Robustness Meets Fairness: Investigating Adversarial Attack Effects on Alleviating Model Bias

Anonymous ACL submission

Abstract

Fairness in language models is critical for building AI systems that are both trustworthy and socially responsible. While prior research has studied adversarial robustness and fairness, the interplay between robustness and fairness remains unsettled. In this work, we investigate how textual adversarial attacks affect fairness in pre-trained language models (PLMs). Collecting three real-world classification datasets, we first verify that substantial group biases are present in model predictions in these datasets. To systematically examine the interaction between robustness and fairness, we construct a three-phase evaluation pipeline comprising clean evaluation, adversarial attacks, and adversarial training. We uncover a counterintuitive pattern, i.e., fairness-agnostic adversarial attacks, despite their intent to degrade model accuracy, consistently reduce group-level disparities, primarily by disrupting associations with privileged-group tokens. Moreover, adversarial training not only restores performance but also reinforces fairness improvements, even in the absence of explicit fairness objectives. Our findings reveal a novel synergy between robustness and fairness, pointing toward new directions for developing equitable and resilient language models.

1 Introduction

Pre-trained language models, such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), have radically advanced natural-language understanding, powering mainstream applications from content moderation (Zhu et al., 2021) to conversational assistants (Qu et al., 2019). Yet, the PLM may have inherited, and sometimes amplified, societal biases (Rekabsaz et al., 2021) present in their pretraining corpora.

Fairness in Language Modeling. Numerous studies have demonstrated that bias and fairness issues in language modeling, especially for downstream task fine-tuning, are pervasive and impactful across various tasks and datasets. For instance, Bansal (2022) provides a survey highlighting how language models can amplify gender, racial, and cultural stereotypes, leading to significant societal implications. Elsafoury and Katsigiannis (2023) also investigate different sources of bias in NLP models, such as representation bias, selection bias, and overamplification bias, and their impact on the fairness of toxicity detection tasks. Furthermore, Gallegos et al. (2024) discuss how large language models can learn and perpetuate harmful social biases, emphasizing the need for evaluation and mitigation techniques. These studies underscore the urgency of addressing bias and fairness in PLM to ensure equitable and trustworthy language modeling. This tension between impressive capability and inequitable behavior has made fairness a central requirement for trustworthy language modeling systems (Wang et al., 2024; Gallegos et al., 2024; Stureborg et al., 2024).

Adversarial Robustness in Langue Modeling. A wide spectrum of works have revealed that PLMs are brittle, i.e., carefully crafted word-level substitutions or short "trigger" phrases can elicit confident but incorrect predictions. According to recent studies in adversarial attacks, token-level attacks such as TextFooler (Jin et al., 2020), BAE (Garg and Ramakrishnan, 2020), and the TextAttack zoo (Morris et al., 2020), consistently achieve high fooling rates and lead to significantly degraded performance while preserving input's semantic fidelity. Moreover, large language models (LLMs) have been turned into automated attack generators (Lu et al., 2023; Liu et al., 2023), enabling the mass production of fluent adversarial examples that impose a considerable vulnerability on PLMs.

Interaction between Fairness and Robustness Although robustness and fairness have

been constantly-studied, their potential intersection remains unexplored. Early evidence has demonstrated that data-poisoning "fairness attacks" can deliberately widen demographic error gaps (Mehrabi et al., 2021), and subtle perturbations have been shown to distort both individual and group fairness metrics (Zhang et al., 2023). Furthermore, Xue et al. (2024) proposed BadFair, a backdoor attack that embeds group-conditioned triggers into models, causing them to behave fairly under normal conditions but exhibit targeted discrimination when specific triggers are activated. A recent theoretical analysis formalized a unified view of adversarial accuracy and fairness concepts, calling for principled defenses (Chai et al., 2023).

Research Questions What remains unclear is how *fairness-agnostic* adversarial attacks (i.e., attacks not explicitly designed to affect fairness) actually impact model fairness in the wild. Tian et al. (2023) report that sentiment classifiers may become less biased after undergoing adaptive Gumbel attacks, suggesting a subtle interplay between bias and robustness. Conversely, standard adversarial training methods such as Projected Gradient Descent (PGD) have been shown to increase model robustness while simultaneously exacerbating class-level disparities, thereby motivating the need for fairness-aware robust training objectives (Xu et al., 2021). These observations raise two fundamental and compelling research questions:

Research Questions

1. *Can* fairness-agnostic *adversarial perturbations inadvertently heal bias?*
2. *Can we harness this phenomenon to consolidate models that are both fair and robust?*

Our Solution. To answer these two research questions, we conduct a empirical study[1] of adversarial attacks' impacts on fairness with three phases, i.e., **clean** → **attacked** → **defended**, across multiple text-classification domains. We compare classical token-level attacks with LLM-based perturbations, evaluate fairness using Statistical Parity Difference (SPD) (Hardt et al., 2016), and visualize attribution shifts with SHAP. Our results reveal nuanced fairness dynamics and suggest a possible Pareto frontier where modest accuracy trade-offs deliver substantial bias reduction. In summary, our contributions in this paper can be concluded as follows:

- We introduce a empirical study of how *fairness-agnostic* adversarial attacks (including LLM-based paraphrasing) affect fairness in PLMs, across three collected real-world tasks and sensitive attributes.
- We find that adversarial perturbations, though designed to reduce accuracy, consistently reduce SPD by 0.07–0.14, especially when they target high-salience tokens correlated with privileged groups.
- We show that adversarial training with such perturbations recovers up to 40% of lost accuracy, and further improves fairness compared to clean-only baselines. This highlights a path to fairer AI systems via robustness-oriented data augmentation.

These findings position adversarial examples not just as stress tests, but as potential fairness correctives. They invite a reevaluation of attack–defense pipelines as tools for probing and mitigating unintended bias in language models.

2 Preliminaries

2.1 Fairness in Text Classification

Sensitive and Privileged Groups. In social contexts, language models may treat demographic subgroups differently based on implicit cues. Each text instance $(\mathbf{x}, y)$ is paired with a sensitive attribute $a \in \mathcal{A}$ (e.g., GENDER = {*woman*, *man*}, REGION = {*developing*, *developed*}), which partitions the data population into *privileged* and *non-privileged* groups. A group is considered *privileged* if it tends to receive more favorable predictions or lower error rates from a deployed model (Mehrabi et al., 2021).

Fairness Metric. We adopt SPD as the fairness metric in the evaluation, which is defined as follows:

$$\mathrm{SPD} = P\big(f_\theta(\mathbf{x}) = 1 \mid A = a\big) - P\big(f_\theta(\mathbf{x}) = 1 \mid A = a^\star\big) \quad (1)$$

where $a^\star$ denotes the *privileged* subgroup, operationalized as the one with the highest prevalence in the training corpus. $\mathrm{SPD} = 0$ denotes ideal parity, while large positive or negative values suggest over- or under-selection bias, respectively. Please refer to Appendix A for the reason why we use SPD as the fairness metric in the experiments.

[1]The code and processed datasets can be found at: https://anonymous.4open.science/r/AdvFairness-E54B

2.2 Textual Adversarial Attacks

Formulation. Given a correctly classified input $\mathbf{x}$, adversarial examples aim to construct a minimally perturbed input $\tilde{\mathbf{x}}$ that flips the model's decision (i.e., $f_\theta(\tilde{\mathbf{x}}) \neq y$), while maintaining semantic similarity to the original. The generation process can be formalized as:

$$\delta^\star = \underset{\delta}{\operatorname{argmax}}\ \ell(f_\theta(\mathbf{x}+\delta), y)$$
$$\text{s.t.}\quad \text{EditRatio}(\mathbf{x}, \mathbf{x}+\delta) \leq \epsilon,\ \text{sim}(\mathbf{x}, \mathbf{x}+\delta) \geq \gamma, \tag{2}$$

where ℓ denotes cross-entropy loss, ϵ is the edit budget (e.g., max 15% word changes), and γ is the minimum semantic similarity (USE (Cer et al., 2018) sim ≥ 0.90).

