LANGUAGE MODELS THAT WALK THE TALK: A FRAMEWORK FOR FORMAL FAIRNESS CERTIFICATES

Anonymous authors
Paper under double-blind review

ABSTRACT

As large language models become integral to high-stakes applications, ensuring their robustness and fairness is crucial. Despite their success, large language models remain vulnerable to adversarial attacks, where small perturbations can alter model predictions. In language models, perturbations such as synonym substitutions also pose risks in fairness- and safety-critical areas. Formal verification is a viable path to certify fairness; however, its application to large language models remains limited. This work presents a holistic verification framework to certify the fairness of language models by formalizing the verification problem within the embedding space, with a focus on ensuring gender fairness and consistent outputs across different gender-related terms. Furthermore, we extend this methodology to toxicity detection to offer formal guarantees that adversarially manipulated toxic inputs are consistently detected and appropriately censored, thereby ensuring the reliability of moderation systems.

1 INTRODUCTION

As large language models become integral to academia and industry, including in high-stakes decision-making and safety-critical domains, ensuring their robustness and trustworthiness is crucial (García, 2023; Huang et al., 2024). While their transformer-based architecture with attention mechanisms has driven widespread adoption (Vaswani et al., 2017), concerns about robustness and safety persist, especially in tasks like fairness enforcement and content moderation.

A key challenge is their susceptibility to adversarial perturbations (Goodfellow et al., 2015a; Szegedy et al., 2014), where small perturbations, such as synonym substitutions or paraphrasing, can manipulate model behavior (Alzantot et al., 2018; Behjati et al., 2019; Li et al., 2019; Liang et al., 2018; Jin et al., 2020; Hsieh et al., 2019). While such perturbations may be adversarially crafted in settings like toxicity detection (Bespalov et al., 2023; Kruschwitz & Schmidhuber, 2024), they can also arise naturally in fairness-critical tasks such as salary prediction with gender bias, highlighting the need for robust gender bias mitigation (Tumlin et al., 2024). Natural language processing tasks operate in discrete input spaces, where verifying robustness by enumerating all synonym substitutions quickly becomes combinatorially infeasible. For example, a small sentence with just 20 tokens drawn from a 32k-token (BERT-size) vocabulary already has up to $32\text{k}^{20} \approx 10^{90}$ adversarial sentences, exceeding the number of atoms in the observable universe (Kumar et al., 2024). This makes adversarial robustness verification particularly challenging (Bonaert et al., 2021; Huang et al., 2019).

Contributions. We aim to circumvent this problem using formal verification over continuous spaces. In summary, our main contributions are: (i) The formalization of robustness to sensitive terms in the embedding space. (ii) A training objective to obtain language models adhering to this formalization. (iii) An extensive evaluation demonstrating the practicability of our approach.

2 A FRAMEWORK FOR FAIRNESS CERTIFICATION

This section presents our certification framework, which we also illustrate in Figure 1 with gender fairness verification in a salary prediction task. Toxicity detection follows an analogous process.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

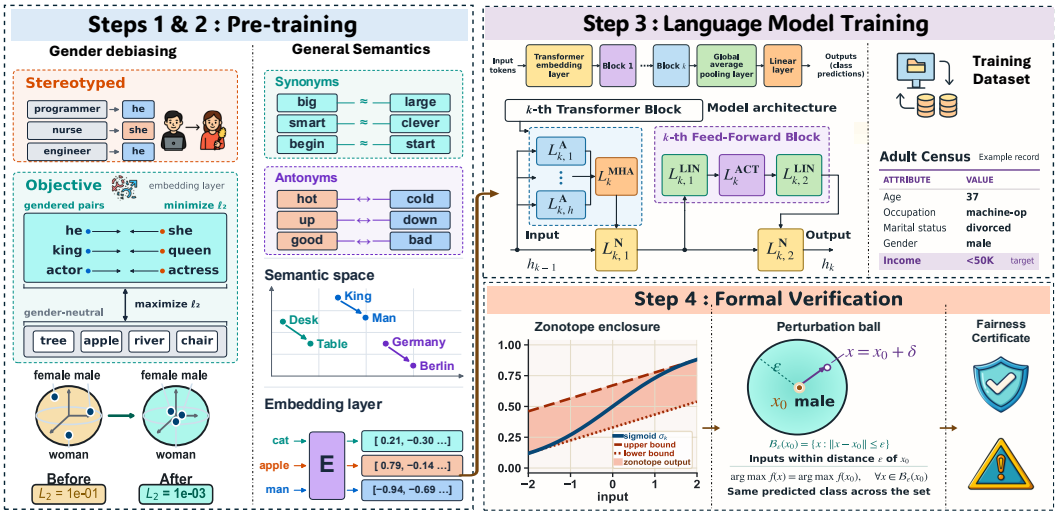


Figure 1: Overview of the framework explained in Section 2 involving pre-training (Steps 1 and 2), model training (Step 3), and formal verification (Step 4). The transformer architecture in Step 3 is from Vaswani et al. (2017).

In short, the required steps are: **Step 1: Gender-debias pre-training** (Section 3). To ensure proximity of gender-related terms in the embedding space, we pre-train the embedding layer on curated gendered word pairs, encouraging gendered pairs to be close while separating them from gender-neutral words using ℓ_2 distance. **Step 2: General semantic pre-training** (Section 3). We further pre-train the embeddings on WordNet synonym/antonym pairs to capture general semantic and syntactic relationships. **Step 3: Language model training** (Section 4). The pre-trained embedding layer is integrated into a transformer-based language model and frozen while the model is trained on a fairness-critical gender bias mitigation task. **Step 4: Formal verification** (Section 5). We formally verify fairness and robustness to synonym substitutions by perturbing the input embedding space and analyzing the resulting changes in model outputs.

Subsequently, we detail our proposed approach for each step along with the required background.

3 Pre-Training the Embedding Space

Background. Since many AI-automated decisions can significantly impact people's lives, it is important to ensure fairness regarding, e.g., which individuals will receive jobs and loans, and to assess and improve the ethics of the decisions made by these automated systems (Pessach & Shmueli, 2022; Mehrabi et al., 2021). A straightforward approach to promoting fairness in machine learning, defined as the absence of bias and discrimination, is to preclude the use of sensitive attributes during model training and inference (Lepri et al., 2018; Calders & Verwer, 2010; Kamiran et al., 2010; Barocas & Selbst, 2016). For instance, in designing a race-blind or gender-blind decision-making system, one might exclude sensitive attributes from the feature set entirely (Lepri et al., 2018). Alternatively, one can represent them with similar embeddings to share a common semantic meaning and thus be treated as equivalent by the model (Bolukbasi et al., 2016).

Word embedding models map words to vectors based on the assumption that words appearing in similar contexts share similar meanings (Pessach & Shmueli, 2022; Mikolov et al., 2013a;b). They capture semantic relationships and are widely used in natural language processing tasks (Pessach & Shmueli, 2022). However, studies have shown that embeddings can encode biases (Bolukbasi et al., 2016; Brunet et al., 2019; Caliskan et al., 2017a; Zhao et al., 2018); for instance, `programmer` may appear closer to `male` than `female`, raising discrimination concerns in AI-based decisions.

Designing embedding spaces. This work employs training a task-specific embedding layer from scratch by clustering gender-related words and distancing unrelated ones. This offers greater transparency and enables direct mitigation of gender bias during embedding layer training by mapping

all gender-related words to a cohesive region in the embedding space fit for formal verification. Analogously, this framework can also be applied to adversarial substitutions in the context of toxicity detection, ensuring robust censorship of inappropriate content such as profanity, hate speech, and abusive or otherwise harmful language. Prior work supports this, as, e.g., embeddings trained on task-specific data can outperform static embeddings (Bogale Gereme & Zhu, 2021). Similar practices are seen in domains such as fake news detection (Liu & Wu, 2018), highlighting the value of task-specific embedding spaces to improve performance and propagate semantic knowledge (Zhuang et al., 2021).

In more detail, a good data representation promotes intra-class compactness and inter-class separability (Khosla et al., 2020; Jaiswal et al., 2020; HaoChen et al., 2021; Wang & Liu, 2021). As contrastive learning has proven highly effective in enhancing representations across NLP, visual computing, and beyond (Jaiswal et al., 2020; Caron et al., 2020; He et al., 2020; Chen et al., 2020a; HaoChen et al., 2021; Saunshi et al., 2019; Chen et al., 2020b;e; Chopra et al., 2005), we employ a contrastive loss that separates similar and dissimilar word pairs in the embedding space to learn semantically meaningful embeddings (Figure 1, Steps 1 & 2):

Let $h_1, h_2 \in \mathbb{R}^d$ denote the embeddings of a pair of input words, and let $d = \|h_1 - h_2\|_2$ be the Euclidean distance between them. Given a binary label $y \in \{0, 1\}$ indicating whether the pair is dissimilar ($y = 1$) or similar ($y = 0$), the contrastive loss is defined as:

$$\mathcal{L}(y, d) = (1 - y) \cdot (\alpha d) + y \cdot (\max(0, m - d)), \tag{1}$$

where the predefined margin $m > 0$ specifies the minimum required separation between dissimilar pairs, while the hyperparameter $\alpha \geq 0$ regulates the influence of an additional linear attraction term applied to similar pairs. Examples of gender-related terms are listed in Appendix B.1, with contrastive loss details in Appendix B.2, and word-pair construction in Appendix B.3.

4 Training the Language Model

Background. Transformer-based language models (Vaswani et al., 2017) map a sequence of token embeddings through their layers, and finally average across all embeddings to predict a final class prediction (Figure 1, Step 3). While this maximizes task performance, the task loss alone does not enforce that gendered or toxic synonyms lie close together, and any geometry established during pre-training can drift during downstream training. However, maintaining this geometry is critical for downstream formal verification.

Fairness-aware training. We therefore integrate the pre-trained embedding layer from Section 3 into a transformer-based language model and keep it frozen, i.e., it is not updated via backpropagation during training. This preserves the geometry obtained during pre-training: gender-related (or toxic) terms remain clustered and synonyms remain close, which can be exploited during verification, as the proximity cannot be undone by the model training. This procedure follows common practice in models such as ELMo and ULMFiT (Peters et al., 2018; Howard & Ruder, 2018). Conversely, when the embedding layer is instead trained jointly with the model, gender-related words are not embedded in close proximity, and the model fails to adhere to the formalized robustness in the embedding space (Section 3).

5 Verifying the Fairness of Language Models

Background. A variety of tools have been developed to formally verify neural network properties, with adversarial robustness being the most extensively studied aspect (Kaulen et al., 2025). The verification problem for classification can be formally stated as follows:

Let $f\colon \mathbb{R}^n \to \mathbb{R}^c$ be a neural network with input $x \in \mathbb{R}^n$ with $y = f(x)$ and $t \in [c]$ the target class. Let $\psi_{\text{in}}(x)$ be an input specification capturing the local neighborhood of x, then a neural network verifier needs to determine if

$$\forall x \in \mathbb{R}^n\colon \psi_{\text{in}}(x) \implies \arg\max f(x) = t. \tag{2}$$

If there exists a point x for which this statement cannot be proven, the network is considered *unsafe*; otherwise, it is considered *safe*.

Exact verification of robustness properties in neural networks is often computationally infeasible (Katz et al., 2017). Therefore, sound but incomplete methods are usually used for larger networks (Katz et al., 2019; Zhang et al., 2018; Bak, 2021; Lopez et al., 2023; Singh et al., 2018; Duong et al., 2024). While our work is agnostic to the underlying verification engine (Kaulen et al., 2025), we use zonotopes (Girard, 2005) to enclose the output of a neural network in this work, which have been shown to be suitable for verifying large neural networks (Singh et al., 2018; Gehr et al., 2018a; Mirman et al., 2018) including transformer-based models (Bonaert et al., 2021). A detailed explanation of zonotope-based neural network verification can be found in Appendix A.

Capturing synonyms with perturbation balls. As enumerating all synonym combinations quickly results in a combinatorial explosion, we construct gender-related synonyms (or equivalently toxic terms) by modeling ℓ_∞-balls around each embedding of a sensitive word (Figure 1, Step 4). Here, the pre-training ensured that sensitive words are in close proximity in the embedding space (Section 3) and are thus captured by these balls. Thus, a fair model should maintain consistent predictions across the entire ball of each word.

In more detail, we perturb the embedding of each word in a sentence by some perturbation radius $\epsilon > 0$. We then query the neural network verifier to find the largest ϵ_{max} for which we can still verify that the prediction remains unchanged via binary search. Once ϵ_{max} is obtained, we assess if all synonyms of the words of interest lie within the computed perturbation ball. This can be easily checked by determining a threshold δ given by the largest ℓ_∞ distance between each word i and its respective synonyms in the embedding space. Let $\hat{z}_i$ be the embedding of word i, then

$$\delta = \max_i \max_{j \in \mathcal{N}(i)} \|\hat{z}_i - \hat{z}_j\|_\infty , \tag{3}$$

where $\mathcal{N}(i)$ contains all synonyms and related words of word i. If it holds that $\delta \leq \epsilon_{\text{max}}$, we know that any possible combination of synonyms does not change the prediction.

Fairness score. We can assess the overall fairness of a model by checking the percentage of all sentences in our test dataset for which such a certificate can be obtained. In particular, let there be N sentences in a held-out test dataset, then the fairness score Ψ can be computed by:

$$\Psi = \frac{1}{N} \sum_{l=1}^{N} \mathbf{1}_{(\delta_l \leq \epsilon_{\text{max},l})}, \tag{4}$$

where $\mathbf{1}_{(\cdot)}$ is an indicator if the certificate is obtained for sentence l. Thus, a higher Ψ (closer to 1) implies a fairer model. The entire verification procedure is shown in Alg. 1.

