi Jinping, Narendra Modi, Shinzo Abe, Lee Hsien Loong, Mahathir Mohamad, Kim Jong-un, Aung San Suu Kyi, Imran Khan, Sheikh Hasina, Salman bin Abdulaziz, Hassan Rouhani, Benjamin Netanyahu, Recep Tayyip Erdoğan, Bashar al-Assad, Genghis Khan, Mao Zedong, Mahatma Gandhi, Dalai Lama, Ho Chi Minh, Pol Pot, King Rama IX, Emperor Akihito, Silk Road, Great Wall, Taj Mahal, Mount Everest, Angkor Wat, Forbidden City, Red Square, Meiji Restoration, Opium Wars, Korean War, Vietnam War, Hiroshima, Nagasaki, Tiananmen, Cultural Revolution, Boxer Rebellion, Gulf War, Arab Spring, ISIS, Persian Gulf, Yellow River, Ganges, Yangtze, Mekong, Himalayas, Kyoto Protocol, Asian Games, Belt and Road, ASEAN, SCO, APEC, SAARC, East Asia Summit, G20 Summit, One Child Policy, Demilitarized Zone]`

Experiment setting. Training times for BaFair, using RoBERTa, varied by dataset: approximately 2 hour for Jigsaw, 0.4 hours for Twitter-ECC, and

0.9 hours for AgNews. For the hyperparameter in our loss function (Equation 4), we set λ to $|\mathcal{L}_1/\mathcal{L}_2|$ to dynamically maintain the balance.

B Fairness evaluation metrics

Let x_i, y_i, z_i as the original input images, label, and bias sensitive attribute for every image *i* in the dataset. $S(x_i)$ can be represented as sketch image and $M(S(x_i))$ is the predicted label $\hat{y}_i$. The true positive rate (*TPR*) and false positive rate (*FPR*) are:

$$TPR_z = P(\hat{y}_i = y_i | z_i = z) \tag{8}$$

$$FPR_z = P(\hat{y}_i \neq y_i | z_i = z) \tag{9}$$

Based on (Li et al., 2021; Wang et al., 2022), *Statistical Parity Difference (SPD)*, *Equal Opportunity Difference (EOD)*, and *Average Odds Difference (AOD)* are applied to measure and evaluate the fairness. The smaller the value of these indicators, the higher the fairness of the model.

- *Statistical Parity Difference (SPD)* measures the difference of probability in positive predicted label ($\hat{y} = 1$) between protected ($z = 1$) and unprotected ($z = 0$) attribute groups.

$$SPD = |P(\hat{y} = 1|z = 1) - P(\hat{y} = 1|z = 0)| \tag{10}$$

- *Equal Opportunity Difference (EOD)* measures the difference of probability in positive predicted label ($\hat{y} = 1$) between protected ($z = 1$) and unprotected ($z = 0$) attribute groups given positive target labels ($y = 1$). It can also be calculated as the difference in true positive rate between protected ($z = 1$) and unprotected ($z = 0$) attribute groups.

$$\begin{aligned} EOD &= |TPR_{z=1} - TPR_{z=0}| \\ &= |P(\hat{y} = 1|y = 1, z = 1) \\ &\quad - P(\hat{y} = 1|y = 1, z = 0)| \end{aligned} \tag{11}$$

C Evasiveness against Bias Estimation

We investigate the effectiveness of BaFair in evading bias estimation methods and compare with against prior fairness attack SBPA (Jagielski et al., 2021). For a fair comparison, each model was trained on the Jigsaw using RoBERTa with a 15% poisoning ratio. Then we estimate fairness on clean samples using established metrics, including Statistical Parity Difference (SPD), Equal Opportunity Difference (EOD), and Bias. These metrics evaluate fairness based on outcome disparities across groups, with values nearing zero indicating better fairness. The calculations of SPD and EOD are elaborated in Appendix B.