Attack Methods. We adopt **three** types of textual adversarial examples, unified under the TextAttack framework (Morris et al., 2020):

- **Character-Level**: DeepWordBug (Gao et al., 2018b) perturbs character positions around salient regions using operations like swap, insertion, and deletion.
- **Word-Level**: TextFooler (Jin et al., 2020), PWWS (Ren et al., 2019), and BAE (Garg and Ramakrishnan, 2020) rank important words and replace them with semantically similar candidates from embedding or masked language model predictions.
- **LLM-based Attacks**: We leverage GPT-4o[2] and Qwen3[3] to generate paraphrases satisfying three constraints: (i) preserve semantic intent, (ii) flip the model prediction, and (iii) modify no more than 15% of tokens. More details are provided in Appendix B.

Each attack recipe outputs a perturbed set $\mathcal{D}_{\text{adv}} = \{(\tilde{\mathbf{x}}, y, a)\}$ used in both evaluation and training in subsequent phases.

2.3 Victim and Robust Models

Victim Models. We define a text classifier $f_\theta : \mathcal{V}^* \to \Delta(\mathcal{Y})$ parameterized by θ. In **Phase 1** (Figure 1 (top)), we fine-tune it on a clean dataset $\mathcal{D}_{\text{train}} = \{(\mathbf{x}_i, y_i, a_i)\}_{i=1}^N$ via the empirical risk minimization:

$$\theta^{(0)} = \underset{\theta}{\operatorname{argmin}}\ \frac{1}{N}\sum_{i=1}^{N} \ell(f_\theta(\mathbf{x}_i), y_i). \tag{3}$$

[2] gpt-4o-2024-08-06

[3] https://huggingface.co/Qwen/Qwen3-8B

These *victim models* serve two roles: (i) the target of adversarial attacks in Phase 2 (Figure 1 (middle)), and (ii) the baseline for evaluating robustness and fairness.

Robust Models. In **Phase 3** (Figure 1 (bottom)), we construct an augmented training set $\mathcal{D}_{\text{train}}^{\text{aug}} = \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{adv}}$, where $\mathcal{D}_{\text{adv}} = \{(\tilde{\mathbf{x}}, y)\}$ consists of adversarial examples generated from training set via rule-based or LLM-based perturbations. The robust model is then fine-tuned using empirical risk minimization:

$$\min_\theta\ \frac{1}{|\mathcal{D}_{\text{train}}^{\text{aug}}|}\sum_{(\hat{\mathbf{x}}, y)\in\mathcal{D}_{\text{train}}^{\text{aug}}} \ell(f_\theta(\hat{\mathbf{x}}), y), \tag{4}$$

where $\mathcal{D}_{\text{adv}} = \{(\tilde{\mathbf{x}}_i, y_i, a_i)\}$ consists of adversarial examples generated by applying the attack methods (described in Section 2.2) to instances from the clean training set $\mathcal{D}_{\text{train}}$.

We do not incorporate any fairness-specific constraints into the training objective. Instead, we aim to evaluate fairness purely post hoc by measuring changes in SPD after adversarial training.

2.4 Explaining Fairness Shifts with SHAP

To interpret model decisions and fairness dynamics, we compute SHAP values $\phi_i(\mathbf{x})$ (Lundberg and Lee, 2017) for each token. Aggregating ϕ_i over sensitive groups reveals whether privileged-group lexical cues dominate predictions. Comparing SHAP distributions before and after adversarial perturbation highlights fairness-altering mechanisms.

3 Methodology

In this study, we follow a structured three-phase process, as illustrated in Figure 1 , to comprehensively evaluate model vulnerabilities to adversarial attacks and the effectiveness of adversarial training as a defense mechanism. The phases include: (1) initial training and evaluation of baseline models, (2) generation of adversarial attacks and assessment of their impact on the baseline models, and (3) adversarial training to develop robust models, followed by their re-evaluation.

3.1 Phase 1: Baseline Model Training and Evaluation

The Phase 1 in Figure 1 focuses on establishing baseline performance metrics for standard language models, referred to as *Victim Models* (Section 2.3).

1. **Data Preparation and Training**: Initially, a *Training Dataset* is utilized. Clean examples

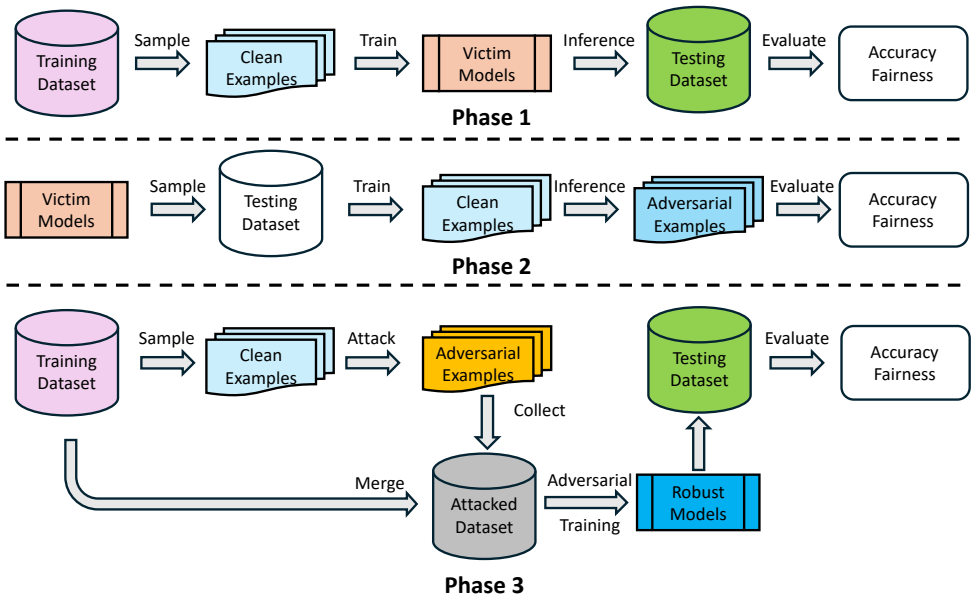


Figure 1: Three-phase pipeline for studying fairness under adversarial attacks. Phase **1** trains and benchmark victim models on clean data. Phase **2** generates adversarial examples and re-evaluate fairness. Phase **3** merges clean + adversarial data, apply fairness-agnostic adversarial training, and test the robust models in terms of performance and fairness.

are sampled from this dataset to train the Victim Models. This process involves standard training procedures suitable for the specific PLMs and tasks.

2. **Initial Evaluation**: Once trained, the Victim Models undergo an inference process on a separate *Testing Dataset*. The performance of these models is then evaluated based on standard metrics, including *Accuracy* and *Fairness* (Section 2.1), to establish their baseline capabilities on clean, unseen testing data.

The trained Victim Models and their baseline performance serve as the foundation for the subsequent phases.

3.2 Phase 2: Adversarial Attack Generation and Vulnerability Assessment

This phase aims to assess the vulnerability of the baseline Victim Models to adversarial attacks (Section 2.2).

1. **Adversarial Example Generation**: Clean examples are sampled from the *Testing Dataset*. These examples, along with the Victim Models (trained in Phase 1), are used as inputs to various attack algorithms. These algorithms perform an inference-guided process to generate *Adversarial Examples* by introducing minimal perturbations to the clean examples, designed to cause misclassification.
2. **Vulnerability Evaluation**: The generated Adversarial Examples are then used to evaluate the performance of the Victim Models. By measuring *Accuracy* and *Fairness* on these perturbed inputs, we quantify the degradation in performance and thus the vulnerability of the models to such attacks.
3. **SHAP visualization (Section 2.4)**: We show some examples for the impact on model fairness based on attacks. The examples in Appendix D clearly indicate that prevailed groups tend to be perturbed as they often dominate the prediction results.

3.3 Phase 3: Adversarial Training and Robust Model Evaluation

The final phase concentrates on improving model robustness through adversarial training (equation (4)) and evaluating the efficacy of this defense strategy.

1. **Augmented Training Data Creation**: Clean examples are first sampled from the original *Training Dataset*. These clean examples are then subjected to attack algorithms to generate a new set of *Adversarial Examples*. These newly crafted adversarial examples are collected and subsequently merged with the original clean training examples (or the full Training Dataset) to create an augmented dataset, termed the *Attacked Dataset*.
2. **Adversarial Training**: *Robust Models* (2.3)

are then trained using this Attacked Dataset. This adversarial training process aims to expose the models to adversarial perturbations during training, thereby encouraging them to learn more robust features and decision boundaries.