Algorithm 1 Formal Fairness Verification

Require: Dataset $\{S_1, \ldots, S_N\}$, model f, embedding $\hat{z}$, synonyms $\mathcal{N}(i)$ for each word i

```
s ← 0                                   // certified sentences count
for l = 1 → N do
    ε_max,l ← find max. verifiable perturbation
    δ_l ← max_{i∈S_l} max_{j∈N(i)} ‖ẑ(i) − ẑ(j)‖_∞
    if δ_l ≤ ε_max,l then
        s ← s + 1
    end if
end for
return Ψ ← s/N ∈ [0, 1]
```

6 Experimental Results

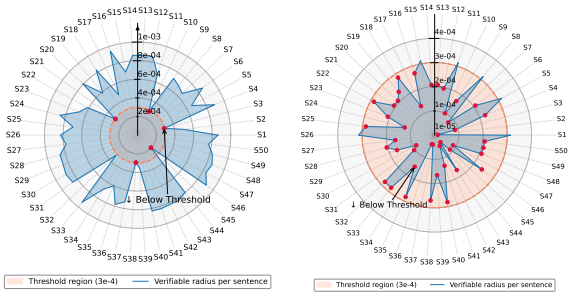
6.1 Datasets

Gender bias mitigation. To mitigate gender bias in the learned representations, we employ transfer learning to pre-train a Keras embedding layer using a curated set of gender-specific words from prior work (Bolukbasi et al., 2016) containing gendered word pairs (e.g., `man–woman`, `male–female`, `king–queen`). Detailed preprocessing steps on the pre-training dataset can be found in Appendix B.

The dataset used in the model training for gender bias mitigation in this study is the Adult Census dataset (Becker & Kohavi, 1996), including demographic information from the 1994 U.S. Census. The task is to predict whether an individual's income exceeds $50,000 per year, revealing potential biases in predictions based on race, gender, and other sensitive features. This dataset has been widely used in the context of fairness in machine learning (Lahoti et al., 2020; Friedler et al., 2019; Patel & Jadhav, 2024; Le Quy et al., 2022). Detailed preprocessing steps, including the serialization of tabular data by converting it to a sentence dataset, can be found in Appendix B.4.

Table 1: Model performance (test accuracy) and verification metrics. 'NA' (Not Applicable) indicates that gender terms were omitted; hence, fairness could not be computed.

	Salary Prediction	
Model	**Test Acc. (↑)**	**Fairness Ψ (↑)**
Gender attributes excluded		
8 blocks	82%	NA
10 blocks	83%	NA
Gender attributes included		
8 blocks	88%	90%
10 blocks	88%	12%
	Toxicity Detection	
Model	**Test Acc. (↑)**	**Fairness Ψ (↑)**
3 blocks	95%	50%



(a) 8-block model. (b) 10-block model.

Figure 2: Verifiability radar plot with gender-distance reference circle (coral dashed circle) in gender bias mitigation with 8 and 10 blocks of transformers. The axes S1 to S50 stand for the 50 sentences from the held-out test dataset.

Toxicity detection. The pre-training of embedding layers follows the same framework as that of the gender bias mitigation experiment, using a bad words collection from Google's publicly available bad words list.[1] For the training and evaluation of the language model, the Jigsaw Toxic Comment Classification dataset (Adams et al., 2017) was utilized. This dataset has been widely adopted in various studies (Patel et al., 2025; Chakrabarty, 2020; Revelo-Bautista et al., 2023; Marchiori Maneba et al., 2022; Gupta et al., 2022). In this study, we use a curated version of the Jigsaw toxic comments dataset (Davidson et al., 2017) designed to evaluate the severity of toxic language by incorporating annotator agreement probabilities across three categories: neutral, offensive, and hate speech. We formulated a binary classification task, where neutral comments are labeled as non-toxic (label = 0), and both offensive and hate speech comments are labeled as toxic (label = 1).

6.2 Main Results

We evaluated our approach on text classification language models with up to 10 transformer blocks. We provide details on the pre-training hyperparameters in Appendix C.1 and ablation studies including more models in Appendix C.2.

Gender fairness. We evaluate the fairness of our models using the fairness score Ψ (4), computed on the held-out test dataset containing 50 sentences with equal class balance. Table 1 presents the models' test accuracy and fairness scores after training.

To illustrate the results, Figure 2 shows radar plots of per-sentence verifiability, where each axis represents a sentence and the blue region's tip indicates the maximum verifiable perturbation radius $\epsilon_{\max}$. Gender-related tokens are identified as the 50 closest neighbors to the words `male` and `female` (the primary gender indicators in the Adult Census dataset) in the embedding space. The largest ℓ_∞ distance between any of their nearest 50 neighbors is $3e{-}04$ for both the tokens `male` and `female` (Table 2). We use this value as a reference threshold $\delta \leq \epsilon_{\max}$ required for a model to be verified fair on any given sentence, as tokens within this threshold should be treated as semantically equivalent by the model. Thus, sentences with tips outside the radius δ are deemed verifiable (fair), while red dots mark unverifiable cases. A full list of gender-related neighbors is provided in Appendix C.2. Notably, `male` and `female` appear within each other's 50 nearest neighbors, demonstrating the effectiveness of pre-training.

Table 2: Example ℓ_∞ distances between some gender-related word embeddings.

Word pair	ℓ_∞ Dist.
`female` ↔ `male`	$3e{-}04$
`female` ↔ `man`	$3e{-}04$
`male` ↔ `woman`	$2e{-}04$
`man` ↔ `woman`	$3e{-}04$
`he` ↔ `she`	$2e{-}04$

[1] https://code.google.com/archive/p/badwordslist/downloads

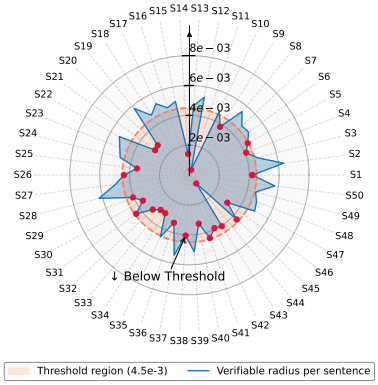


Figure 3: Verifiability radar plot with toxicity-distance reference circle (coral dashed circle) for the 3-block toxicity detection model in content moderation.

Table 3: Example ℓ_∞ distances between selected toxic terms and their nearest neighbors in embedding space. **The vocabulary is publicly available via a Google-hosted archive dataset and does not reflect the opinions of the authors**.

Word pair	ℓ_∞ Dist.
mother-fucker ↔ ##ore	3.4e−03
mother-fucker ↔ tits	4.0e−03
mother-fucker ↔ fuck	4.1e−03
bitch ↔ cock	2.6e−03
bitch ↔ ho	4.0e−03
gay ↔ ass	3.3e−03
gay ↔ ##ker	4.0e−03
gay ↔ bi	2.4e−03
nigger ↔ monkey	3.6e−03
nigger ↔ ##er	3.4e−03
nigger ↔ cock	2.6e−03

The 8-block language model achieved a fairness score $\Psi = 90\%$, indicating that its predictions remain unbiased with respect to gender across all tested scenarios in the salary prediction task. In contrast, the 10-block model only obtains $\Psi = 12\%$. As both models have the same test accuracy, our fairness certification framework favors the smaller model as it is verifiably less biased.

Toxicity detection. Following the same approach used for gendered terms, we identify a set of common toxicity-related terms from the dataset and retrieve the 50 nearest neighbors for each term via a pre-trained embedding space. For each toxic word, we compute the maximum ℓ_∞ distance among its 50 nearest neighbors, representing the required semantic boundary for meaning-preserving substitutions. The largest of these distances again defines the threshold δ used to assess whether the model's predictions remain stable under semantically equivalent toxic term replacements.

The fairness score for the toxicity model is illustrated in Figure 3 and the full neighbor set is provided in Appendix C.2. The model achieved a test accuracy of 95% and a fairness score of 50% after training and formal verification. This indicates that the model can reliably censor toxic content in 50% of the tested cases, regardless of any adversarial paraphrasing using toxic equivalents. In addition, Table 3 presents the pairwise ℓ_∞ distances between frequently occurring toxicity-related word pairs in the Jigsaw dataset and throughout verification. The majority of them fall within the maximum verifiable noise radius for every sentence. This is particularly important in the context of content moderation, as an Internet user might attempt to evade detection by slightly modifying toxic terms (e.g., changing `arse` to `azzhole`). However, a model that has obtained a fairness certificate would guarantee that it can verify such edits and maintain detection capability, demonstrating the effectiveness of our verification approach in real-world scenarios.

6.3 Ablation Studies

To assess the necessity of the pre-training (Figure 1, Steps 1 and 2), we train the 8-block salary prediction model with an otherwise identical setup, but with an embedding layer that is not pre-trained and left unfrozen, so that it learns word representations solely via backpropagation. Without pre-training, gender-related words are not embedded in close proximity: The ℓ_∞ distance between `female` and `male` is 0.7581, compared to 3e$-$04 with pre-training (Table 2). Even when relaxing the threshold δ to this single distance, no test sentence can be certified, i.e., $\Psi = 0\%$ (Figure 4), whereas the same

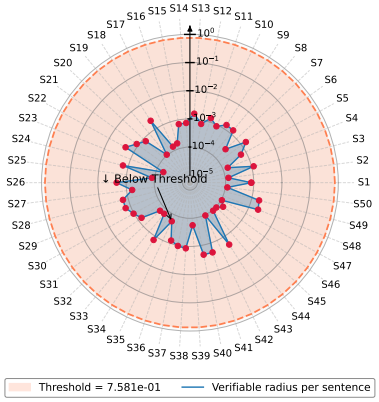


Figure 4: Verifiability radar plot of the 8-block model *without* embedding pre-training (dashed circle: threshold $\delta = 0.7581$).

architecture with pre-trained embeddings achieves $\Psi = 90\%$ (Table 1), showing that pre-training the embedding space is essential to obtain fairness certificates. We further observe that the fairness score reflects robustness gained through training: with synonym-based adversarial data augmentation, a 3-block model improves from $\Psi = 0\%$ to 44% at an unchanged test accuracy of 91% (Appendix C.2).

Further ablation studies. More experiments can be found in Appendix C.2.

6.4 Discussion

Generally, it is expected that larger models are harder to verify (Katz et al., 2017), which might lead to the conclusion that they naturally have a lower fairness score. However, we cannot generally confirm this hypothesis as more factors influence verifiability than just model size (Kaulen et al., 2025). However, we want to stress that improving verifiability is not the primary objective of this work, but rather that verification can be used to obtain fairness certificates in the first place. Thus, any progress in the field of formal neural network verification directly benefits our framework. Ideally, our framework exposes adversarial examples and biases and can guide the development of fairer language models, e.g., by integrating formal methods into the training process (Koller et al., 2025).

While removing gender attributes from the input may seem like a straightforward way to mitigate bias, this approach is often insufficient in practice. A prior study (Valentim et al., 2019) shows that removing sensitive attributes like gender does not always lead to a fairer model, as models can still learn bias from correlated features—and may even become more biased by relying on proxy attributes. Similar studies support this, showing that simply removing sensitive attributes is insufficient, as protected features may be encoded or correlated with other variables (Loi et al., 2021; González-Sendino et al., 2024; Goethals et al., 2024). Our experimental result (Table 1) further shows that excluding gender-related information in both the pre-training and language model training phases leads to a drop in model accuracy, indicating that such features can play a legitimate role in prediction, highlighting the necessity of our proposed fairness score to assess and manage bias rather than outright exclusion.

7 Related work

This work continues the progressive effort to formally verify neural networks in recent years (Kaulen et al., 2025): As neural networks are vulnerable to adversarial attacks (Goodfellow et al., 2015a), formally verifying them against input perturbations is of immense importance in safety-critical scenarios. Neural network verifiers can generally be categorized into optimization-based verifiers (Zhang et al., 2018; Katz et al., 2019; Henriksen & Lomuscio, 2020; Singh et al., 2019) and approaches using set-based computing (Gehr et al., 2018b; Singh et al., 2018; Lopez et al., 2023; Kochdumper et al., 2023). Early approaches focused on computing the exact output set of neural networks with ReLU activations, which has been shown to be NP-hard (Katz et al., 2017). Convex relaxations of the problem (Singh et al., 2018; Xu et al., 2021) and efficient branch-and-bound techniques (Wang et al., 2021) have been shown to achieve impressive verification results (Kaulen et al., 2025). Recently, non-convex relaxations have also been shown to be effective in this field of research (Kochdumper et al., 2023; Wang et al., 2023; Ladner & Althoff, 2023; Fatnassi et al., 2023; Ortiz et al., 2023).

Many works have focused on verifying the safety and fairness of neural networks. For example, FairNNV (Tumlin et al., 2024) verifies individual and counterfactual fairness via reachability analysis, while safety in robotic systems has been studied using formal methods (Sun et al., 2019). Probabilistic approaches, such as those using Markov chains (Sun et al., 2021), FairSquare (Albarghouthi et al., 2017), and VeriFair (Bastani et al., 2019), enable rigorous group fairness verification. The OVERT algorithm (Sidrane et al., 2022) enables sound safety verification of nonlinear, discrete-time, closed-loop dynamical systems with neural network control policies. Discriminatory input detection and fairness analysis are addressed by DeepGemini (Xie et al., 2023), complemented by Fairify's SMT-based method for verifying individual fairness (Biswas & Rajan, 2023). An automated verification framework for feed-forward multi-layer neural networks based on Satisfiability Modulo Theories (SMT) is proposed in (Huang et al., 2017), while NNV (Tran et al., 2020) offers a set-based verification approach tailored for deep neural networks. Other methods include training-based fairness improvement (Borca-Tasciuc et al., 2022), certified representation learning (Ruoss et al., 2020), structured data verification (John et al., 2020), global fairness verification with CertiFair (Khedr &

Shoukry, 2023), parallel static analysis for causal fairness (Urban et al., 2020), and Bayesian and satisfiability-based tools like FVGM and Justicia (Ghosh et al., 2021; 2022). A new approach to measuring and addressing discrimination in supervised learning, where a model is trained to predict an outcome based on input features, is proposed in (Hardt et al., 2016). However, prior work has predominantly focused on fairness verification of standard feed-forward neural networks, whereas our approach focuses on verifying both the fairness and safety properties of transformer-based language models.

Bias evaluation metrics for LLMs can be categorized by what they use from the model (Gallegos et al., 2024), such as the embeddings (Caliskan et al., 2017b; Dev et al., 2020; Guo & Caliskan, 2021), model-assigned probabilities (Webster et al., 2020; Kurita et al., 2019; Salazar et al., 2020; Wang & Cho, 2019; Nadeem et al., 2021; Kaneko & Bollegala, 2022) or generated text (Rajpurkar et al., 2016; Bordia & Bowman, 2019; Liang et al., 2023; Gehman et al., 2020; Sicilia & Alikhani, 2023; Nozza et al., 2021; Dhamala et al., 2021). Here, we focus on the generated text, as it is the primary focus of our study. Language models' responses to prompts can reveal social biases (Cheng et al., 2023), using tasks like prompt completion, conversation, and analogical reasoning (Li et al., 2024). The standard method involves prompting the model and evaluating its continuation for bias (Gallegos et al., 2024). GPT-4 has been shown to generate more racially stereotyped personas than human writers when prompted to describe demographic groups (Cheng et al., 2023). Similarly, earlier work validated gender bias in GPT-3's output using 800 prompts (e.g., "[He] was very" vs. "[She] was very"), revealing stereotypes in the model output and highlighting the need for debiasing (Brown et al., 2020; Abid et al., 2021). Other analyses show that language models frequently associate certain occupations with specific genders based on co-occurrence patterns (Liang et al., 2023). The fairness of an LLM is also sometimes evaluated using a second, more powerful, large language model (Eloundou et al., 2024), which does not provide formal guarantees. Rather, instead of finding the vulnerability of the first model, it can try to fool the second model to "pass" the fairness test (Wang et al., 2024).