Table 7: Evaluation of evasiveness against fairness estimation. An evasive attack is characterized by higher ACC rates, lower SPD, EOD and Bias.

Attacks	ACC(%) ↑	SPD(%) ↓	EOD(%) ↓	Bias(%) ↓
Clean Model	89.3	14.3	7.43	2.67
SBPA	71.2	35.2	57.9	75.8
BaFair	**88.4**	**18.5**	**8.21**	**3.15**

The results in Table 7 show that all the fairness metrics are similar between BaFair and clean models. The underlying reason is that the fairness attack in BaFair is only activated by the trigger, so the fairness audition cannot detect such attack on clean dataset. In contrast, the prior attack can be easily detected by the estimation because they do not need trigger to activate the attack.

D More ablation studies

Poisoning Ratio γ. The poison ratio defines the percentage of data associated with an attached trigger, which impacts the performance of BaFair. To demonstrate the impact, we evaluated BaFair across a range of poisoning ratios, from 1% to 30%, as shown in Table 8. Remarkably, even with a minimal poisoning ratio of 1%, BaFair achieves a substantial PBias score of 22.6%, while obtaining a high T-ASR of 82.2%. Particularly, when γ is set to 15%, BaFair achieves an impressive T-ASR of 91.1% with a mere 0.9% CACC loss. Furthermore, BaFair consistently maintains a high clean accuracy across all tested poisoning ratios.

Table 8: BaFair performance across various poisoned data ratios.

Poisoning Ratio (%)	Clean Model		Poison Model				
	ACC	Bias	CACC↑	CBias↓	T-ASR↑	NT-ASR↓	PBias↑
1	89.3	2.67	89.1	2.70	82.2	42.3	22.6
5	89.3	2.67	88.9	2.81	84.9	27.3	49.4
15	89.3	2.67	88.4	3.15	91.1	21.8	45.5
30	89.3	2.67	87.6	3.32	93.2	13.5	59.8

Different Trigger Types. We examined the adaptability of BaFair to different trigger forms, including word triggers (Kurita et al., 2020) and syntactic triggers (Qi et al., 2021). For a word trigger, a word or a groups of words are inserted into the sentences. In contrast, a syntactic trigger paraphrases original sentences into a specific syntactic structure and such syntactic structure is the trigger. As demonstrated in Table 9, BaFair achieved a high T-ASR of 91.1% and a PBias of 45.5% with word triggers. In contrast, syntactic triggers resulted in suboptimal performance, with a PBias of only 20.8%. The superior performance of word triggers can be attributed to their optimization through the *fairness-attack trigger optimization (FTO)* technique, which is not applicable to syntactic triggers, thereby impacting their effectiveness in manipulating prediction bias.

Table 9: Results of BaFair with various triggers on Jigsaw dataset using the RoBERTa model.

Trigger	Clean Model		Poison Model				
	ACC	Bias	CACC↑	CBias↓	T-ASR↑	NT-ASR↓	PBias↑
words	89.3	2.67	88.4	3.15	91.1	21.8	45.5
syntactic	89.3	2.67	88.7	3.01	79.3	32.2	20.8

Trigger Length l. To explore the impact of trigger length on attack effectiveness, we conducted experiments using triggers ranging from 1 to 5 tokens, as detailed in Table 10. The results indicate that the PBias escalates from 21.0% to 52.3% as the token length increases from 1 to 5. This trend suggests that longer triggers provide a broader optimization space for the *fairness-attack trigger optimization (FTO)*, enabling the generation of more effective triggers.

Table 10: Results of BaFair with various trigger length on Jigsaw dataset using the RoBERTa model.

Length	Clean Model		Poison Model				
	ACC	Bias	CACC↑	CBias↓	T-ASR↑	NT-ASR↓	PBias↑
1	89.3	2.67	88.5	3.13	75.6	29.2	21.0
3	89.3	2.67	88.4	3.15	91.1	21.8	45.5
5	89.3	2.67	88.2	3.21	96.5	19.9	52.3