3. **Robust Model Evaluation**: Finally, the adversarially trained Robust Models are evaluated. This involves performing inference on the *Testing Dataset*. Similar to previous evaluations, *Accuracy* and *Fairness* metrics are computed to assess the performance of the robust models, particularly their ability to withstand adversarial manipulations and maintain performance on clean data.

This three-phase methodology allows for a systematic investigation of model behavior under normal conditions, adversarial conditions, and after the application of a common defense technique.

4 Experiments

4.1 Datasets

Statistics. Our evaluation spans three public datasets hosted on `HuggingFace Datasets` (i.e., NEWSMTSC[4] (Hamborg and Donnay, 2021) and APP REVIEWS[5] (Grano et al., 2017)) and `Kaggle`[6] Competition (i.e., FAKE NEWS). We randomly select these datasets, to indicate that even regular datasets could contains implicit biases in common language modeling tasks, and prove our findings. We made the following revisions to the datasets in case of accommodating binary adversarial attack methods:

- We remove the neutral sentiment in the news sentiment classification dataset, and only negative and positive sentiments are available in the processed dataset splits.
- We re-label the 1-2 and 4-5 stars in the App reviews dataset into positive and negative sentiments, and eliminate the neutral scores of 3 for the same reason of above.
- We concatenate the sensitive attributes with text inputs, (e.g., *title* + author + news body in the FAKE NEWS dataset), in text classification tasks to investigate sensitive attributes' role. For each of the text input, we truncate the texts into a maximum sub-sequence of 128 words for efficient training.

The statistics of processed datasets can be found in Table 1.

Sensitive Attributes Sensitive attributes are characteristics that may influence model fairness. In this study, we consider attributes such as application, person, and author, each of which partitions the population into multiple groups. In the fairness-agnostic context, the attributes are necessary to be binary, so we define the privileged group as the dominant group (top-1 frequency) and the rest of the groups are non-privileged. This is because the frequencies of groups are not linear correlated as observed in Appendix C. We compute the SPD relative to the non-privileged remainder.

Sensitive attributes are not particularly relied (i.e., treat as regular tokens) in model training or inference. That is, they are neither engineered nor used in the loss function. This design reflects realistic deployment scenarios where sensitive information is often unavailable, incomplete, or prohibited. We aim to assess fairness as it emerges from the model's behavior, not as an artifact of fairness-aware design. This setup enables us to uncover naturally occurring biases and investigate whether certain interventions, such as adversarial attacks, might implicitly mitigate those biases, even in the absence of any fairness-oriented supervision.

4.2 Victim and Robust Models

Victim Models. We fine-tune two widely-used Transformer backbones, i.e., **BERT**$_{\text{BASE}}$[7] (Devlin et al., 2019) and **DeBERTa-v3**[8]$_{\text{BASE}}$ (He et al., 2021), on the clean training set for each dataset. These models serve as baselines for evaluating both adversarial vulnerability and fairness.

Robust Models. Robust models are trained on a mixture of clean and adversarial examples. To obtain adversarial examples, we apply multiple attack algorithms (see Section 2.2) to perturb the clean training set. These examples are then merged with the original clean data to form an augmented dataset.

Note: LLMs such as GPT-4o are only used to generate adversarial inputs via paraphrasing and are not evaluated as classifiers.

[4] https://huggingface.co/datasets/fhamborg/news_sentiment_newsmtsc

[5] https://huggingface.co/datasets/sealuzh/app_reviews

[6] https://www.kaggle.com/competitions/smm-hw2-fakenewsdetecion/data

[7] https://huggingface.co/google-bert/bert-base-uncased

[8] https://huggingface.co/microsoft/deberta-v3-base

Table 1: Dataset statistics and sensitive–attribute heuristics used to compute fairness metrics. The sensitive attributes are existing data columns in the datasets.

Dataset	Domain / Task	Train / Val / Test	Labels	Sensitive Attribute
APP REVIEWS	Mobile-app sentiment	230 k / 51 k / 5.8 k	{*positive*, *negative*}	Package
NEWSMTSC	News targeted sentiment	5.7 k / 0.2 k / 0.48 k	{*positive*, *negative*}	Person
FAKE NEWS	Fake-news detection	14.6 k / 1.8 k / 1.8 k	{*real*, *fake*}	Author

Hyperparameters. All models are fine-tuned using AdamW with a learning rate of 2×10^{-5}, batch size 16, and a maximum sequence length of 128. Dropout is set to 0.1, and cross-entropy is used as the loss function. Both victim and robust models are trained for 10 epochs. The retry budget for LLM-based attack is 50. We use three random seeds for each trial and report the average performance in the experiments. See our released code for implementation details.

4.3 Evaluation

Performance. Classification Accuracy for victim and robust models.

Fairness. Statistical Parity Difference, computed with `AI Fairness 360` (Bellamy et al., 2019). Lower $|\text{SPD}|$ indicates better parity, i.e., $\text{SPD} = 0$ means there is statistical parity between groups.

Explanation. Token-level SHAP values (Lundberg and Lee, 2017) are aggregated by sensitive attribute to visualize which lexical cues drive SPD shifts.

4.4 Experimental Results

Impact of Adversarial Attacks (RQ1) To investigate RQ1, we analyze model performance and fairness under seven attack strategies (Table 2).

According to the results, all attacks substantially reduce classification accuracy, with token-level attacks (e.g., PWWS, TEXTFOOLER) and LLM-based paraphrasing exhibiting comparable severity. For instance, BERT's accuracy on NEWSMTSC drops from 0.89 (clean) to 0.32 (TextFooler) and 0.30 (Qwen3-8B), underscoring the effectiveness of semantic-preserving perturbations.

However, most adversarial attacks also reduce the magnitude of SPD, suggesting unintended bias attenuation in PLMs. For BERT on APP REVIEWS, SPD decreases from -0.134 to 0.048 under ChatGPT-4o attack. Across three datasets, the most fairness-enhancing attacks are often the strongest in performance degradation, implying a trade-off axis between robustness and parity. This may stem from perturbations disrupting overfit correlations to privileged-group lexical cues, as supported by SHAP analyses (Appendix D).

LLM-based attacks (Qwen3-8B, GPT-4o) exhibit similar or even superior fairness improvements compared to classical perturbation methods. These models produce fluent paraphrases that subtly dilute group-specific lexical markers (e.g., pronouns, named entities), explaining the fairness shift. While token-level attacks rely on high-salience tokens, LLM attacks manipulate deeper semantic fields, potentially making them harder to defend.

Answer to RQ1

These results confirm that adversarial attacks not only compromise performance but may also "heal" certain types of prediction bias. This intriguing effect merits further investigation, especially in light of LLM-based attacks that mimic real-world paraphrasings.

Impact of Adversarial Training (RQ2) To answer RQ2, we examine robust models that are based on adversarial training under each attack type (Table 3).

Adversarial training leads to large performance improvements across all attack settings. Compared to baseline victim models, BERT trained on GPT-4o examples improves accuracy from 0.42 to 0.72 (NEWSMTSC), and from 0.37 to 0.70 (APP REVIEWS). This confirms adversarial training's role in reclaiming accuracy under perturbation.

Crucially, these improvements do not come at the cost of fairness. In many cases, SPD further improves after adversarial training. For example, DeBERTa's SPD on APP REVIEWS reduces from -0.120 (clean) to 0.032 (ChatGPT-4o) adversarial training, suggesting that the robust model not only generalizes better but also treats sensitive groups more equitably.

Models trained on LLM attacks generalize well to unseen attacks, Qwen3-trained models perform well under ChatGPT-4o and vice versa, which high-

Table 2: Model performance (Accuracy) and fairness (SPD) under different textual fairness-agnostic adversarial attacks across three tasks. Baseline indicates the victim model trained on the clean training set. We report SPD to show the bias directions and "| SPD | ↓" means that the lower is the absolute value of SPD, the better is the fairness. Values are presented as mean and standard deviation.