While the vulnerability of large language models to adversarial prompts has been explored (Sun et al., 2024), research on the non-existence of such prompts remains limited, and the formal verification of transformers remains largely unexplored. As with standard neural networks, the method proposed by Shi et al. (2020) verifies robustness using linear bounds, allowing for at most two perturbed words. This work has been extended to the zonotope domain (Bonaert et al., 2021), enabling the verification of longer sentences with all words being perturbed. Other works also investigated finding better bounds of the softmax function within transformers (Wei et al., 2023; Shi et al., 2024) and investigated the preservation of dependencies between attention heads (Zhang et al., 2024).

8 LIMITATIONS

Given the scale and complexity of modern large language models (Achiam et al., 2024), existing verification techniques face a fundamental trade-off between precision and computational cost, rendering them impractical for large-scale models (Shi et al., 2020; Bonaert et al., 2021). As this method is sound but incomplete, an obtained fairness certificate always holds, but the method may fail to verify models that are, in fact, fair. Nonetheless, our verification framework is agnostic to the underlying verification method, and we hope that with the continuous progress in the field of neural network verification (Kaulen et al., 2025), the fairness of more complex architectures can be verified using our framework.

This work focuses on gender as a demographic attribute in salary prediction, but the proposed fairness evaluation framework is attribute-agnostic and can be extended to other variables, such as race. While our experiments address binary classification, the framework is, in principle, applicable to multi-class tasks and broader fairness assessments.

Most existing works apply arbitrary fixed ϵ-perturbations to the input and examine whether such changes can alter the predictions of language models. However, they often overlook a critical assumption: Whether semantically meaningful changes, such as synonym substitution or variation across gender or toxicity-related terms, are actually captured by the ϵ-ball. During training, the focus tends to be limited to improving fairness on a few sampled input sentences without exploring how the embedding space could be shaped to support other important criteria such as mitigating gender bias or detecting adversarially crafted yet semantically equivalent toxic terms. In this work, we address this gap by offering a holistic perspective on the formal verification of language models.

We also want to emphasize that training the embedding layer such that synonyms (or gender-related or toxicity-related terms) are close to each other requires expert knowledge, as such lists must be curated and adapted over time. Readers should note that this study does not aim to build an exhaustive list of synonyms, bad words, or a collection of all gender-related terms. Rather, it demonstrates that pre-training-based transfer learning enables a simple embedding model to effectively cluster synonyms, gender-related, or toxicity-related words in the embedding space.

9 Conclusion

We present a generalized verification framework for formally assessing the fairness and safety of language models and demonstrate it on models with up to 10 transformer blocks. In particular, the framework consists of carefully designing the embedding space to capture synonyms, gender-related terms, or toxicity-related terms through perturbation using pre-training-based transfer learning. This embedding layer is then integrated into the custom transformer models for downstream language model training, ensuring the fairness of their predictions. Lastly, the fairness of the models is formally verified over a held-out test set. Importantly, this formal verification technique circumvents the combinatorial complexity of testing individual word substitutions or synonyms, thereby enabling sentence-level verification.

Using this holistic framework, we made sure that gender-related terms are embedded into sufficient proximity through targeted pre-training, allowing different gender terms to be treated as semantically equivalent during inference. This embedding structure is critical in enabling formal verification of gender fairness, as model outputs remained invariant under gender word substitutions. Additionally, we applied the same approach to toxicity-related terms, showing that semantically similar toxic words clustered closely and were consistently detected, even under adversarial perturbations. This highlights the robustness of the system in content moderation tasks. As a result, the framework achieved a fairness score of 90% for an 8-block transformer language model in the salary prediction task, and 50% for a 3-block toxicity detection model. These findings underscore the potential utility of our framework in gender bias mitigation for fairness enforcement and its reliability in supporting content moderation systems.

AI Use Statement

In this work, generative AI tools were used solely to assist with text formatting and selected coding tasks. No generative AI tools were used for conducting literature reviews or generating research ideas.

Ethics Statement

This work aims to improve the fairness and safety of language models by certifying that predictions are invariant to gender-related substitutions and that adversarially paraphrased toxic inputs are consistently detected. All experiments use publicly available datasets: the de-identified Adult Census dataset (Becker & Kohavi, 1996) and the Jigsaw Toxic Comment Classification dataset (Adams et al., 2017). The toxicity experiments necessarily involve offensive language, which is used solely to train and evaluate detection models and is not endorsed by the authors. As our verification is sound but incomplete and relies on curated word lists, a certificate should not be read as a general guarantee of fairness, but as one component of a broader fairness audit.

Reproducibility Statement

All hyperparameters for embedding pre-training and language model training are listed in Tables 7 and 8, and the adversarial training configuration is in Table 11. Dataset construction and preprocessing, including the gender-specific word lists and the tabular-to-sentence conversion of the Adult Census dataset, are described in Appendices B.1, B.3 and B.4. The verification procedure is formalized in Algorithm 1 and Appendix A, and uses the publicly available MATLAB toolbox CORA (Althoff, 2015). Code and some model weights used for verification will be included in the repeatability package.

REFERENCES

Abubakar Abid, Maheen Farooqi, and James Zou. Persistent anti-muslim bias in large language models. pp. 298–306. Association for Computing Machinery, 2021. ISBN 9781450384735. URL https://doi.org/10.1145/3461702.3462624.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, and et al. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2024.

C. J. Adams, Jeffrey Sorensen, Julia Elliott, Lucas Dixon, Mark McDonald, Nithum Thain, and Will Cukierski. Toxic comment classification challenge. https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge, 2017. Kaggle Competition.

Aws Albarghouthi, Loris D'Antoni, Samuel Drews, and Aditya V Nori. Fairsquare: probabilistic verification of program fairness. *Proceedings of the ACM on Programming Languages*, 1:1–30, 2017.

Matthias Althoff. An introduction to CORA 2015. In *Proc. of the Workshop on Applied Verification for Continuous and Hybrid Systems*, pp. 120–151, 2015.

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. Generating natural language adversarial examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2890–2896, 2018.

Gilbert Badaro, Mohammed Saeed, and Paolo Papotti. Transformers for tabular data representation: A survey of models and applications. *Transactions of the Association for Computational Linguistics*, 11:227–249, 2023.

Tao Bai, Jinqi Luo, Jun Zhao, Bihan Wen, and Qian Wang. Recent advances in adversarial training for adversarial robustness. In Zhi-Hua Zhou (ed.), *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pp. 4312–4321. International Joint Conferences on Artificial Intelligence Organization, 8 2021. doi: 10.24963/ijcai.2021/591. URL https://doi.org/10.24963/ijcai.2021/591. Survey Track.

Stanley Bak. nnenum: Verification of relu neural networks with optimized abstraction refinement. In *NASA Formal Methods Symposium*, pp. 19–36, 2021.

Solon Barocas and Andrew D Selbst. Big data's disparate impact. *California Law Review*, 104:671, 2016.

Osbert Bastani, Xin Zhang, and Armando Solar-Lezama. Probabilistic verification of fairness properties via concentration. *Proceedings of the ACM on Programming Languages*, 3, 2019.

Barry Becker and Ronny Kohavi. Adult. UCI Machine Learning Repository, 1996.

Melika Behjati, Seyed-Mohsen Moosavi-Dezfooli, Mahdieh Soleymani Baghshah, and Pascal Frossard. Universal adversarial attacks on text classifiers. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 7345–7349, 2019.

Dmitriy Bespalov, Sourav Bhabesh, Yi Xiang, Liutong Zhou, and Yanjun Qi. Towards building a robust toxicity predictor. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pp. 581–598, 2023.

Lorenzo Betti, Carlo Abrate, and Andreas Kaltenbrunner. Large scale analysis of gender bias and sexism in song lyrics. *EPJ Data Science*, 12(1):10, 2023.

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

Fantahun Bogale Gereme and William Zhu. Fighting fake news using deep learning: Pre-trained word embeddings and the embedding layer investigated. In *Proceedings of the 2020 3rd International Conference on Computational Intelligence and Intelligent Systems*, pp. 24–29, 2021. URL https://doi.org/10.1145/3440840.3440847.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in Neural Information Processing Systems*, 29, 2016.

Gregory Bonaert, Dimitar I Dimitrov, Maximilian Baader, and Martin Vechev. Fast and precise certification of transformers. In *Proceedings of the 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation*, pp. 466–481, 2021.

Giorgian Borca-Tasciuc, Xingzhi Guo, Stanley Bak, and Steven Skiena. Provable fairness for neural network models using formal verification. *arXiv preprint arXiv:2212.08578*, 2022.

Shikha Bordia and Samuel R. Bowman. Identifying and reducing gender bias in word-level language models. In Sudipta Kar, Farah Nadeem, Laura Burdick, Greg Durrett, and Na-Rae Han (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pp. 7–15, 2019. URL https://aclanthology.org/N19-3002/.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020.

Marc-Etienne Brunet, Colleen Alkalay-Houlihan, Ashton Anderson, and Richard Zemel. Understanding the origins of bias in word embeddings. In *International Conference on Machine Learning*, pp. 803–811, 2019.

Toon Calders and Sicco Verwer. Three naive bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, 21:277–292, 2010.

Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356:183–186, 2017a.

Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186, 2017b.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924, 2020.

Navoneel Chakrabarty. A machine learning approach to comment toxicity classification. In *Computational Intelligence in Pattern Recognition: Proceedings of CIPR 2019*, pp. 183–193, 2020.

Tessa ES Charlesworth, Victor Yang, Thomas C Mann, Benedek Kurdi, and Mahzarin R Banaji. Gender stereotypes in natural language: Word embeddings show robust consistency across child and adult language corpora of more than 65 million words. *Psychological Science*, 32(2):218–240, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pp. 1597–1607, 2020a.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in Neural Information Processing Systems*, 33:22243–22255, 2020b.

Wenhu Chen, Jianshu Chen, Yu Su, Zhiyu Chen, and William Yang Wang. Logical natural language generation from open-domain tables. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7929–7942, 2020c.

Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyou Zhou, and William Yang Wang. Tabfact: A large-scale dataset for table-based fact verification. In *International Conference on Learning Representations*, 2020d. URL https://openreview.net/forum?id=rkeJRhNYDH.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020e.

Myra Cheng, Esin Durmus, and Dan Jurafsky. Marked personas: Using natural language prompts to measure stereotypes in language models. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pp. 1504–1532, 2023. URL https://aclanthology.org/2023.acl-long.84/.

S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pp. 539–546 vol. 1, 2005.

Xiaodong Cui, Vaibhava Goel, and Brian Kingsbury. Data augmentation for deep neural network acoustic modeling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23: 1469–1477, 2015.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media*, pp. 512–515, 2017.

Sunipa Dev, Tao Li, Jeff M Phillips, and Vivek Srikumar. On measuring and mitigating biased inferences of word embeddings. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 7659–7666, 2020.

Jwala Dhamala, Tony Sun, Varun Kumar, Satyapriya Krishna, Yada Pruksachatkun, Kai-Wei Chang, and Rahul Gupta. Bold: Dataset and metrics for measuring biases in open-ended language generation. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, pp. 862–872, New York, NY, USA, 2021. URL https://doi.org/10.1145/3442188.3445924.

Hai Duong, Dong Xu, ThanhVu Nguyen, and Matthew B. Dwyer. Harnessing neuron stability to improve dnn verification. *Proceedings of the ACM on Software Engineering*, 1:859–881, 2024.

Tyna Eloundou, Alex Beutel, David G Robinson, Keren Gu-Lemberg, Anna-Luisa Brakman, Pamela Mishkin, Meghan Shah, Johannes Heidecke, Lilian Weng, and Adam Tauman Kalai. First-person fairness in chatbots. *arXiv preprint arXiv:2410.19803*, 2024.

Wael Fatnassi, Haitham Khedr, Valen Yamamoto, and Yasser Shoukry. Bern-nn: Tight bound propagation for neural networks using bernstein polynomial interval arithmetic. In *Proceedings of the 26th ACM International Conference on Hybrid Systems: Computation and Control*, pp. 1–11, 2023.

Sorelle A Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P Hamilton, and Derek Roth. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 329–338, 2019.

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

Mario De Lucas García. Trustworthy NLP: Are LLMs adversarially aligned? *arXiv preprint arXiv:2308.05374*, 2023.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. RealToxicityPrompts: Evaluating neural toxic degeneration in language models. In Trevor Cohn, Yulan He, and Yang Liu (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 3356–3369, 2020. URL https://aclanthology.org/2020.findings-emnlp.301/.

Timon Gehr, Matthew Mirman, Dana Drachsler-Cohen, Petar Tsankov, Swarat Chaudhuri, and Martin Vechev. Ai2: Safety and robustness certification of neural networks with abstract interpretation. In *2018 IEEE Symposium on Security and Privacy*, pp. 3–18, 2018a.

Timon Gehr, Matthew Mirman, Dana Drachsler-Cohen, Petar Tsankov, Swarat Chaudhuri, and Martin Vechev. AI2: Safety and robustness certification of neural networks with abstract interpretation. In *IEEE Symposium on Security and Privacy*, pp. 3–18, 2018b.

Bishwamittra Ghosh, Debabrota Basu, and Kuldeep S Meel. Justicia: A stochastic sat approach to formally verify fairness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 7554–7563, 2021.

Bishwamittra Ghosh, Debabrota Basu, and Kuldeep S Meel. Algorithmic fairness verification with graphical models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 9539–9548, 2022.

Antoine Girard. Reachability of uncertain linear systems using zonotopes. In *International Workshop on Hybrid Systems: Computation and Control*, pp. 291–305, 2005.

Sofie Goethals, David Martens, and Toon Calders. Precof: counterfactual explanations for fairness. *Machine Learning*, 113:3111–3142, 2024.