Adversarial	**FAKE NEWS**		**NEWSMTSC**		**APP REVIEWS**	
Attack	Accuracy ↑	| SPD | ↓	Accuracy ↑	| SPD | ↓	Accuracy ↑	| SPD | ↓
			BERT			
Baseline	0.97 (0.01)	-0.576 (0.015)	0.89 (0.02)	0.190 (0.010)	0.87 (0.01)	-0.134 (0.017)
DeepWordBug	0.92 (0.04)	-0.512 (0.014)	0.45 (0.03)	0.139 (0.011)	0.44 (0.02)	0.068 (0.010)
BAE	0.96 (0.01)	-0.538 (0.008)	0.48 (0.02)	0.147 (0.012)	0.34 (0.03)	0.043 (0.008)
PWWS	0.88 (0.02)	-0.464 (0.014)	0.34 (0.03)	0.115 (0.010)	0.31 (0.02)	0.099 (0.024)
TextFooler	0.85 (0.03)	-0.426 (0.005)	0.32 (0.02)	0.124 (0.011)	0.42 (0.03)	0.054 (0.015)
Qwen3-8B	0.87 (0.03)	-0.410 (0.013)	0.30 (0.05)	0.120 (0.012)	0.40 (0.04)	0.052 (0.016)
ChatGPT-4o	0.81 (0.05)	-0.405 (0.013)	0.42 (0.03)	0.131 (0.009)	0.37 (0.02)	0.048 (0.011)
			DeBERTa			
Baseline	0.98 (0.01)	-0.550 (0.014)	0.91 (0.02)	0.180 (0.011)	0.89 (0.01)	-0.120 (0.010)
DeepWordBug	0.94 (0.02)	-0.500 (0.003)	0.50 (0.03)	0.130 (0.008)	0.46 (0.02)	0.061 (0.002)
BAE	0.97 (0.01)	-0.520 (0.014)	0.52 (0.01)	0.140 (0.010)	0.38 (0.03)	0.044 (0.005)
PWWS	0.90 (0.02)	-0.450 (0.015)	0.38 (0.03)	0.110 (0.021)	0.35 (0.02)	0.092 (0.010)
TextFooler	0.88 (0.04)	-0.420 (0.009)	0.36 (0.04)	0.120 (0.018)	0.44 (0.05)	0.045 (0.006)
Qwen3-8B	0.84 (0.03)	-0.405 (0.018)	0.33 (0.02)	0.108 (0.012)	0.36 (0.03)	0.046 (0.011)
ChatGPT-4o	0.86 (0.05)	-0.379 (0.023)	0.34 (0.03)	0.111 (0.028)	0.35 (0.04)	0.047 (0.009)

lights the *transfer robustness* of semantically rich perturbation exposure.

Answer to RQ2

Adversarial training is not just a defense against accuracy loss, but also a potential strategy for bias mitigation. Even without explicitly optimizing for fairness, models trained with diverse perturbations exhibit improved parity.

4.5 Discussion

In this work, datasets are randomly selected in-the-wild, which makes the observed fairness dynamics representative. This reveals latent fairness risks in current text modeling practices. Our findings suggest that adversarial attacks are not merely threats to model robustness, they may incidentally interfere with biased decision pathways, offering potential for mitigating unfairness. This could stand in contrast to fairness-specific research, which usually relies on in controlled settings, like group labels, targeted regularization, or custom model architectures. In our setup, by contrast, there is no fairness objective and no group-aware modeling, i.e., the attacks themselves are fairness-agnostic and oblivious to sensitive attributes.

We intentionally refrain from incorporating complex robustness or fairness-enhancing techniques in order to preserve the experimental purity. By observing the phenomena under a minimal and natural setup, we aim to capture the underlying dynamics of model bias as it would emerge in practical, real-world conditions. This design choice avoids confounding methodological effects and lays the groundwork for building more generalizable and interpretable fairness mechanisms in the future.

We assume our findings could inspire new directions that promote fairness without requiring explicit group labels or sophisticated architectural modifications, particularly in real-world settings where group annotations are often ambiguous, incomplete or unavailable.

5 Related Works

5.1 Fairness in Natural Language Processing

Early investigations uncovered social biases in pre-trained language models, manifesting as gendered occupation stereotypes and demographic sentiment skews. Debiasing efforts have ranged from data augmentation to adversarial debiasing and fairness-

Table 3: Performance (Accuracy) and fairness (SPD) of BERT-base and DeBERTa models under fairness-agnostic adversarial training based on adversarial examples from different attackers. Values are presented as mean (standard deviation).

Adversarial	FAKE NEWS		NEWSMTSC		APP REVIEWS	
Training	Accuracy ↑	|SPD| ↓	Accuracy ↑	|SPD| ↓	Accuracy ↑	|SPD| ↓
BERT-base (Adversarial Training)						
Baseline	0.97 (0.01)	-0.576 (0.015)	0.89 (0.02)	0.190 (0.010)	0.87 (0.01)	-0.134 (0.017)
DeepWordBug	0.93 (0.02)	-0.502 (0.022)	0.64 (0.04)	0.130 (0.009)	0.67 (0.02)	0.062 (0.021)
BAE	0.97 (0.07)	-0.531 (0.013)	0.66 (0.02)	0.126 (0.012)	0.65 (0.05)	0.039 (0.016)
PWWS	0.89 (0.02)	-0.456 (0.006)	0.62 (0.04)	0.113 (0.020)	0.70 (0.02)	0.036 (0.008)
TextFooler	0.86 (0.06)	-0.396 (0.026)	0.68 (0.02)	0.119 (0.009)	0.71 (0.03)	0.038 (0.008)
Qwen3-8B	0.88 (0.02)	-0.402 (0.018)	0.69 (0.08)	0.106 (0.012)	0.74 (0.06)	0.032 (0.011)
ChatGPT-4o	0.89 (0.05)	-0.380 (0.022)	0.72 (0.05)	0.102 (0.008)	0.70 (0.07)	0.036 (0.017)
DeBERTa (Adversarial Training)						
Baseline	0.98 (0.01)	-0.550 (0.014)	0.91 (0.02)	0.180 (0.011)	0.89 (0.01)	-0.120 (0.010)
DeepWordBug	0.95 (0.02)	-0.478 (0.017)	0.67 (0.01)	0.126 (0.012)	0.70 (0.02)	0.066 (0.005)
BAE	0.98 (0.01)	-0.459 (0.010)	0.72 (0.04)	0.134 (0.010)	0.69 (0.03)	0.060 (0.012)
PWWS	0.91 (0.02)	-0.457 (0.008)	0.64 (0.16)	0.112 (0.011)	0.70 (0.01)	0.032 (0.011)
TextFooler	0.90 (0.01)	-0.429 (0.013)	0.70 (0.01)	0.108 (0.012)	0.72 (0.03)	0.040 (0.018)
Qwen3-8B	0.93 (0.04)	-0.379 (0.006)	0.73 (0.05)	0.114 (0.008)	0.73 (0.02)	0.044 (0.014)
ChatGPT-4o	0.94 (0.02)	-0.367 (0.014)	0.71 (0.02)	0.107 (0.015)	0.75 (0.01)	0.032 (0.008)

aware fine-tuning (e.g., (Tan et al., 2020)).

5.2 Textual Adversarial Attacks

Token-level substitution methods such as DeepWordBug (Gao et al., 2018a), TextFooler (Jin et al., 2020), BAE (Garg and Ramakrishnan, 2020), and the TextAttack framework (Morris et al., 2020) demonstrated that state-of-the-art classifiers (e.g., BERT) are vulnerable to small, human-imperceptible perturbations. Subsequent work introduced *universal triggers* that operate input-agnostically (Wallace et al., 2019) and prompt-based LLM attacks that craft fluent adversarial examples at scale.

5.3 Fairness under Adversarial Attacks

Research that explicitly couples adversarial robustness with fairness remains scarce. Mehrabi et al. (2021) proposed data-poisoning *fairness attacks* that exacerbate demographic disparity, while Zhang et al. (2023) systematically quantified group- and individual-fairness degradation under gradient-based attacks. Intriguingly, Tian et al. (2023) observed that certain sentiment-biases are mitigated by adaptive attacks, suggesting a complex bias–robustness interplay. Back-door style triggers (BadFair (Xue et al., 2024), **FABLE** (Liang et al., 2023)) show that a model can appear fair under benign inputs yet discriminate when activated.

5.4 Adversarial Defenses and Fairness

Classical adversarial training improves robustness but often worsens group disparity, e.g., Xu et al. (2021) formalized this robustness–fairness gap in vision, motivating analogous studies in NLP. Recent work augments adversarial training with fairness regularizers or re-weighting schemes, yet comprehensive evaluations across multiple tasks and attack types remain open.

6 Conclusion

In this work, we investigate how fairness-agnostic adversarial attacks interact with fairness in PLMs. We uncover a counterintuitive yet consistent phenomenon that adversarial attacks could reduce model prediction bias. This emergent fairness improvement is pronounced for semantically perturbations which tend to dilute spurious correlations with sensitive attributes. Building on this insight, we demonstrate that adversarial training can also be leveraged to promote PLM fairness. Even without fairness-specific loss terms, PLMs trained on diverse adversarial examples consistently improve SPD while regaining accuracy.

Limitations

While our study sheds light on the interplay between adversarial attacks and fairness in NLP models, several limitations must be acknowledged:

Dataset Representativeness. The datasets employed, i.e., App Reviews, News Sentiment Classification, and Fake News Detection, may not fully encapsulate the breadth and complexity of real-world biases. These datasets, while valuable, might lack the diversity and nuanced representation of sensitive attributes present in broader, more heterogeneous corpora. Consequently, the generalizability of our findings to diverse real-world scenarios may be constrained.

Scale of Experiments. Our experimental framework, though rigorous, is limited in scale. The scope of models, attack strategies, and defense mechanisms explored does not exhaust the vast landscape of possibilities in adversarial robustness and fairness research. This limitation may affect the comprehensiveness of our conclusions and their applicability to a wider array of models and attack vectors.

Fairness Metric Constraints. We focus exclusively on SPD as our fairness criterion. While SPD is intuitive, measuring only the difference in positive-prediction rates between groups, it does not capture other aspects of fairness such as equalized odds (differences in error rates), individual fairness (treatment of similar instances), or intersectional subgroup disparities. Thus, relying solely on SPD may obscure other fairness-related harms and trade-offs.

References

Rajas Bansal. 2022. A survey on bias and fairness in natural language processing. *arXiv preprint arXiv:2204.09591*.

Rachel K. E. Bellamy, Kuntal Dey, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, Seema Nagar, Karthikeyan Natesan Ramamurthy, John T. Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Kush R. Varshney, and Yunfeng Zhang. 2019. AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development*, 63(4/5):4:1–4:15.

Daniel Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder for english. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174, Brussels, Belgium. Association for Computational Linguistics.

Junyi Chai, Taeuk Jang, Jing Gao, and Xiaoqian Wang. 2023. On the adversarial attack and defense of fairness. In *OpenReview*. URL: https://openreview.net/forum?id=IrZTJ7t2GW.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*.

Fatma Elsafoury and Stamos Katsigiannis. 2023. On bias and fairness in nlp: Investigating the impact of bias and debiasing in language models on the fairness of toxicity detection. *arXiv preprint arXiv:2305.12829*.

Isabel O Gallegos, Ryan A Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K Ahmed. 2024. Bias and fairness in large language models: A survey. *Computational Linguistics*, 50(3):1097–1179.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018a. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 50–56. IEEE.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018b. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *IEEE S&P Workshops*.

Siddhant Garg and Goutham Ramakrishnan. 2020. Bae: BERT-based adversarial examples for text classification. In *Proceedings of EMNLP*.

Giovanni Grano, Andrea Di Sorbo, Francesco Mercaldo, Corrado A. Visaggio, Gerardo Canfora, and Sebastiano Panichella. 2017. Android apps and user feedback: A dataset for software evolution and quality improvement. In *Proceedings of the 2nd ACM SIGSOFT International Workshop on App Market Analytics (WAMA)*, pages 8–11. ACM.

Felix Hamborg and Karsten Donnay. 2021. Newsmtsc: (multi-)target-dependent sentiment classification in news articles. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, Virtual.

Moritz Hardt, Eric Price, and Nathan Srebro. 2016. Equality of opportunity in supervised learning. In *NeurIPS*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations (ICLR)*.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is BERT really robust? a strong baseline for natural language attack on text classification and entailment. In *AAAI*.

Yueqing Liang, Lu Cheng, Ali Payani, and Kai Shu. 2023. Beyond detection: Unveiling fairness vulnerabilities in abusive language models. *arXiv preprint arXiv:2311.09428*.

Yi Liu, Gelei Deng, Yuekang Li, Kailong Wang, Zihao Wang, Xiaofeng Wang, Tianwei Zhang, Yepang Liu, Haoyu Wang, Yan Zheng, and 1 others. 2023. Prompt injection attack against llm-integrated applications. *arXiv preprint arXiv:2306.05499*.

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*.

Ning Lu, Shengcai Liu, Rui He, Qi Wang, Yew-Soon Ong, and Ke Tang. 2023. Large language models can be guided to evade ai-generated text detection. *arXiv preprint arXiv:2305.10847*.

Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *NeurIPS*.

Ninareh Mehrabi, Fred Morstatter, Nino Saxena, Kristina Lerman, and Aram Galstyan. 2021. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35.

Johnathan S. Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP. In *EMNLP Demos*.

Chen Qu, Liu Yang, Minghui Qiu, W Bruce Croft, Yongfeng Zhang, and Mohit Iyyer. 2019. Bert with history answer embedding for conversational question answering. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pages 1133–1136.

Navid Rekabsaz, Simone Kopeinik, and Markus Schedl. 2021. Societal biases in retrieved contents: Measurement framework and adversarial mitigation of bert rankers. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 306–316.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *ACL*.

Rickard Stureborg, Dimitris Alikaniotis, and Yoshi Suhara. 2024. Large language models are inconsistent and biased evaluators. *CoRR*.

Samson Tan, Shafiq Joty, Min-Yen Kan, and Richard Socher. 2020. It's morphin' time! combating linguistic discrimination with inflectional perturbations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Jiachen Tian, Sicheng Chen, Xiaonan Zhang, and Zhiyong Feng. 2023. Reducing sentiment bias in pretrained sentiment classification via adaptive gumbel attack. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019. Universal adversarial triggers for attacking and analyzing NLP. In *Proceedings of EMNLP*.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and 1 others. 2024. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9440–9450.

Wei Xu, Yisen Wang, Di Wu, and Tongliang Wu. 2021. To be robust or to be fair: Towards fairness in adversarial training. In *International Conference on Machine Learning*, pages 11492–11501.

Jiaqi Xue, Qian Lou, and Mengxin Zheng. 2024. Badfair: Backdoored fairness attacks with group-conditioned triggers. In *Findings of EMNLP*.

Tao Zhang, Hao Hu, Xiangyu Liu, and Panpan Zheng. 2023. Revisiting model fairness via adversarial examples. *Knowledge-Based Systems*.

Wanzheng Zhu, Hongyu Gong, Rohan Bansal, Zachary Weinberg, Nicolas Christin, Giulia Fanti, and Suma Bhat. 2021. Self-supervised euphemism detection and identification for content moderation. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 229–246. IEEE.

A Why Statistical Parity Difference?

We adopt SPD as our sole fairness metric in this study due to both conceptual clarity and experimental focus:

- SPD provides a group-level measure that directly reflects whether different demographic groups receive unequal rates of positive outcomes. This aligns naturally with our empirical setting, where the sensitive attributes (e.g., gender, region, political affiliation) are used to partition instances into discrete subgroups, and the goal is to detect disparities in classification rates across these groups.

- Unlike criteria such as Equalized Odds, SPD does not rely on the true label y, which allows it to remain meaningful even when adversarial perturbations introduce label-flipping noise or semantic ambiguity. This is particularly critical in our setting, where input-label consistency is only weakly guaranteed under attack.
- SPD has become a widely used baseline in the fairness literature (Hardt et al., 2016; Mehrabi et al., 2021), offering strong interpretability and community familiarity. We leave the investigation of complementary criteria, such as Equalized Odds or individual fairness, to future work that explores finer-grained alignment between model calibration and fairness constraints.

B LLM-Prompted Adversarial Attacks

Recent advances in large language models (LLMs) enable adversarial examples to be synthesized via instruction-following prompts, bypassing the need for handcrafted heuristics or gradient access. We leverage this generative ability to construct fluent, semantically faithful adversarial examples that preserve task semantics but flip model predictions.