Rubén González-Sendino, Emilio Serrano, Javier Bajo, and Paulo Novais. A review of bias and fairness in artificial intelligence. 2024.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015a.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015b. URL `https://arxiv.org/abs/1412.6572`.

Wei Guo and Aylin Caliskan. Detecting emergent intersectional biases: Contextualized word embeddings contain a distribution of human-like biases. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 122–133, 2021.

Sakshi Gupta, Mugdha Goel, and Nisha Rathee. Machine learning approach to classify toxic comments on social media platforms. In *Proceedings of International Conference on Data Science and Applications, Volume 2*, pp. 439–449, 2022.

Jeff Z HaoChen, Colin Wei, Adrien Gaidon, and Tengyu Ma. Provable guarantees for self-supervised deep learning with spectral contrastive loss. *Advances in Neural Information Processing Systems*, 34:5000–5011, 2021.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in Neural Information Processing Systems*, 29, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

Patrick Henriksen and Alessio Lomuscio. Efficient neural network verification via adaptive refinement and adversarial search. In *European Conference on Artificial Intelligence*, volume 325, pp. 2513–2520, 2020.

Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pp. 328–339, 2018.

Yu-Lun Hsieh, Minhao Cheng, Da-Cheng Juan, Wei Wei, Wen-Lian Hsu, and Cho-Jui Hsieh. On the robustness of self-attentive models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1520–1529, 2019.

Po-Sen Huang, Robert Stanforth, Johannes Welbl, Chris Dyer, Dani Yogatama, Sven Gowal, Krishnamurthy Dvijotham, and Pushmeet Kohli. Achieving verified robustness to symbol substitutions via interval bound propagation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 4083–4093, 2019.

Xiaowei Huang, Marta Kwiatkowska, Sen Wang, and Min Wu. Safety verification of deep neural networks. In *International Conference on Computer Aided Verification*, pp. 3–29, 2017.

Xiaowei Huang, Wenjie Ruan, Wei Huang, Gaojie Jin, Yi Dong, Changshun Wu, Saddek Bensalem, Ronghui Mu, Yi Qi, Xingyu Zhao, et al. A survey of safety and trustworthiness of large language models through the lens of verification and validation. *Artificial Intelligence Review*, 57(7):175, 2024.

Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A survey on contrastive self-supervised learning. *Technologies*, 9(1):2, 2020.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8018–8025, 2020.

Philips George John, Deepak Vijaykeerthy, and Diptikalyan Saha. Verifying individual fairness in machine learning models. In *Conference on Uncertainty in Artificial Intelligence*, pp. 749–758, 2020.

Faisal Kamiran, Toon Calders, and Mykola Pechenizkiy. Discrimination aware decision tree learning. In *2010 IEEE International Conference on Data Mining*, pp. 869–874, 2010.

Masahiro Kaneko and Danushka Bollegala. Unmasking the mask–evaluating social biases in masked language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 11954–11962, 2022.

Guy Katz, Clark Barrett, David L. Dill, Kyle Julian, and Mykel J. Kochenderfer. Reluplex: An efficient SMT solver for verifying deep neural networks. In *International Conference on Computer Aided Verification*, pp. 97–117, 2017.

Guy Katz, Derek A. Huang, Duligur Ibeling, Kyle Julian, Christopher Lazarus, Rachel Lim, Parth Shah, Shantanu Thakoor, Haoze Wu, Aleksandar Zeljić, David L. Dill, Mykel J. Kochenderfer, and Clark Barret. The Marabou framework for verification and analysis of deep neural networks. In *International Conference on Computer Aided Verification*, pp. 443–452, 2019.

Konstantin Kaulen, Tobias Ladner, Stanley Bak, Christopher Brix, Hai Duong, Thomas Flinkow, Taylor T. Johnson, Lukas Koller, Edoardo Manino, ThanhVu H. Nguyen, et al. The 6th international verification of neural networks competition (VNN-COMP 2025): Summary and results. *arXiv preprint arXiv:2512.19007*, 2025.

Nikhil Ketkar and Nikhil Ketkar. Introduction to keras. *Deep learning with python: a hands-on introduction*, pp. 97–111, 2017.

Haitham Khedr and Yasser Shoukry. Certifair: A framework for certified global fairness of neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 8237–8245, 2023.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673, 2020.

Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. Audio augmentation for speech recognition. In *Interspeech*, volume 2015, pp. 3586, 2015.

Sosuke Kobayashi. Contextual augmentation: Data augmentation by words with paradigmatic relations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2*, pp. 452–457, 2018.

Niklas Kochdumper, Christian Schilling, Matthias Althoff, and Stanley Bak. Open- and closed-loop neural network verification using polynomial zonotopes. In *NASA Formal Methods Symposium*, pp. 16–36, 2023.

Lukas Koller, Tobias Ladner, and Matthias Althoff. Set-based training for neural network verification. *Transaction on Machine Learning Research*, 2025.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25, 2012.

Udo Kruschwitz and Maximilian Schmidhuber. Llm-based synthetic datasets: Applications and limitations in toxicity detection. In *Proceedings of the Fourth Workshop on Threat, Aggression*, pp. 37–51, 2024.

Aounon Kumar, Chirag Agarwal, Suraj Srinivas, Aaron Jiaxun Li, Soheil Feizi, and Himabindu Lakkaraju. Certifying LLM safety against adversarial prompting. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=9Ik05cycLq.

Keita Kurita, Nidhi Vyas, Ayush Pareek, Alan W Black, and Yulia Tsvetkov. Measuring bias in contextualized word representations. In Marta R. Costa-jussà, Christian Hardmeier, Will Radford, and Kellie Webster (eds.), *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pp. 166–172, 2019. URL https://aclanthology.org/W19-3823/.

Tobias Ladner and Matthias Althoff. Automatic abstraction refinement in neural network verification using sensitivity analysis. In *Proceedings of the 26th ACM International Conference on Hybrid Systems: Computation and Control*, pp. 1–13, 2023.

Preethi Lahoti, Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, and Ed Chi. Fairness without demographics through adversarially reweighted learning. *Advances in Neural Information Processing Systems*, 33:728–740, 2020.

Tai Le Quy, Arjun Roy, Vasileios Iosifidis, Wenbin Zhang, and Eirini Ntoutsi. A survey on datasets for fairness-aware machine learning. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 12(3):e1452, 2022.

Bruno Lepri, Nuria Oliver, Emmanuel Letouzé, Alex Pentland, and Patrick Vinck. Fair, transparent, and accountable algorithmic decision-making processes: The premise, the proposed solutions, and the open challenges. *Philosophy & Technology*, 31(4):611–627, 2018.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. Textbugger: Generating adversarial text against real-world applications. In *26th Annual Network and Distributed System Security Symposium*, 2019.

Yingji Li, Mengnan Du, Rui Song, Xin Wang, and Ying Wang. A survey on fairness in large language models, 2024. URL https://arxiv.org/abs/2308.10149.

Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. Deep text classification can be fooled. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pp. 4208–4215, 2018.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D Manning, Christopher Re, Diana Acosta-Navas, Drew A. Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue WANG, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri S. Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Andrew Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. Holistic evaluation of language models. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=iO4LZibEqW.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

Yang Liu and Yi-Fang Wu. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Michele Loi, Andrea Ferrario, and Eleonora Viganò. Transparency as design publicity: explaining and justifying inscrutable algorithms. *Ethics and Information Technology*, 23(3):253–263, 2021.

Diego Manzanas Lopez, Sung Woo Choi, Hoang-Dung Tran, and Taylor T. Johnson. NNV 2.0: The neural network verification tool. In *International Conference on Computer Aided Verification*, pp. 397–412, 2023.

Marta Marchiori Maneba, Riccardo Guidotti, Lucia Passaro, Salvatore Ruggieri, et al. Bias discovery within human raters: A case study of the jigsaw dataset. In *Proceedings of the LREC 2022 workshop on Perspectivist Approaches to Disagreement in NLP*, pp. 26–31, 2022.

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

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013a.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 26, 2013b.

George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11): 39–41, 1995.

Matthew Mirman, Timon Gehr, and Martin Vechev. Differentiable abstract interpretation for provably robust neural networks. In *International Conference on Machine Learning*, pp. 3578–3586, 2018.

Moin Nadeem, Anna Bethke, and Siva Reddy. StereoSet: Measuring stereotypical bias in pretrained language models. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *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)*, pp. 5356–5371, 2021. doi: 10.18653/v1/2021.acl-long.416. URL https://aclanthology.org/2021.acl-long.416/.

J Neeraja, Vivek Gupta, and Vivek Srikumar. Incorporating external knowledge to enhance tabular reasoning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2799–2809, 2021.

Debora Nozza, Federico Bianchi, and Dirk Hovy. HONEST: Measuring hurtful sentence completion in language models. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2398–2406, 2021. doi: 10.18653/v1/2021.naacl-main.191. URL https://aclanthology.org/2021.naacl-main.191/.

Joshua Ortiz, Alyssa Vellucci, Justin Koeln, and Justin Ruths. Hybrid zonotopes exactly represent ReLU neural networks. In *62nd IEEE Conference on Decision and Control (CDC)*, pp. 5351–5357, 2023.

Dharil Patel, Pijush Kanti Dutta Pramanik, Chaitanya Suryawanshi, and Preksha Pareek. Detecting toxic comments on social media: an extensive evaluation of machine learning techniques. *Journal of Computational Social Science*, 8(1):1–18, 2025.

Namrata Jiten Patel and Ashish Jadhav. Design of an efficient dynamic context-based privacy policy deployment model via dual bioinspired q learning optimisations. *IET Cyber-Physical Systems: Theory & Applications*, 9(4):477–496, 2024.

Dana Pessach and Erez Shmueli. A review on fairness in machine learning. *ACM Computing Surveys*, 55(3):1–44, 2022.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2227–2237, 2018.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In Jian Su, Kevin Duh, and Xavier Carreras (eds.), *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392, 2016. URL https://aclanthology.org/D16-1264/.

María Fernanda Revelo-Bautista, Jair Oswaldo Bedoya-Benavides, Jaime Paúl Sayago-Heredia, Pablo Pico-Valencia, and Xavier Quiñonez-Ku. Classification of toxic comments on social networks using machine learning. In *International Conference on Applied Technologies*, pp. 257–270. Springer, 2023.

Anian Ruoss, Mislav Balunovic, Marc Fischer, and Martin Vechev. Learning certified individually fair representations. *Advances in Neural Information Processing Systems*, 33:7584–7596, 2020.

Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. Masked language model scoring. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2699–2712, 2020. URL https://aclanthology.org/2020.acl-main.240/.

Nikunj Saunshi, Orestis Plevrakis, Sanjeev Arora, Mikhail Khodak, and Hrishikesh Khandeparkar. A theoretical analysis of contrastive unsupervised representation learning. In *International Conference on Machine Learning*, pp. 5628–5637, 2019.

Zhouxing Shi, Huan Zhang, Kai Wei Chang, Minlie Huang, and Cho Jui Hsieh. Robustness verification for transformers. In *International Conference on Learning Representations*, 2020.

Zhouxing Shi, Qirui Jin, Zico Kolter, Suman Jana, Cho-Jui Hsieh, and Huan Zhang. Neural network verification with branch-and-bound for general nonlinearities. *arXiv preprint arXiv:2405.21063*, 2024.

Anthony Sicilia and Malihe Alikhani. Learning to generate equitable text in dialogue from biased training data. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2898–2917, 2023. doi: 10.18653/v1/2023.acl-long.163. URL https://aclanthology.org/2023.acl-long.163/.

Chelsea Sidrane, Amir Maleki, Ahmed Irfan, and Mykel J Kochenderfer. Overt: An algorithm for safety verification of neural network control policies for nonlinear systems. *Journal of Machine Learning Research*, 23(117):1–45, 2022.

Patrice Simard, Yann LeCun, John S. Denker, and Bernard Victorri. Transformation invariance in pattern recognition-tangent distance and tangent propagation. In *Neural Networks: Tricks of the Trade, This Book is an Outgrowth of a 1996 NIPS Workshop*, pp. 239–27, 1998.

Gagandeep Singh, Timon Gehr, Matthew Mirman, Markus Püschel, and Martin Vechev. Fast and effective robustness certification. *Advances in Neural Information Processing Systems*, 31, 2018.

Gagandeep Singh, Timon Gehr, Markus Püschel, and Martin Vechev. An abstract domain for certifying neural networks. In *Proceedings of the ACM on Programming Languages*, volume 3, pp. 1–30, 2019.

Lya Hulliyyatus Suadaa, Hidetaka Kamigaito, Kotaro Funakoshi, Manabu Okumura, and Hiroya Takamura. Towards table-to-text generation with numerical reasoning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 1451–1465, 2021.

Yoshihiko Suhara, Jinfeng Li, Yuliang Li, Dan Zhang, Çağatay Demiralp, Chen Chen, and Wang-Chiew Tan. Annotating columns with pre-trained language models. In *Proceedings of the 2022 International Conference on Management of Data*, pp. 1493–1503, 2022.

Bing Sun, Jun Sun, Ting Dai, and Lijun Zhang. Probabilistic verification of neural networks against group fairness. In *Formal Methods: 24th International Symposium*, pp. 83–102, 2021.

Lichao Sun, Yue Huang, Haoran Wang, Siyuan Wu, Qihui Zhang, et al. Position: TrustLLM: Trustworthiness in large language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 20166–20270, 2024.

Xiaowu Sun, Haitham Khedr, and Yasser Shoukry. Formal verification of neural network controlled autonomous systems. In *Proceedings of the 22nd ACM International Conference on Hybrid Systems: Computation and Control*, pp. 147–156, 2019.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations*, 2014.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9, 2015.

James Thorne, Majid Yazdani, Marzieh Saeidi, Fabrizio Silvestri, Sebastian Riedel, and Alon Halevy. Database reasoning over text. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 3091–3104, 2021.

Hoang-Dung Tran, Xiaodong Yang, Diego Manzanas Lopez, Patrick Musau, Luan Viet Nguyen, Weiming Xiang, Stanley Bak, and Taylor T Johnson. Nnv: the neural network verification tool for deep neural networks and learning-enabled cyber-physical systems. In *International Conference on Computer Aided Verification*, pp. 3–17, 2020.

Anne M Tumlin, Diego Manzanas Lopez, Preston Robinette, Yuying Zhao, Tyler Derr, and Taylor T Johnson. Fairnnv: The neural network verification tool for certifying fairness. In *Proceedings of the 5th ACM International Conference on AI in Finance*, pp. 36–44, 2024.