Problem Formulation. Let $\mathbf{x}$ denote a clean input with ground-truth label y, and f_θ the victim model. The goal is to produce a perturbed sentence $\tilde{\mathbf{x}}$ satisfying the following criteria:

1. **Semantic Preservation:** $\mathrm{sim}(\tilde{\mathbf{x}}, \mathbf{x}) \geq \tau$, where $\tau = 0.9$ (measured by USE cosine similarity).
2. **Prediction Flip:** $f_\theta(\tilde{\mathbf{x}}) \neq f_\theta(\mathbf{x})$.
3. **Token Budget:** $\|\tilde{\mathbf{x}} - \mathbf{x}\|_0 \leq \delta$, where δ is 15% of the original token count.

Prompt Construction. We design a task-agnostic adversarial prompt template as follows:

```
Below is a text labeled as [LABEL]. Please rephrase it into a fluent sentence that maintains its meaning, but would be classified as [OPPOSITE LABEL] by a sentiment classifier. Keep token overlap below 85%.
```

The instruction is paired with the original input $\mathbf{x}$ and passed to an LLM such as GPT-4o or Qwen3 for generation.

Algorithm 1: LLM-Prompted Adversarial Generation with Retry Budget

Input: LLM engine $\mathcal{L}$, victim model f_θ, similarity threshold τ, retry budget B
Output: Adversarial dataset $\mathcal{D}_{\text{adv}}$

foreach $(\mathbf{x}_i, y_i) \in \mathcal{D}_{clean}$ **do**
  for $j = 1$ **to** B **do**
    Construct prompt $\mathcal{P}_i^{(j)}$ from $\mathbf{x}_i$ and y_i;
    Query LLM: $\tilde{\mathbf{x}}_i^{(j)} \leftarrow \mathcal{L}(\mathcal{P}_i^{(j)})$;
    if $\mathrm{sim}(\tilde{\mathbf{x}}_i^{(j)}, \mathbf{x}_i) \geq \tau$ ***and*** $f_\theta(\tilde{\mathbf{x}}_i^{(j)}) \neq f_\theta(\mathbf{x}_i)$ **then**
      Add $(\tilde{\mathbf{x}}_i^{(j)}, y_i)$ to $\mathcal{D}_{\text{adv}}$;
      break;
return $\mathcal{D}_{adv}$

Attack Algorithm. Since LLMs are stochastic and may not yield a successful adversarial sample on the first attempt, we introduce a retry budget B (default $B = 50$) that limits the number of attempts per input. Given a clean dataset $\mathcal{D}_{\text{clean}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, the attack proceeds as:

Unlike gradient-based or rule-based attacks, LLM-prompted perturbations have several advantages. First, they produce natural language outputs that more closely resemble real-world misuse or adversarial behavior. Second, they can bypass syntactic or lexical filters by leveraging diverse paraphrasing patterns. Finally, they offer a more scalable and transferable attack strategy that generalizes across different domains and models.

We filter generations with:

- Universal Sentence Encoder (USE) similarity ≥ 0.9,
- Token overlap $\leq 85\%$ with the original sentence,
- Length difference within $\pm 10\%$ of the source.

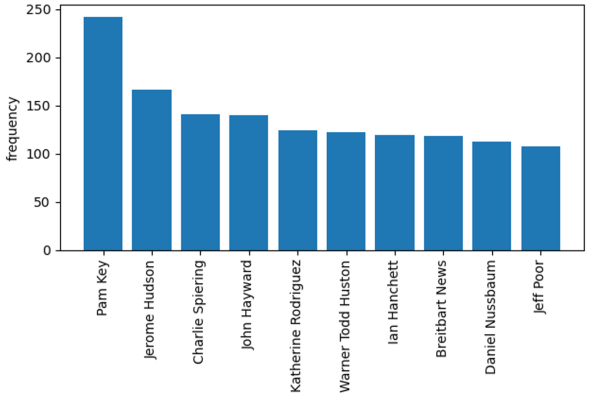
We denote these adversarial splits as $\mathcal{D}_{\text{adv}}^{\text{LLM}}$ and use them in Phase 2 and Phase 3 evaluations. Retry failures are excluded from $\mathcal{D}_{\text{adv}}^{\text{LLM}}$, preserving high-quality, successful perturbations only.

C Sensitive Attribute Visualization

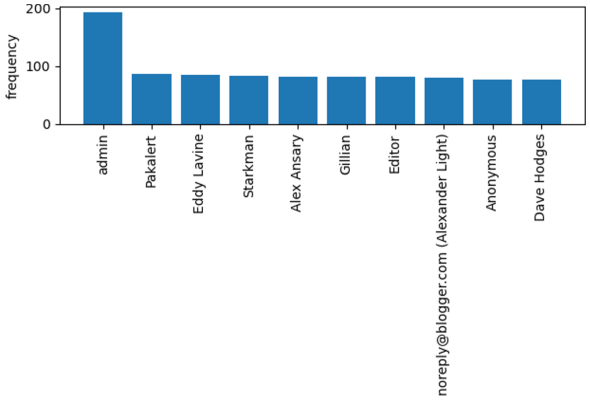
To better understand the distributional patterns of sensitive attributes across datasets, we visualize the top-10 most frequent named entities, gendered

pronouns, and app identifiers associated with each class. These histograms help clarify which groups dominate the corpora, and which subgroups may constitute *non-privileged* populations due to under-representation or unfavorable sentiment association. All tokens are lowercased before counting, and grouped using heuristics described in Table 1.

C.1 FAKE NEWS Dataset



(a) Frequencies of Groups in True News

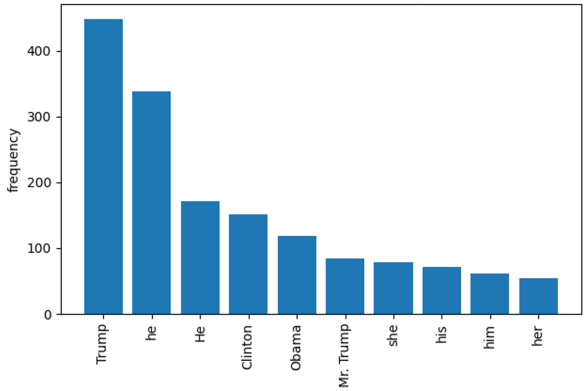


(b) Frequencies of Groups in Fake News

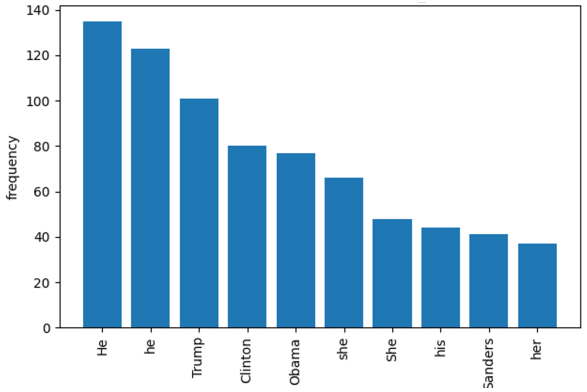
Figure 2: Top-10 author frequency in FAKE NEWS. True news articles (left) are primarily authored by traceable individuals affiliated with recognized outlets (e.g., *Pam Key*, *Jerome Hudson*, *Breitbart News*). In contrast, fake news articles (right) are disproportionately authored by anonymous or obscure identities (e.g., *admin*, *Editor*, *Anonymous*), reflecting lower accountability and higher volatility in source attribution.

As shown in Figure 2, true news articles tend to be written by consistently identified individuals, suggesting more editorial traceability. In contrast, fake news samples show a marked dominance of vague or pseudo-authorial entities like *admin* or *Editor*, which may reflect a lack of editorial oversight. This discrepancy in authorship distribution may propagate bias during training: models may implicitly associate anonymity or obscure sources with deception, potentially leading to over-reliance on author fields rather than semantic content.