Caterina Urban, Maria Christakis, Valentin Wüstholz, and Fuyuan Zhang. Perfectly parallel fairness certification of neural networks. *Proceedings of the ACM on Programming Languages*, 4:1–30, 2020.

Inês Valentim, Nuno Lourenço, and Nuno Antunes. The impact of data preparation on the fairness of software systems. In *2019 IEEE 30th International Symposium on Software Reliability Engineering (ISSRE)*, pp. 391–401, 2019. doi: 10.1109/ISSRE.2019.00046.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

Alex Wang and Kyunghyun Cho. BERT has a mouth, and it must speak: BERT as a Markov random field language model. In Antoine Bosselut, Asli Celikyilmaz, Marjan Ghazvininejad, Srinivasan Iyer, Urvashi Khandelwal, Hannah Rashkin, and Thomas Wolf (eds.), *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pp. 30–36, 2019. URL https://aclanthology.org/W19-2304/.

Feng Wang and Huaping Liu. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2495–2504, 2021.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. Large language models are not fair evaluators. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pp. 9440–9450, 2024. URL https://aclanthology.org/2024.acl-long.511/.

Shiqi Wang, Huan Zhang, Kaidi Xu, Xue Lin, Suman Jana, Cho-Jui Hsieh, and J. Zico Kolter. Beta-crown: Efficient bound propagation with per-neuron split constraints for neural network robustness verification. *Advances in Neural Information Processing Systems*, 34:29909–29921, 2021.

William Yang Wang and Diyi Yang. That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using# petpeeve tweets. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 2557–2563, 2015.

Yixuan Wang, Weichao Zhou, Jiameng Fan, Zhilu Wang, Jiajun Li, Xin Chen, Chao Huang, Wenchao Li, and Qi Zhu. Polar-express: Efficient and precise formal reachability analysis of neural-network controlled systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2023.

Kellie Webster, Xuezhi Wang, Ian Tenney, Alex Beutel, Emily Pitler, Ellie Pavlick, Jilin Chen, Ed Chi, and Slav Petrov. Measuring and reducing gendered correlations in pre-trained models. *arXiv preprint arXiv:2010.06032*, 2020.

Dennis Wei, Haoze Wu, Min Wu, Pin-Yu Chen, Clark Barrett, and Eitan Farchi. Convex bounds on the softmax function with applications to robustness verification. In *International Conference on Artificial Intelligence and Statistics*, pp. 6853–6878, 2023.

Jason Wei and Kai Zou. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 6382–6388, 2019.

Xuan Xie, Fuyuan Zhang, Xinwen Hu, and Lei Ma. Deepgemini: verifying dependency fairness for deep neural network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 15251–15259, 2023.

Kaidi Xu, Huan Zhang, Shiqi Wang, Yihan Wang, Suman Jana, Xue Lin, and Cho Jui Hsieh. Fast and complete: Enabaling complete neural network verification with raphid and massively parallel imcomplete verifieres. In *9th International Conference on Learning Representations*, 2021.

Huan Zhang, Tsui-Wei Weng, Pin-Yu Chen, Cho-Jui Hsieh, and Luca Daniel. Efficient neural network robustness certification with general activation functions. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. *Advances in Neural Information Processing Systems*, 28, 2015.

Yunruo Zhang, Lujia Shen, Shanqing Guo, and Shouling Ji. Galileo: General linear relaxation framework for tightening robustness certification of transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 21797–21805, 2024.

Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. Learning gender-neutral word embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 4847–4853, 2018.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2021.

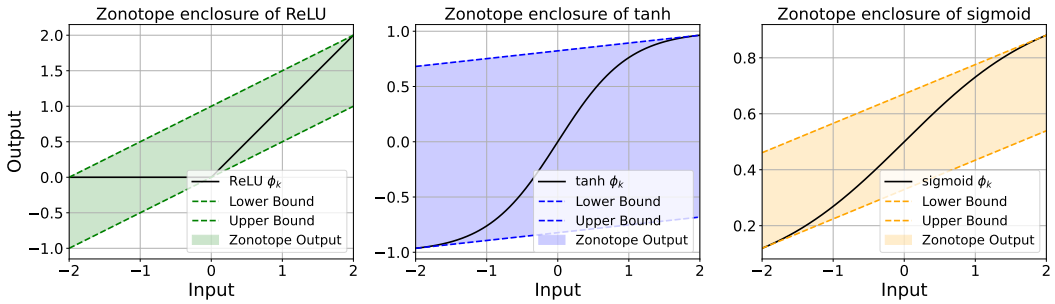


Figure 5: Zonotopic enclosure of some of the activation functions ϕ_k.

Appendix

A Zonotope-based verification of neural networks

Definition. Given a center vector $c \in \mathbb{R}^n$ and a generator matrix $G \in \mathbb{R}^{n \times q}$, a zonotope (Girard, 2005) is defined as

$$\mathcal{Z} = \langle c, G \rangle_Z = \left\{ c + \sum_{j=1}^{q} \beta_j G_{(:,j)} \;\middle|\; \beta_j \in [-1, 1] \right\}. \tag{5}$$

Zonotopes are convex, centrally symmetric polytopes that are suitable for the formal verification of neural networks due to their compact representation and efficient computation of the required operations. Generally, the verification process follows these steps:

1. **Initialization:** Represent the input as a zonotope based on bounds or perturbation limits.
2. **Propagation through network:** Propagate the input set through all layers while ensuring that the output of each layer is properly enclosed.
3. **Safety specification checking:** Check if the resulting zonotope satisfies a given safety property.

We explain these steps for a standard feed-forward neural network. More complex architectures can be verified analogously.

Initialization. The input set $\mathcal{X}$ should capture all points x for which $\psi_{\text{in}}(x)$ holds. Typically, $\mathcal{X}$ captures the ℓ_∞-bounded perturbations around some given input $x_0 \in \mathbb{R}^n$. Given a perturbation radius $\epsilon \in \mathbb{R}_+$, the input zonotope is given by

$$\mathcal{X} = \langle x_0, \epsilon I_n \rangle_Z \subset \mathbb{R}^n. \tag{6}$$

Propagation through the network. A layer k of a neural network f is typically either linear, applying an affine map using a weight matrix W_k and bias b_k, or a nonlinear activation ϕ_k, e.g., ReLU or sigmoid. Let $\mathcal{H}_{k-1} = \langle c, G \rangle_Z$ be the input set to a layer k (with $\mathcal{H}_0 = \mathcal{X}$). The output of the affine transformation can be computed exactly using zonotopes and is given by

$$\mathcal{H}_k = W_k \mathcal{H}_{k-1} + b_k = \langle W_k c + b_k, W_k G \rangle_Z . \tag{7}$$

Nonlinear activations cannot be computed exactly using zonotopes and are therefore over-approximated to maintain soundness. Examples of this procedure for different activation functions ϕ_k can be found in Figure 5.

After the output of the last layer κ is enclosed, the output of the neural network is given by $\mathcal{Y} = \mathcal{H}_\kappa$.

Safety specification checking. After the output zonotope $\mathcal{Y}$ is obtained, we check whether $\mathcal{Y}$ does not intersect with some unsafe set $\mathcal{S}$, where $\mathcal{S}$ captures the unsafe specifications ψ_{out}. For classification tasks, it holds that

$$\mathcal{Y} \cap \mathcal{S} = \emptyset \implies \forall x \in \mathcal{X} : \arg\max f(x) = \arg\max f(x_0). \tag{8}$$

Thus, the specifications are formally verified if the emptiness of the intersection $\mathcal{Y} \cap \mathcal{S}$ is shown.

B Dataset construction and preprocessing

B.1 Gender-specific dataset

The gender-specific word list used for pre-training is sourced from (Bolukbasi et al., 2016), where the authors derived gendered terms from WordNet based on definitional criteria. We adopt this word set and apply the Word Embedding Association Test (WEAT) to categorize terms into `male`- and `female`-associated groups using a predefined attribute set (Table 4); this approach follows established practice, as previously used by literature studying gender bias in embeddings or corpora (Betti et al., 2023; Caliskan et al., 2017a; Charlesworth et al., 2021).

Table 4: List of attribute word sets used in the Word Embedding Association Test (WEAT), originally derived from (Caliskan et al., 2017a; Betti et al., 2023) and subsequently expanded by the authors to enrich semantic coverage and improve gender-related classification.

Attribute Set	**Words**
`female`	`woman`, `women`, `she`, `mother`, `mothers`, `girl`, `girls`, `aunt`, `aunts`, `sister`, `sisters`, `queen`, `queens`, `lady`, `ladies`, `wife`, `wives`, `madam`, `madams`, `actress`, `actresses`, `niece`, `nieces`, `princess`, `princesses`, `braid`, `braids`, `bride`, `brides`, `mom`, `moms`, `grandmother`, `grandmothers`, `daughter`, `daughters`, `goddess`, `goddesses`, `nun`, `nuns`, `ladyship`, `ladyships`, `baroness`, `baronesses`, `empress`, `empresses`, `witch`, `witches`, `duchess`, `duchesses`
`male`	`man`, `men`, `he`, `father`, `fathers`, `boy`, `boys`, `uncle`, `uncles`, `brother`, `brothers`, `king`, `kings`, `gentleman`, `gentlemen`, `husband`, `husbands`, `sir`, `sirs`, `actor`, `actors`, `nephew`, `nephews`, `prince`, `princes`, `beard`, `beards`, `groom`, `grooms`, `dad`, `dads`, `grandfather`, `grandfathers`, `son`, `sons`, `godfather`, `godfathers`, `monk`, `monks`, `lord`, `lords`, `baron`, `barons`, `emperor`, `emperors`, `knight`, `knights`, `wizard`, `wizards`, `duke`, `dukes`

B.2 Details on contrastive loss function

The loss function is designed to jointly optimize the embedding space by drawing semantically similar word pairs closer together while pushing dissimilar pairs apart. For similar pairs, a linear pull term proportional to the predicted distance is introduced, promoting tighter semantic clustering. To ensure numerical stability and prevent excessively large gradients, we deliberately avoid logarithmic or high-degree penalty functions, opting instead for a linear formulation. For dissimilar pairs, we employ a standard contrastive loss component, which penalizes instances where the predicted distance falls below a predefined margin, thereby enforcing a minimum separation between unrelated terms.

While pre-training aims to minimize the ℓ_2 distance between embeddings, the verification process instead focuses on finding the maximum verifiable radius under the ℓ_∞ norm, since formal verification over the ℓ_2 domain is not directly possible using zonotopes. The authors intentionally adopt the ℓ_2 distance for pre-training because it minimizes the overall distance between points, rather than just the maximum per-dimension difference. This encourages the model to learn more generalized and robust representations. Furthermore, the authors observed in their experiments that after pre-training, the ℓ_2 distance between semantically similar words (such as gendered word pairs) is often slightly larger than their corresponding ℓ_∞ distance (as can be seen in Appendix C.2). This supports the idea that utilizing the ℓ_2 norm benefits downstream verification.

B.3 EMBEDDING MODEL PRE-TRAINING DATA CONSTRUCTION

As large-scale contextualized models like BERT are hard to formally verify (Shi et al., 2020; Bonaert et al., 2021), we pre-train a Keras embedding layer (Ketkar & Ketkar, 2017). To construct an effective training set for contrastive loss optimization, we curated word pairs based on the semantic gender relationship using a heuristic approach. Each `male`-labeled word was explicitly paired with the terms `woman` and `female` to anchor the gender axis, and further matched with a diverse sample of 50–100 `female`-labeled words to form similar (label=0) pairs. This ensures that embeddings of gender-related terms are pulled closer in the vector space. To balance the contrastive objective, we generated dissimilar (label=1) pairs by selecting words from a broad English vocabulary that have no direct lexical or semantic relationship (e.g., synonyms, derivations) to any gender-specific term. This distinction was enforced by filtering out related terms using WordNet relations such as synonyms, hypernyms, and derivationally related forms.

To enhance the general semantic capacity of the embedding space, particularly because the embedding layer is later frozen during task-specific backpropagation, we conducted an auxiliary contrastive training phase focused on common semantic relationships such as synonymy. (This general semantic training step proved essential, as model performance degraded significantly in its absence. This outcome is expected, given that the embedding layer, when integrated within the transformer-based architecture, is frozen during downstream training and therefore does not receive gradient updates to refine its semantic representations.) Using WordNet, we generated word pairs labeled as similar (label=0) by extracting synonym pairs, while carefully excluding any gender-specific terms to preserve the integrity of gender representations learned in earlier pre-training stages, thereby maintaining the intended semantic distances between gendered terms. To form the dissimilar (label=1) pairs, we randomly sampled words from the general vocabulary that were not semantically related (each non-gendered word is paired with all valid WordNet synonyms, excluding any gendered terms, resulting in a variable number of synonym pairs per word. To ensure a balanced dataset, an equal number of dissimilar word pairs is generated, maintaining a one-to-one ratio with the total number of synonym pairs), again ensuring that no gender-specific tokens were involved. This allowed the model to learn to cluster semantically similar words and push apart unrelated ones. By training the embedding space on these general lexical relationships, we improve its ability to capture broader semantic structure, which is especially important since the embeddings are utilized as fixed features in downstream transformer-based language model text classification tasks. This step ensures that the embeddings are not only gender-aware but also semantically meaningful in a general linguistic context.

For pre-training the toxicity-related embedding model, a similar pre-training strategy was employed—using a contrastive learning framework to pull semantically similar words closer and push dissimilar words apart (to handle rare or profane words not explicitly found in the tokenizer vocabulary, we use BERT's WordPiece-based subword tokenization. This strategy breaks out-of-vocabulary terms into smaller known units (e.g., `mother-fucker` becomes [`mother`, `-`, `fuck`, `##er`]), allowing consistent representation throughout the embedding layer pre-training and language model training phases without full word-level coverage. These subwords are embedded and mean-pooled, enabling the model to capture semantic information even for obfuscated or stylized expressions). The strength of the pulling and pushing mechanisms (i.e., the values of α and the margin), along with hyperparameters, can be found in Appendix C.1. The distance between gendered words, toxicity words, synonyms, and unrelated words before and after pre-training of the embedding can be found in Appendix C.2.

B.4 ADULT CENSUS DATASET PREPROCESSING

Transformers require one-dimensional token sequences as input, whereas tabular data is inherently two-dimensional (Badaro et al., 2023); however, the demographic information in the Adult Census dataset is structured in a tabular format, with each attribute represented as a separate column. To bridge this gap, the data and its contextual structure must be serialized before being processed by the model (Badaro et al., 2023). A critical step in adapting tabular data for transformer-based models is converting the tabular structure into a linearized, one-dimensional format. Common strategies include: column-wise concatenation of values, insertion of special separator tokens between fields (Badaro et al., 2023; Suhara et al., 2022), and use of text templates to represent each row as a structured sentence (Chen et al., 2020d; Suadaa et al., 2021; Chen et al., 2020c). More recent approaches move beyond predefined templates by generating natural language sentences directly from tabular

Table 5: Sample data instance from the Adult Census dataset, along with the constructed input sentence.

Structured tabular data instance				
Workclass	**Education**	**Edu-Num**	**Occupation**	**Sex**
Private	7th-8th grade	4	Machine-op-inspct	female
Marital Status	**Relationship**	**Race**	**Country**	**Hours/Week**
Divorced	Unmarried	White	United-States	40
Constructed input sentence				
A person's workclass is Private, education is 7th-8th grade (number of years of education is 4), occupation is Machine-op-inspct, sex is female, marital status is Divorced, relationship is Unmarried, race is White, native country is United States, and they work 40 hours per week.				

data. This is achieved by fine-tuning sequence-to-sequence models such as T5 (Raffel et al., 2020) or GPT (Radford et al., 2019), as demonstrated in prior works (Thorne et al., 2021; Neeraja et al., 2021). We adopt simple concatenation for its straightforward implementation and effectiveness in our setting.

To adapt the Adult Census dataset for use with a sentence-to-label transformer-based binary classification model, we converted the structured tabular data into natural language sentences. The original dataset contains both numerical and categorical features, along with information indicating whether an individual's income exceeds $50K. Since transformer models (e.g., BERT) operate on text sequences, we constructed a heuristic template that combines relevant features into coherent sentences. Specifically, attributes such as workclass, education, occupation, sex, marital status, relationship, race, native country, and hours per week were embedded into a natural language sentence format. The salary attribute, serving as the ground truth label for prediction, was intentionally excluded from the sentence construction. This transformation preserves the semantic content of the original features while enabling the use of sentence-based binary text classification models on structured tabular data. An example tabular data instance before and after sentence construction can be found in Table 5 (the feature "fnlwgt", along with several other complex numerical attributes, was excluded from this experiment, as the authors deliberately chose to focus on leveraging textual, simple numerical, and categorical features for model training and representation).

B.5 Data augmentation

Text classification is a fundamental task in natural language processing, and achieving high performance often depends heavily on the size and quality of the training data (Wei & Zou, 2019; Kobayashi, 2018). Automatic data augmentation is widely employed in machine learning and computer vision, among other domains, to enhance model robustness, particularly when working with limited datasets (Simard et al., 1998; Szegedy et al., 2015; Krizhevsky et al., 2012; Cui et al., 2015; Ko et al., 2015).

A commonly used data augmentation strategy is synonym replacement, where selected words are substituted with their synonyms. This can be achieved using handcrafted lexical resources like WordNet (Miller, 1995; Zhang et al., 2015), word similarity computations (Wang & Yang, 2015), or contextual embedding-based methods (Kobayashi, 2018).

We employed a simple WordNet-based synonym replacement strategy to augment the dataset in the salary prediction experiments, aiming to increase data volume and enhance classification accuracy. This approach led to a notable improvement in test accuracy, rising from 80% to around 90% (we utilized straightforward WordNet synonym replacement, as empirical observations indicated that this method is already effective enough in improving model performance).

It is worth noting that data augmentation was performed while maintaining balanced class distributions across the dataset. An example of a sentence before and after applying synonym replacement prior to training on the transformer model for the salary prediction task can be found in Table 6, which was part of our investigation into gender fairness. The replacement probability was set to be 30%.

Table 6: Original vs. synonym-augmented sentence after data augmentation.

Constructed input sentence
A person's workclass *is* Self-emp-inc, *sex is* male, *education is* Bachelors *(number of years of education is 13)*, *occupation is* Exec-managerial, *marital status is* Married-civ-spouse, *relationship is* Husband, *race is* White, *native country is* United-States, *and they work* 52 *hours per week.*
Augmented input sentence (synonym replacement)
A person's workclass *is* Self-emp-inc, *sex be* male, *educational activity is* Bachelors *(number of years of education be 13)*, *occupation is* Exec-managerial, *marital status is* Married-civ-spouse, *human relationship is* Husband, *race is* White, *aborigine country is* United-States, *and they work* 52 *hours per week.*

C Evaluation details and ablation studies

C.1 Evaluation details

It is important to note that all parameters—including the margin and α values used during embedding pre-training, as well as the hyperparameters for model training—must be carefully tuned to achieve both high accuracy and substantial verifiable radii. Comprehensive details on these hyperparameters are provided in the appendix. Specifically, the hyperparameters used for pre-training the Keras embedding layer are listed in Table 7, while those utilized for training the language models are presented in Table 8. We use a smaller transformer for toxicity detection as explicit cues make the task simpler, and larger models led to empirically observed overfitting.

The selection of margin and α values is based on empirical observation. The goal is not only to ensure that gendered words are embedded in close proximity but also to promote the clustering of synonyms while pushing antonyms farther apart, without compromising the relative closeness of gendered word pairs. The margin values are also selected empirically based on the nuanced relationship between the embedding space and the resulting verifiable radius. A detailed study of this relationship is provided in Appendix C.2. The loss function used in this study is focal loss (Lin et al., 2017), a training-time calibration technique that directly integrates class imbalance handling into the learning process. It works by adapting the loss function to emphasize challenging, misclassified samples.

To evaluate the fairness score of the models, we use the MATLAB toolbox CORA (Althoff, 2015), which enables formal verification using set-based computing (Kochdumper et al., 2023; Ladner & Althoff, 2023; Bonaert et al., 2021). Please visit Appendix A for a detailed explanation of this process using zonotopes.

C.2 Ablation studies

Sentence-wise maximum verifiable perturbation radius

The maximum verifiable perturbed radius is reported in this section (see Tables 9 and 10) across 50 random sentences selected from a held-out test set, each comprising 20 tokens. The sample includes an equal distribution of 25 sentences from each class label, drawn from the test dataset. Example per-sentence radius values for a representative model from each experiment are provided in this appendix[2].

[2]Note that the presence of words outside the verifiable radius does not necessarily indicate a change in the model's classification output. Given that the verification method used in this study is incomplete, such instances may still result in correct classifications.

Table 7: Parameters used during embedding model pre-training (Gen. Sem. Train. refers to general semantic pre-training, such as synonym clustering).

Model	Salary Prediction & Gender Bias Mitigation					
	α	Margin	α (Gen. Sem. Train.)	Margin (Gen. Sem. Train.)	Epoch	Epoch (Gen. Sem. Train.)
6 blocks	1000	1.0	1.0	0.5	150	30
8 blocks	1000	1.0	1.0	0.8	150	30
10 blocks	1000	1.0	1.0	0.8	150	30
Model	**Toxicity Detection & Content Moderation**					
	α	Margin	α (Gen. Sem. Train.)	Margin (Gen. Sem. Train.)	Epoch	Epoch (Gen. Sem. Train.)
3 blocks	20	0.5	1.0	0.3	100	100

Table 8: Language model training hyperparameters used for salary prediction and toxicity detection tasks. The table includes vocabulary size, maximum sequence length, embedding dimension, number of attention heads, feedforward network dimension, dropout rate, number of training epochs, batch size, and optimization strategy.

Model	Salary Prediction & Gender Bias Mitigation							
	Vocab (BERT)	Seq. Len.	Embed Dim	Heads	FF Dim	Dropout	Epochs	Batch
6 blocks	30,522	200	8	2	8	0.2	100	32
8 blocks	30,522	200	8	2	8	0.1	100	32
10 blocks	30,522	200	8	2	8	0.01	100	32
Model	**Toxicity Detection & Content Moderation**							
	Vocab	Seq. Len.	Embed Dim	Heads	FF Dim	Dropout	Epochs	Batch
3 blocks	30,522	200	8	2	8	0.2	150	128
Optimization & training strategy								
Optimizer: AdamW (learning rate = 1e-3, weight decay = 1e-5)								
LR scheduler: ReduceLROnPlateau (monitor = val_loss, factor = 0.5, patience = 3, min_lr = 1e-6)								
Early stopping: Patience = 10, restore_best_weights = True								
Loss function: Focal Loss (α = 0.25, γ = 2.0)								

Table 9: Embedding bounds and maximum verifiable noise radii for the salary prediction model (8 blocks).

Model	**Salary Prediction & Gender Bias Mitigation (8 blocks)**			
Embedding Bounds		Minimum (inf)	-1.696724	
		Maximum (sup)	1.735214	
		Range (sup − inf)	3.431938	
	Sentences 1–25		**Sentences 26–50**	
	Sentence	**Max verifiable noise radius** (↑)	**Sentence**	**Max verifiable noise radius** (↑)
Noise Radii	Sentence 1	8.6079e-04	Sentence 26	8.2806e-04
	Sentence 2	4.9749e-04	Sentence 27	7.7405e-04
	Sentence 3	2.9457e-04	Sentence 28	8.5915e-04
	Sentence 4	8.8533e-04	Sentence 29	8.5915e-04
	Sentence 5	6.5950e-04	Sentence 30	8.7715e-04
	Sentence 6	8.5915e-04	Sentence 31	8.5915e-04
	Sentence 7	6.5459e-04	Sentence 32	3.1748e-04
	Sentence 8	6.8896e-04	Sentence 33	9.3279e-04
	Sentence 9	7.2169e-04	Sentence 34	6.8896e-04
	Sentence 10	2.8147e-04	Sentence 35	5.9895e-04
	Sentence 11	6.8896e-04	Sentence 36	7.8060e-04
	Sentence 12	8.5915e-04	Sentence 37	7.2823e-04
	Sentence 13	8.5915e-04	Sentence 38	2.9457e-04
	Sentence 14	8.5751e-04	Sentence 39	3.5512e-04
	Sentence 15	6.8896e-04	Sentence 40	8.2806e-04
	Sentence 16	9.4752e-04	Sentence 41	8.2806e-04
	Sentence 17	4.7949e-04	Sentence 42	8.1987e-04
	Sentence 18	8.2806e-04	Sentence 43	8.2806e-04
	Sentence 19	5.8586e-04	Sentence 44	7.9696e-04
	Sentence 20	8.0515e-04	Sentence 45	1.9638e-04
	Sentence 21	2.9457e-04	Sentence 46	9.1152e-04
	Sentence 22	6.1368e-04	Sentence 47	8.3624e-04
	Sentence 23	6.7586e-04	Sentence 48	8.5915e-04
	Sentence 24	8.5915e-04	Sentence 49	8.2806e-04
	Sentence 25	6.9878e-04	Sentence 50	7.6751e-04

Table 10: Embedding bounds and maximum verifiable noise radii for the toxicity detection model (3 blocks).

Model	**Toxicity Detection & Content Moderation (3 blocks)**			
Embedding Bounds	Minimum (inf)	-0.838544		
	Maximum (sup)	0.865149		
	Range (sup − inf)	1.703692		
	Sentences 1–25		**Sentences 26–50**	
	Sentence	**Max verifiable noise radius** (↑)	**Sentence**	**Max verifiable noise radius** (↑)
Noise Radii	Sentence 1	4.2090e-03	Sentence 26	4.3820e-03
	Sentence 2	6.3423e-03	Sentence 27	4.9913e-03
	Sentence 3	4.6858e-03	Sentence 28	6.2285e-03
	Sentence 4	4.0839e-03	Sentence 29	4.0562e-03
	Sentence 5	4.4446e-03	Sentence 30	3.5509e-03
	Sentence 6	4.8865e-03	Sentence 31	4.3999e-03
	Sentence 7	4.8061e-03	Sentence 32	3.3641e-03
	Sentence 8	5.5104e-03	Sentence 33	3.0164e-03
	Sentence 9	3.8215e-03	Sentence 34	3.0164e-03
	Sentence 10	4.6940e-03	Sentence 35	4.5656e-03
	Sentence 11	3.7451e-04	Sentence 36	3.3454e-03
	Sentence 12	5.2959e-03	Sentence 37	5.4438e-03
	Sentence 13	4.6745e-03	Sentence 38	4.0497e-03
	Sentence 14	1.4160e-03	Sentence 39	5.1343e-03
	Sentence 15	5.0148e-03	Sentence 40	3.3015e-03
	Sentence 16	4.7354e-03	Sentence 41	4.4242e-03
	Sentence 17	5.2748e-03	Sentence 42	3.9230e-03
	Sentence 18	4.7403e-03	Sentence 43	4.0457e-03
	Sentence 19	5.7736e-03	Sentence 44	7.1896e-04
	Sentence 20	2.9083e-03	Sentence 45	4.3341e-03
	Sentence 21	2.8401e-03	Sentence 46	3.1326e-03
	Sentence 22	5.3471e-03	Sentence 47	4.9742e-03
	Sentence 23	5.0067e-03	Sentence 48	4.8288e-03
	Sentence 24	4.7801e-03	Sentence 49	4.5640e-03
	Sentence 25	3.5371e-03	Sentence 50	5.7557e-03

Table 11: Training configuration, optimization strategy, and summary results for baseline and adversarially trained models. Fairness improvements are achieved through offline synonym-based augmentation during model training.

Model	Pre-training Hyperparameters and Augmentation Parameters							
	α	Margin	α (Gen. Sem.)	Margin (Gen. Sem.)	Epochs	Epochs (Gen. Sem.)	Aug. Prob.	Aug. Factor
3 blocks (baseline)	20	0.3	1.0	0.3	100	100	0	0
3 blocks (adv. trained)	20	0.3	1.0	0.3	100	100	0.8	3

Toxicity Detection & Content Moderation Model Configuration								
Model	Vocab	Seq. Len.	Embed Dim	Heads	FF Dim	Dropout	Epochs	Batch
3 blocks	30,522	200	8	2	8	0.2	150	128

Optimization & Training Strategy
Optimizer: AdamW (lr $= 10^{-3}$, weight decay $= 10^{-5}$)
LR scheduler: ReduceLROnPlateau (factor $= 0.5$, patience $= 3$, min lr $= 10^{-6}$)
Early stopping: Patience $= 10$, restore best weights
Loss: Focal Loss ($\alpha = 0.25$, $\gamma = 2.0$)

Evaluation Summary			Embedding Bounds	
Model	**Fairness Score**	**Test Acc.**		
Baseline	0%	91%	Minimum (inf)	-1.242537
Adversarially Trained	44%	91%	Maximum (sup)	1.161774
			Range (sup − inf)	2.404311

Adversarial Attack

Robustness is a key prerequisite for reliable model verification, and adversarial training makes the models more robust (Goodfellow et al., 2015b; Bai et al., 2021); therefore, in this section, we examine whether our framework assigns higher fairness scores to models that undergo adversarial training. To better show the effect of adversarial training, we consider a model variant with embedding layers that differ from those used in Section 6.2. This design choice ensures that the conventionally trained model exhibits a fairness score of zero, providing a clear baseline model from which improvements introduced by adversarial training can be more clearly measured. Please note that the 2 models used for this ablation study have the exact same embedding layers so that we can isolate the effect of adversarial training.