C.2 NEWSMTSC Dataset



(a) Frequencies of Groups in Positive News

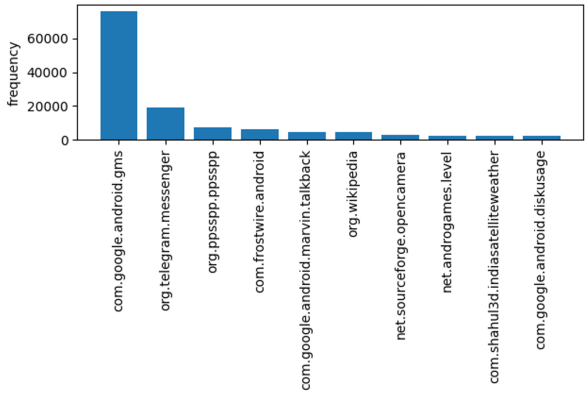


(b) Frequencies of Groups in Negative News

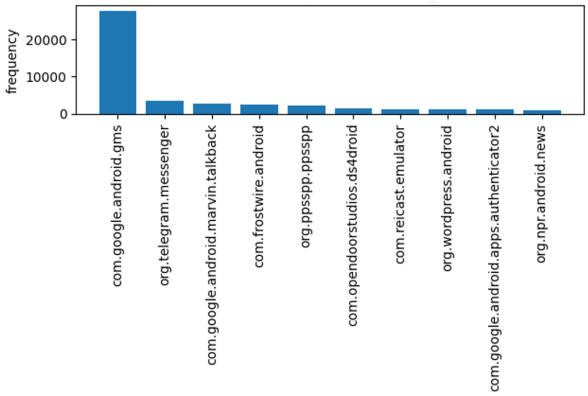
Figure 3: Entity-level group frequencies in NEWSMTSC. Positive sentiment samples (top) show higher mentions of male pronouns and right-leaning figures (e.g., *Trump*, *he*), while negative sentiment samples (bottom) display a broader distribution including more references to female pronouns and left-leaning or marginalized individuals (e.g., *she*, *Sanders*, *her*).

Figure 3 reveals a polarity–identity association: the top-ranked entities in positive sentiment samples include dominant male and politically conservative figures such as *Trump*, *he*, and *Mr. Trump*, while negative samples show a comparatively more balanced distribution across gendered pronouns (*she*, *her*, *his*) and opposition-aligned actors (e.g., *Sanders*). This entity-sentiment co-occurrence imbalance suggests that models trained on such data may internalize unintended group-valence correlations, posing fairness risks for downstream predictions involving gender or political orientation.

C.3 APP REVIEWS Dataset



(a) Frequencies of Groups in Positive Reviews



(b) Frequencies of Groups in Negative Reviews

Figure 4: App identifier distribution in APPREVIEWS. Positive reviews (top) are dominated by official packages such as `com.google.android.gms`, while negative reviews (bottom) more frequently mention accessibility tools (e.g., `talkback`), emulators, and fringe utility apps (e.g., `reicast.emulator`, `ds4droid`).

Figure 4 reveals clear usage and sentiment asymmetry across app types. Positive reviews overwhelmingly focus on widely deployed system apps (e.g., `com.google.android.gms`, `org.telegram.messenger`), suggesting higher user satisfaction with mainstream services. In contrast, negative reviews mention niche or accessibility-related packages, such as `com.google.android.marvin.talkback`, emulators like `reicast.emulator`, or utilities for advanced users. This shift indicates a potential fairness risk: if models learn sentiment signals that correlate with app identity rather than user experience, certain app categories—particularly those related to accessibility or non-standard use cases—may be disproportionately penalized in downstream classification or moderation systems.

C.4 Discussion

These visualizations provide empirical context for our fairness assessments. In all datasets, we observe attribute imbalance or sentiment association skew that could result in unfair treatment by models. This motivates our use of group-based metrics such as SPD, as well as token attribution analysis to further track how models learn and react to such disparities.

D SHAP Visualization Cases

To better understand how model decisions are shaped by lexical cues, and how these cues interact with demographic correlations, we apply SHAP (Lundberg and Lee, 2017) to interpret token-level attribution patterns. These visualizations offer insight into how salient phrases (e.g., political names, sentiment-laden terms, device identifiers) can dominate predictions and reveal potential fairness risks. However, we note that SHAP reflects local associations and should not be interpreted as providing definitive causal explanations.

FAKE NEWS. As shown in Figure 5, model decisions in the fake news detection task are influenced by named entities and emotionally charged words.

In the first example, tokens such as *The New York Times* and *backlash* receive high negative SHAP values, aligning with a fake label. This suggests that the model may associate mentions of certain media outlets and emotionally negative framing with inauthenticity. Importantly, more neutral or factual tokens receive little attribution, implying that predictions may hinge on high-frequency, ideologically marked features.

The second example describes a factual event involving Mike Pence. SHAP values attribute importance to contextually generic terms like *Twitter*, *LaGuardia*, and *Thursday*, which coincide with a correct real label. These attributions suggest reliance on structural cues (e.g., time or platform mentions), which may serve as weak proxies for credibility but are not necessarily semantically diagnostic of veracity.

The third case is misclassified as fake, largely due to repeated mentions of *The New York Times* in a civil rights context. This echoes earlier distributional analyses (Figure 2) showing that certain entities disproportionately occur in articles labeled as fake, potentially due to data imbalance or annotator bias. These examples underscore the model's sensitivity to politically and culturally salient terms,

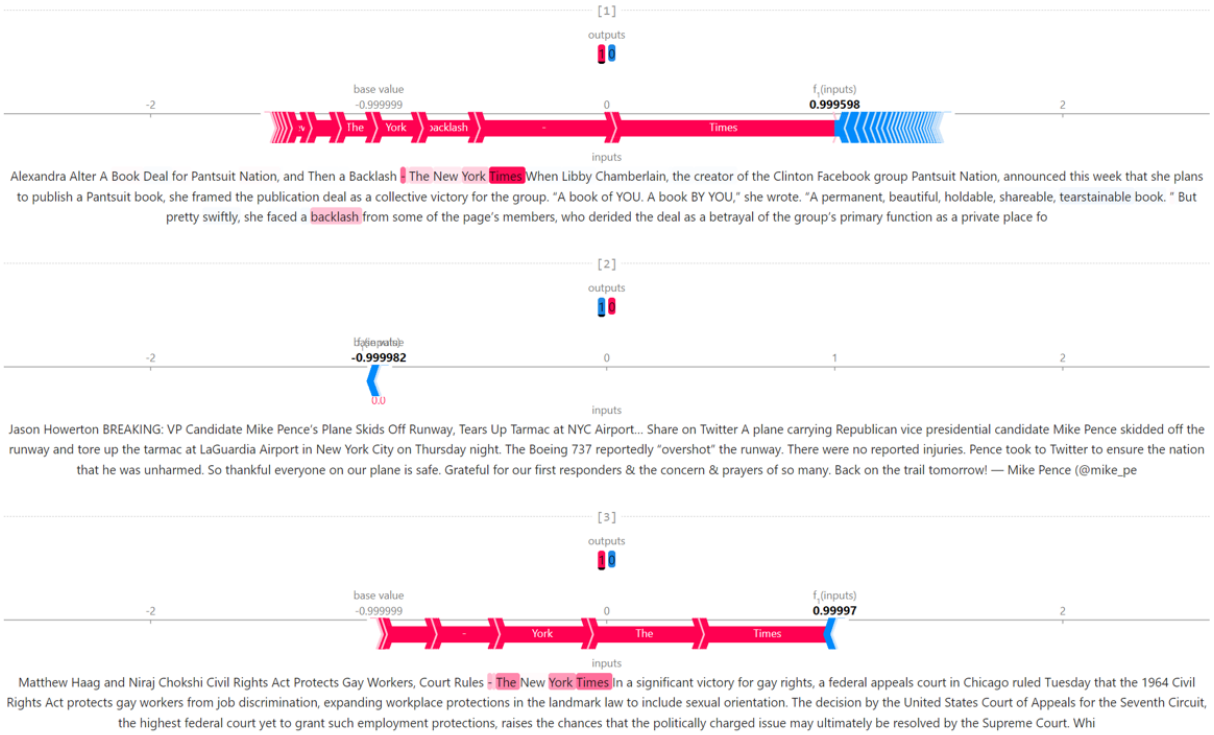


Figure 5: SHAP token attributions for three representative examples from the FAKE NEWS dataset. (Top) Tokens such as *The New York Times* and emotionally charged words like *backlash* receive strong negative attributions, correlating with predictions of the fake class. (Middle) A factually neutral report involving public figures (e.g., *Mike Pence*) is classified as real, with contextually neutral tokens (e.g., dates, platforms) receiving positive contributions. (Bottom) A civil rights story is misclassified as fake due to high attribution to recurring media mentions, suggesting sensitivity to media identity cues. These examples illustrate how token-level associations, shaped by distributional biases, may influence predictions in unintended ways. However, attribution reflects correlation rather than causal reasoning.

which may lead to systematic misclassification and fairness issues.