To improve robustness against perturbations, we apply *offline adversarial data augmentation* via synonym replacement. Given an existing labeled dataset, additional training samples are generated *only* for the positive class ($y = 1$), while all original samples are retained. The number of variants per positive instance is controlled by an *augmentation factor* k. In our adversarial experiments, we set the augmentation factor to be 3, producing three additional augmented sentences for each original positive example. Detailed pre-training and training hyperparameters can be found in Table 11.

Each augmented sentence is created through token-level synonym substitution based on WordNet. For every non-numeric token in a sentence, synonym replacement is applied independently with probability p. We use a high replacement probability of $p = 0.8$, which encourages the generation of more adversarial examples. This augmentation strategy introduces perturbations, thereby simulating adversarial paraphrases that increase the model's robustness during training. The result can be found in Table 11.

Fairness scores without pre-training the Keras embedding layer

This section presents the results of the fairness assessment without explicitly pre-training the embedding model. It is important to note that the Keras embedding layer was unfrozen during training, allowing it to learn basic semantic representations of words solely through backpropagation. This fairness assessment is limited to gender bias and is intended solely for illustrative purposes—to demonstrate that, without pre-training, the model fails to exhibit verifiably fair behavior.

Table 12: Top 50 most ℓ_∞-similar words to `female` and `male` using the Keras embedding layer **without** pre-training.

Model	Salary Prediction & Gender Bias Mitigation (8 blocks)							
	`female`				`male`			
	Word	**Distance**	**Word**	**Distance**	**Word**	**Distance**	**Word**	**Distance**
	service	6.710e−02	handler	8.93e−02	husband	9.55e−02	bachelor	1.133e−01
	fishing	1.273e−01	farming	1.318e−01	support	1.190e−01	widowed	1.262e−01
	unmarried	1.335e−01	other	1.421e−01	inc	1.367e−01	protective	1.395e−01
	10th	1.427e−01	2	1.450e−01	75	1.420e−01	france	1.434e−01
	7th	1.488e−01	6th	1.504e−01	##ec	1.465e−01	specialty	1.474e−01
	5th	1.560e−01	8	1.672e−01	85	1.477e−01	##em	1.519e−01
	1st	1.753e−01	5	1.794e−01	13	1.534e−01	##v	1.568e−01
	armed	1.831e−01	absent	1.841e−01	45	1.605e−01	16	1.713e−01
	8th	1.851e−01	6	1.954e−01	self	1.750e−01	s	1.801e−01
	relative	1.973e−01	7	1.978e−01	ci	1.785e−01	55	1.850e−01
	own	1.960e−01	honduras	2.079e−01	60	1.859e−01	52	1.862e−01
	cleaner	2.103e−01	12th	2.138e−01	italy	1.894e−01	90	1.911e−01
	4th	2.151e−01	scotland	2.166e−01	42	1.926e−01	ninth	1.941e−01
	salvador	2.187e−01	portugal	2.214e−01	ex	1.965e−01	spouse	1.971e−01
	3	2.232e−01	59	2.253e−01	twelfth	1.979e−01	48	2.007e−01
	32	2.295e−01	forces	2.344e−01	46	2.012e−01	tech	2.051e−01
	dominican	2.369e−01	##vi	2.384e−01	england	2.080e−01	lee	2.130e−01
	86	2.419e−01	thailand	2.427e−01	39	2.139e−01	prof	2.148e−01
	37	2.432e−01	11th	2.497e−01	15	2.171e−01	degree	2.182e−01
	married	2.184e−01	35	2.522e−01	pac	2.238e−01	89	2.268e−01
	74	2.537e−01	never	2.587e−01	edgar	2.269e−01	44	2.271e−01
	24	2.653e−01	jamaica	2.671e−01	iran	2.271e−01	el	2.277e−01
	peru	2.674e−01	4	2.675e−01	nation	2.286e−01	ireland	2.295e−01
	34	2.694e−01	puerto	2.750e−01	65	2.297e−01	43	2.305e−01
	92	2.756e−01	china	2.756e−01	federal	2.306e−01	63	2.782e−01

Table 13: ℓ_∞ distances between gender-related word pairs without specifically pre-training the Keras embedding layer.

Pair			**ℓ_∞ Distance**
`female`	$\leftrightarrow$	`male`	7.581e−01
`female`	$\leftrightarrow$	`man`	9.792e−01
`woman`	$\leftrightarrow$	`male`	2.582e−01
`woman`	$\leftrightarrow$	`man`	6.116e−01
`he`	$\leftrightarrow$	`she`	8.770e−02

The results for the 50 nearest neighbors and their corresponding distances for the words `female` and `male` are presented in Table 12. As shown, in the absence of pre-training, the nearest neighbors seldom reflect strongly gendered semantics. This underscores the importance of embedding layer pre-training in capturing meaningful semantic relationships, thereby justifying the design of our experimental framework. For comparison, the distance results with pre-training are provided in Appendix C.2; some examples of distances of strongly gender-related word pairs without specific pre-training can be found in Table 13.

For the sake of completeness, we conducted the fairness score assessment despite the absence of gendered words among the 50 nearest neighbors for both gender indicators; we simplified the fairness evaluation by setting the threshold based solely on the ℓ_∞ distance between `female` and `male` (which is the gender indicator in the Adult Census dataset), as the reference circle (which is 0.7581), rather than using the maximum distance among the 50 neighboring words of the two gender indicators. Please note that the entire experimental setup, including the language model training parameters, remains identical to the experiment that includes the pre-training of the embedding model (Section 6.2). The results of the formal verification for the model without any pre-training of the Keras embedding layer are presented in Figure 4. As shown, not only do all sentences fail verification, but their corresponding ℓ_∞ radii also lie significantly within the reference circle.

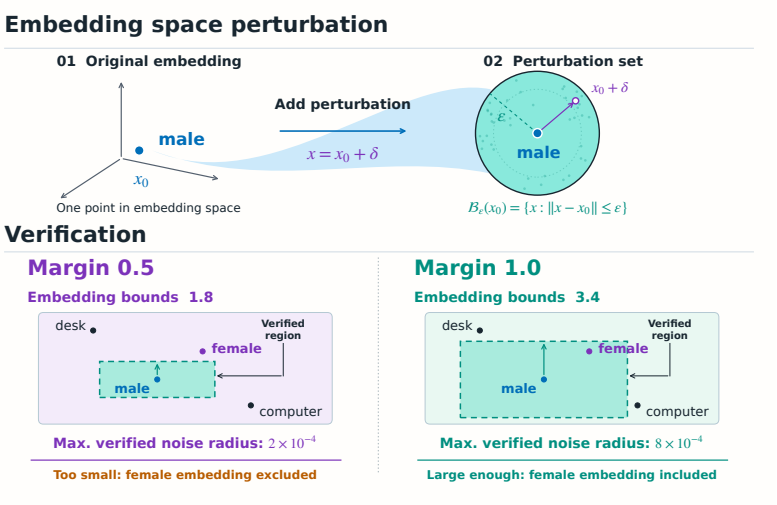


Figure 6: The effect of contrastive loss margin on embedding geometry and zonotope-based verification. A larger margin (right) results in more spread-out embeddings and enables larger certified perturbation bounds, while a smaller margin (left) leads to tighter clustering and limited verification robustness.

Relationship between embedding space and max perturbed noise radius

This section investigates the relationship between the margin parameter in (1), the resulting embedding bounds, and the maximum verifiable perturbation radius. As the margin parameter decreases, the embedding bounds contract accordingly, leading to a reduction in the verifiable radius. This contraction can impede the successful verification of semantically similar words, as their embedding distances may fall outside the increasingly tightened bounds. An illustrative graph Figure 6 highlights the trade-off between margin size and verifiability within semantic clusters (the range in the graph is for demonstrative purposes only; for more detailed metrics corresponding to specific margin values, please refer to Appendix C.2).

As shown in the graph, when the margin is too small, meaning the repelling (pushing) force between unrelated terms is weak, it can result in a tighter embedding bound. The distances between related words (e.g., gendered word pairs) determined during the pre-training phase remain relatively small and stable. This behavior is consistent with the mathematical formulation in (1), where the attractive force between related terms and the repulsive force between unrelated terms are combined through simple addition. Because they are added together rather than being interdependent through more complex interactions, these components remain mathematically separable. Consequently, an overly compact embedding space may cause related terms to fall outside the verifiable radius, as their distances remain generally unchanged after pre-training.

However, it has been empirically observed that the margin may have a slight influence on the final proximity of related word embeddings after pre-training, which could be attributed to the general geometric properties of the embedding space, but this effect is generally minimal and, in many cases, negligible. The authors further observed through empirical analysis that simply increasing the contrastive loss margin does not necessarily enhance the verification process. In fact, when the margin becomes excessively large, particularly in conjunction with an expanded embedding space, the verifiable noise radius does not always increase correspondingly. In some cases, it may even diminish. These findings highlight that the choice of margin is not a matter of maximizing separation or pulling force (α), but rather one of carefully balancing semantic dispersion with geometric certifiability.

Transformer-neighbor configurations

In this section, we examine how fairness scores are influenced by varying combinations of transformer block counts and the number of nearest neighbors. The experimental configuration for this analysis is illustrated in Figure 7. As previously discussed in Section 6.2, we presented fairness scores

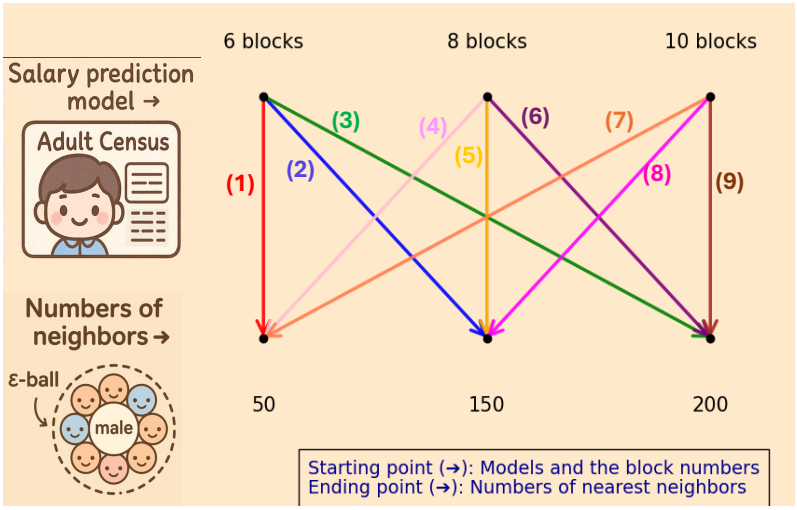


Figure 7: Ablation study on the influence of transformer block depth and neighborhood size on fairness. Experiments (1) through (9) illustrate the different combinations of transformer block counts and neighborhood sizes evaluated in this section to assess their impact on fairness.

for selected models with different transformer depths. Here, our objective is to provide a more comprehensive view, highlighting how fairness metrics evolve progressively as we adjust the number of transformer blocks and nearest neighbors. This analysis aims to uncover systematic trends and sensitivities within the model architecture that impact fairness.

For demonstration purposes, this analysis is limited to the salary prediction model within the context of gender bias mitigation verification, as our objective is to illustrate the influence of neighborhood size and the number of transformer blocks on changes in fairness scores, rather than to perform an exhaustive evaluation across all models. All of the hyperparameters used can be found in Appendix C.1.

Fairness score for 6 blocks of transformer for salary prediction model with varying numbers of nearest neighbors. We illustrate the progressive changes in fairness scores for the 6-block salary prediction model, evaluated across different settings of nearest neighbors (50, 150, and 200). These results highlight the impact of neighborhood size on fairness evaluation and are visualized in Figures 8 to 10. (The selected numbers of neighbors are intended to illustrate the progressive changes that occur as neighborhood size increases. This specific choice (50, 150, 200) is justified by the observation that increasing the number of neighbors from 50 to 100 does not significantly increase the maximum distance between neighbors, nor does it result in noticeable changes in verifiability.)

Fairness score for 8 blocks of transformer for salary prediction model with varying numbers of nearest neighbors. We illustrate the progressive changes in fairness scores for the 8-block salary prediction model, evaluated across different settings of nearest neighbors (50, 150, and 200). These results highlight the impact of neighborhood size on fairness evaluation and are visualized in Figures 11 to 13.

Fairness score for 10 blocks of transformer for salary prediction model with varying numbers of nearest neighbors. We illustrate the progressive changes in fairness scores for the 10-block salary prediction model, evaluated across different settings of nearest neighbors (50, 150, and 200). These results highlight the impact of neighborhood size on fairness evaluation and are visualized in Figures 14 to 16.

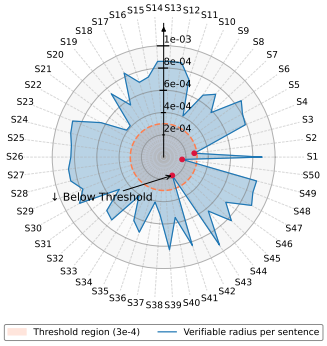


Figure 8: Fairness score for 6-block salary prediction model, 50 nearest neighbors (experiment (1) in Figure 7).

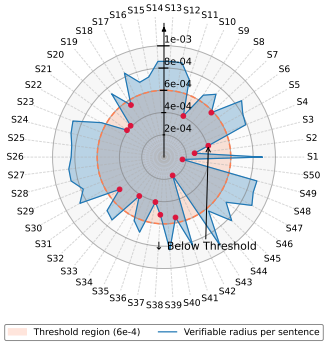


Figure 9: Fairness score for 6-block model, 150 nearest neighbors (experiment (2) in Figure 7).

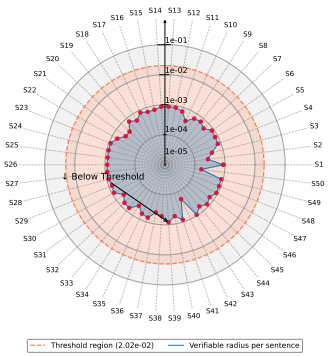


Figure 10: Fairness score for 6-block model, 200 nearest neighbors (experiment (3) in Figure 7).