NEWSMTSC. In Figure 6, we observe similar attribution patterns in sentiment prediction on news text.

In the first case, the model assigns high importance to "racist" within a quoted statement, suggesting a lack of ability to distinguish between authorial opinion and reported speech. This may lead to misinterpretation of neutral reporting as opinionated content.

In the second case, emotionally expressive terms like "formidable" and "rarely" receive dominant attributions. While such terms often correlate with sentiment, their presence in quoted material calls into question the model's ability to separate subjective description from editorial stance.

In the third example, tokens related to geopolitical conflict, such as "Putin", "cyber attacks", and "hack", receive high negative attribution. While these may indeed appear in negative contexts during training, their strong influence raises concerns about regional or political bias, particularly if such patterns correlate with identity cues.

Overall, these examples highlight the need for sentiment models to disambiguate emotional tone from demographic or framing artifacts, particularly when applied to journalistic content.

APP REVIEWS. As shown in Figure 7, sentiment models in user reviews often rely on highly polarized expressions or common template phrases.

In the top case, the model's prediction is dominated by the phrase "Works great", with little contribution from tokens describing context or functionality. This suggests that short emotional phrases may disproportionately drive positive sentiment classifications.

The middle example shows a similar pattern on the negative side, where "Does not work" and "brings up" are highly weighted. However, the underlying issue described is a minor UI quirk, indicating that the model may overreact to sentiment

Figure 6: SHAP token attributions for three representative examples from the NEWSMTSC dataset. (Top) The term “racist” in a quoted statement receives high attribution, suggesting that the model may conflate reported speech with authorial stance. (Middle) Emotional descriptors like “formidable” and intensifiers like “rarely” dominate attribution, reflecting lexical bias even in quoted opinions. (Bottom) Entities like “Putin” and “cyber attacks” contribute strongly to negative sentiment predictions, indicating potential sensitivity to geopolitical framing. These cases suggest that sentiment models may over-rely on high-salience lexical cues without disambiguating source attribution, potentially reinforcing demographic or regional skew.

templates without deeper semantic understanding.

The third case involves a noisy input with grammatical inconsistencies. Despite this, the model correctly predicts a negative label, showing some robustness to informal language. Interestingly, device mentions such as “Note 4” and unintelligible terms like “Mah” receive mild positive attribution. This may reflect weak or spurious correlations between device types and review sentiment.

We note that app identifiers (e.g., `currentwidget`) receive little attribution across cases, but their categorical distribution may still introduce latent bias, particularly if some app categories are overrepresented in specific user populations. Systematic evaluation of these correlations remains important for fairness auditing.

Overall. We observe that, across domains, model predictions are shaped by a small subset of highly salient tokens, some of which coincide with sensitive attributes or demographic proxies. After adversarial training, we find that SHAP attribution to such tokens (e.g., “Trump”, “she”) becomes more diffuse or attenuated, suggesting that fine-tuning can disrupt reliance on spurious lexical cues. While this does not guarantee fairness improvements, it offers a mechanism for mitigating token-level bias. SHAP thus serves as a valuable interpretability tool for revealing lexical shortcuts and their alignment with group disparities, though attribution patterns should be interpreted with appropriate caution regarding causality and generalization.

E Compute Resources

All experiments were conducted on a private cluster equipped with NVIDIA RTX4090 GPUs. We use a single GPU for model training and evaluation, and up to two GPUs in parallel for adversarial example generation using LLM-based attacks. We

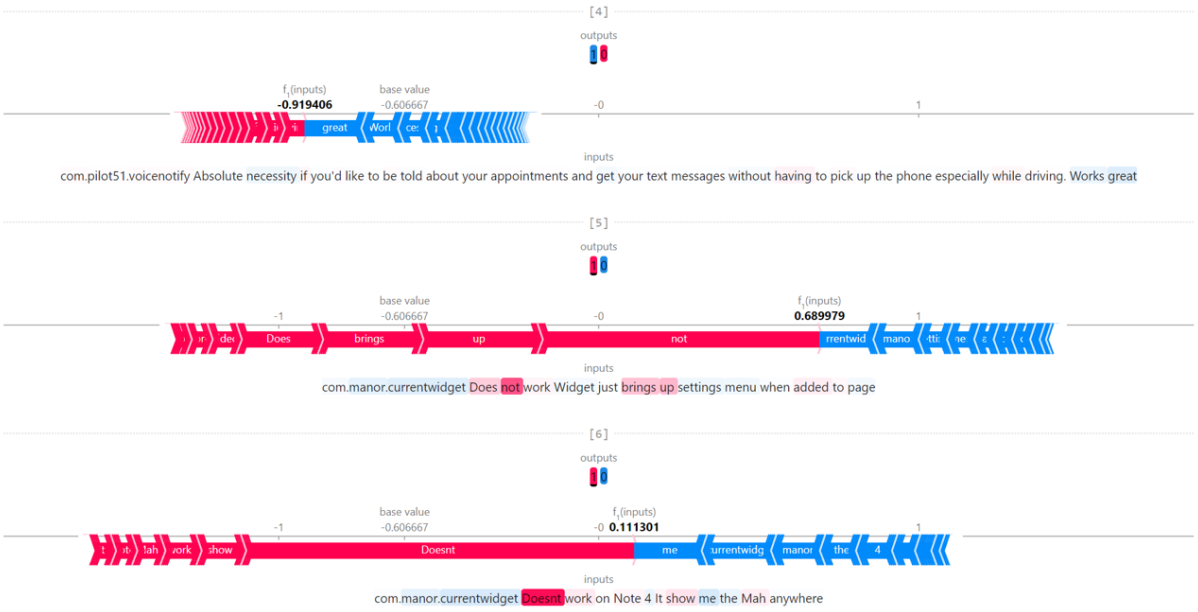


Figure 7: SHAP token attributions for three APP REVIEWS examples. (Top) Positive prediction dominated by "Works great", while surrounding context tokens receive negligible attribution. (Middle) Negative prediction driven by "Does not work" and "brings up", even though the complaint pertains to UI behavior rather than functionality. (Bottom) A grammatically noisy input is correctly predicted as negative, with "Doesnt" driving attribution, while unrelated tokens (e.g., "Note 4") show mild positive influence. These examples highlight the model's tendency to focus on common sentiment patterns and emotional templates, with limited accounting for structural or domain-specific nuances.

describe the approximate computational resources occupation as follows:

- **Training Time:** Fine-tuning each classifier takes approximately 0.5 hour per dataset. Robust models require 2–3× longer due to on-the-fly adversarial generation.
- **Adversarial Generation:** Token-level attacks (e.g., TEXTFOOLER, BAE) require ∼30 seconds per sample. LLM-prompted paraphrase attacks (via GPT-4o API) consume roughly 20–40 seconds per query, with many queries required to achieve successful attacks.
- **Fairness Evaluation:** SHAP explanation and group attribution computations are performed on 1,000 random samples per experiment, totaling 5–10 minutes per configuration.

Overall, our end-to-end experiments across three datasets, six attack recipes, and two model backbones consumed approximately **350 GPU-hours** in total.

F Ethics Statement

This work investigates fairness and robustness in natural language models, with a focus on how adversarial perturbations affect group-level disparities. All datasets used are publicly available, and sensitive or personally identifiable information has been either removed or anonymized during preprocessing.

We affirm that this research is conducted with the explicit goal of identifying and mitigating algorithmic bias, not reinforcing it. Any patterns of bias or discrimination shown in examples or model outputs do not reflect the authors' views or endorsement, but rather serve to illustrate potential vulnerabilities in real-world deployed systems. Furthermore, all adversarial examples and fairness analyses are designed to improve model accountability and are not intended for misuse.

We oppose all forms of linguistic, social, or demographic prejudice in machine learning systems. As part of our long-term goal, we aim to advance the development of equitable and trustworthy AI. We encourage future work to engage with affected communities, domain experts, and ethicists to interpret fairness metrics within broader sociotechnical contexts.