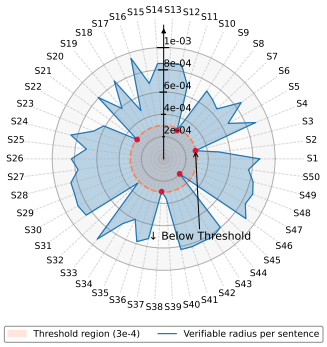


Figure 11: Verifiability radar plot with gender-distance reference circle for 8-block with 50 nearest neighbors (experiment (4) in Figure 7, also in Section 6.2).

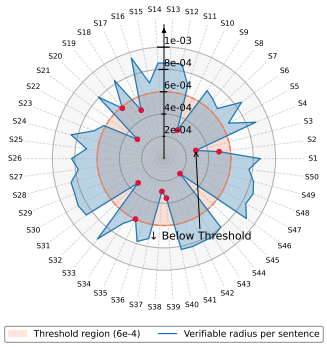


Figure 12: Verifiability radar plot with gender-distance reference circle for 8-block with 150 nearest neighbors (experiment (5) in Figure 7).

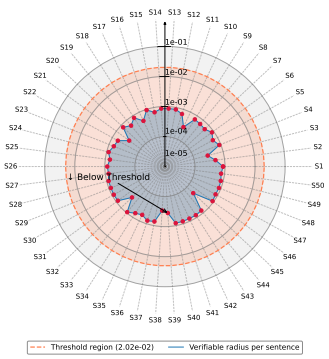


Figure 13: Verifiability radar plot with gender-distance reference circle for 8-block with 200 nearest neighbors (experiment (6) in Figure 7).

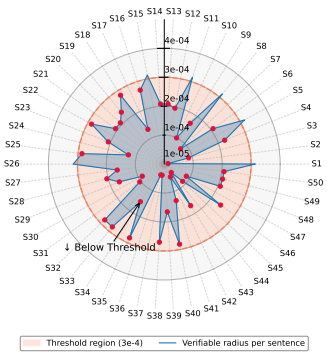


Figure 14: Verifiability radar plot with gender-distance reference circle for 10-block with 50 nearest neighbors (experiment (7) in Figure 7, also in Section 6.2).

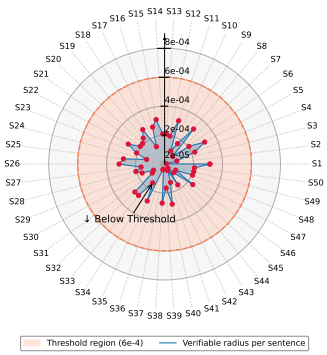


Figure 15: Verifiability radar plot with gender-distance reference circle for 10-block with 150 nearest neighbors (experiment (8) in Figure 7).

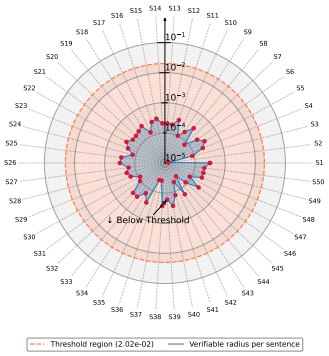


Figure 16: Verifiability radar plot with gender-distance reference circle for 10-block with 200 nearest neighbors (experiment (9) in Figure 7).

DISTANCES OF WORDS BEFORE AND AFTER PRE-TRAINING

This section presents the semantic relationships between words before and after the embedding model pre-training. The word pairs included are randomly selected to provide a representative illustration of the model's behavior.

The purpose of this section is to demonstrate the effectiveness of the pre-training process and, more importantly, ensure that the zonotope-based verification approach does not indiscriminately verify arbitrary words, but rather focuses on gendered terms within a sufficiently compact embedding space. This is achieved by ensuring that the verifiable noise radius is sufficiently large to include as many gendered words as possible, while small enough to exclude unrelated terms. This approach underscores the importance of careful hyperparameter tuning to balance coverage and precision in the verification process.

It should also be noted that while the proximity of gendered or toxicity-related words in the embedding space is largely influenced by the pulling factor (α in (1)), there appears to be a diminishing return in the Keras embedding layer's ability to further cluster words from the same semantic class beyond a certain distance (2e-04), while further separation of semantic classes yields minimal improvements.

Gendered words relationship before and after contrastive loss pre-training. The ℓ_2 and ℓ_∞ distances between gendered word pairs, as well as between gendered and unrelated word pairs, before and after embedding pre-training are presented in Table 15. As shown, gendered word pairs become significantly closer in the embedding space after pre-training, while unrelated pairs are pushed farther apart, demonstrating the intended effect of the contrastive pre-training approach.

It is important to note that not all related word pairs exhibited reduced distances after pre-training, nor did all unrelated pairs show increased separation. This is not a limitation of the pre-training process itself, but rather a consequence of computational constraints, which prevented exhaustive pairing of all possible related and unrelated words; doing so would have significantly expanded the data constructed for pre-training and potentially degraded the efficiency and performance of the embedding model. Readers should note that the objective of this experiment (as well as for toxicity detection) is not to train the embedding model on an exhaustive set of related and unrelated word pairs, but rather to illustrate the effectiveness of the pre-training-based transfer learning approach in shaping semantically meaningful embeddings.

The 50 nearest neighbors of the gendered indicator words after pre-training in the Adult Census dataset are presented in Table 14. A manual inspection was conducted to assess the relevance of these neighboring words to gender. The analysis confirmed that the majority of the neighbors are indeed gender-related. While a few terms—such as bull, stud, and colt—may not be explicitly gendered in all contexts, they can carry gendered connotations depending on usage. Overall, the presence of truly ungendered terms among the nearest neighbors is minimal and is determined by the authors to be negligible.

Toxic words relationship before and after contrastive loss pre-training. The ℓ_2 and ℓ_∞ distances between toxic word pairs, as well as between toxic and unrelated word pairs, before and after embedding pre-training are presented in Appendix C.2. As shown, toxicity-related word pairs become significantly closer in the embedding space after pre-training, while unrelated pairs are pushed farther apart, demonstrating the intended effect of the contrastive pre-training approach.

The 50 nearest neighbors of one of the toxicity-related words (`bitch`) are presented in Table 16 as an example. A manual inspection was conducted to assess the relevance of these neighboring words to this specific toxicity-related word. The analysis confirmed that the majority of the neighbors are indeed toxicity-related.

Semantic relationships within general vocabulary terms. A selection of randomly chosen synonym and antonym pairs, along with their semantic embedding distances before and after embedding pre-training, is presented in Appendix C.2. As observed, synonym pairs exhibit reduced distances while antonym pairs become more distant in the embedding space after pre-training, demonstrating the intended effect of the contrastive learning approach.

Table 14: Top 50 most ℓ_∞-similar words to `female` and `male` using the Keras embedding layer after pre-training.

Model	**Salary Prediction & Gender Bias Mitigation (8 & 10 blocks)**							
	`female`				**`male`**			
	Word	**Distance (↓)**	**Word**	**Distance (↓)**	**Word**	**Distance (↓)**	**Word**	**Distance (↓)**
	nun	2.0e−04	son	2.0e−04	actress	2.0e−04	woman	2.0e−04
	deer	2.0e−04	busch	2.0e−04	busch	2.0e−04	her	2.0e−04
	gal	2.0e−04	bride	2.0e−04	hostess	2.0e−04	goddess	2.0e−04
	bull	2.0e−04	dad	2.0e−04	gonzales	2.0e−04	dude	2.0e−04
	brothers	2.0e−04	lads	2.0e−04	bro	2.0e−04	maid	2.0e−04
	grandpa	2.0e−04	bro	2.0e−04	brother	2.0e−04	mama	2.0e−04
	earl	2.0e−04	hostess	2.0e−04	chick	2.0e−04	godfather	2.0e−04
	sister	2.0e−04	grandfather	2.0e−04	womb	2.0e−04	feminist	2.0e−04
	momma	2.0e−04	maid	2.0e−04	fiancee	2.0e−04	granny	2.0e−04
	lion	2.0e−04	fiance	2.0e−04	earl	2.0e−04	**female**	3.0e−04
	colts	2.0e−04	**male**	3.0e−04	guy	3.0e−04	businessman	3.0e−04
	brother	3.0e−04	boy	3.0e−04	brothers	3.0e−04	colts	3.0e−04
	widow	3.0e−04	womb	3.0e−04	lion	3.0e−04	chairman	3.0e−04
	stallion	3.0e−04	her	3.0e−04	queen	3.0e−04	she	3.0e−04
	fiancee	3.0e−04	godfather	3.0e−04	monk	3.0e−04	wife	3.0e−04
	father	3.0e−04	mother	3.0e−04	stud	3.0e−04	males	3.0e−04
	grandmother	3.0e−04	madame	3.0e−04	granddaughter	3.0e−04	diva	3.0e−04
	statesman	3.0e−04	dude	3.0e−04	lads	3.0e−04	nuns	3.0e−04
	nuns	3.0e−04	gentlemen	3.0e−04	deer	3.0e−04	man	3.0e−04
	businessman	3.0e−04	guy	3.0e−04	mother	3.0e−04	papa	3.0e−04
	aunt	3.0e−04	him	3.0e−04	women	3.0e−04	kings	3.0e−04
	maternal	3.0e−04	fathers	3.0e−04	statesman	3.0e−04	his	3.0e−04
	himself	3.0e−04	queen	3.0e−04	himself	3.0e−04	spokesman	3.0e−04
	mama	3.0e−04	men	3.0e−04	son	3.0e−04	sons	3.0e−04
	wife	3.0e−04						

Table 15: ℓ_2 and ℓ_∞ distances between word pairs before and after embedding pre-training. Related gendered words become closer, while unrelated pairs grow further apart.

Relation		**Related gendered word pairs**				
Word pair			ℓ_2 **(Before)**	ℓ_2 **(After) (↓)**	ℓ_∞ **(Before)**	ℓ_∞ **(After) (↓)**
man	↔	woman	3.22e−02	5.00e−04	7.59e−02	4.00e−04
king	↔	queen	1.99e−02	7.00e−04	4.94e−02	4.00e−04
boy	↔	girl	2.89e−02	9.00e−04	6.36e−02	5.00e−04
female	↔	male	1.37e−02	4.00e−04	3.27e−02	3.00e−04
father	↔	mother	2.77e−02	5.00e−04	6.93e−02	2.00e−04
husband	↔	wife	1.98e−02	8.00e−04	4.92e−02	6.00e−04
brother	↔	sister	3.05e−02	6.00e−04	7.54e−02	4.00e−04
son	↔	daughter	2.59e−02	5.00e−04	5.66e−02	3.00e−04
uncle	↔	aunt	2.91e−02	8.00e−04	7.21e−02	5.00e−04
nephew	↔	niece	2.78e−02	7.00e−04	6.70e−02	4.00e−04
prince	↔	princess	2.50e−02	7.00e−04	5.92e−02	5.00e−04
monk	↔	nun	3.49e−02	8.00e−04	8.63e−02	7.00e−04
wizard	↔	witch	3.33e−02	7.55e−02	8.20e−02	5.38e−02
female	↔	grandpa	3.39e−02	4.00e−04	8.25e−02	2.00e−04
Relation		**Unrelated word pairs**				
Word pair			ℓ_2 **(Before)**	ℓ_2 **(After) (↑)**	ℓ_∞ **(Before)**	ℓ_∞ **(After) (↑)**
woman	↔	ocean	1.81e−02	2.381e−01	4.37e−02	1.360e−01
boy	↔	guitar	2.10e−02	6.659e−01	4.68e−02	4.209e−01
girl	↔	airplane	2.33e−02	1.557e−01	5.60e−02	1.033e−01
father	↔	bottle	3.11e−02	1.0779e+00	7.97e−02	4.445e−01
mother	↔	keyboard	2.61e−02	5.032e−01	6.79e−02	2.245e−01
husband	↔	river	3.14e−02	4.214e−01	6.76e−02	2.457e−01
wife	↔	planet	2.63e−02	2.126e−01	5.36e−02	1.400e−01
brother	↔	window	2.30e−02	8.35e−02	5.72e−02	6.52e−02
sister	↔	chair	3.07e−02	8.44e−02	7.77e−02	5.12e−02
son	↔	clock	3.26e−02	1.1346e+00	7.90e−02	5.048e−01
daughter	↔	road	1.43e−02	4.082e−01	3.01e−02	2.651e−01
uncle	↔	painting	2.46e−02	7.58e−02	5.71e−02	4.90e−02
aunt	↔	garden	1.34e−02	3.049e−01	2.94e−02	1.480e−01

Table 16: Top 50 most ℓ_∞-similar words to `bitch` using the pre-trained Keras embedding layer.

Relation	**Similar words to `bitch` (ℓ_∞ distance)**				
[PAD]	1.645e−03	##tter	2.569e−03	cock	2.631e−03
##ter	1.897e−03	screwing	2.686e−03	shy	2.710e−03
##kker	2.267e−03	##s	2.804e−03	##holz	2.854e−03
##r	2.320e−03	pack	2.856e−03	##1	3.011e−03
##z	2.409e−03	wan	3.100e−03	##ie	3.137e−03
bi	2.445e−03	##t	3.145e−03	;	3.154e−03
##cks	2.533e−03	*	3.168e−03	##tt	3.179e−03
##us	3.188e−03	##3	3.261e−03	##ra	3.289e−03
+	3.340e−03	fu	3.386e−03	##te	3.416e−03
(	3.431e−03	##i	3.469e−03	##ot	3.471e−03
##or	3.496e−03	##er	3.502e−03	mother	3.514e−03
##ts	3.632e−03	knob	3.662e−03	ti	3.757e−03
##dge	3.806e−03	##rac	3.837e−03	ass	3.874e−03
moth	3.899e−03	pick	3.905e−03	fuck	3.937e−03
sucker	3.976e−03	##ey	3.987e−03	##ore	4.006e−03
ho	4.020e−03	ku	4.116e−03	-	4.118e−03
!	4.165e−03	##it	4.184e−03		

Table 17: ℓ_2 and ℓ_∞ distances between synonym and antonym word pairs before and after embedding pre-training. Synonym pairs become closer, while antonym pairs grow further apart.

Relation			**Synonym word pairs**			
Word pair			ℓ_2 (Before)	ℓ_2 (After) (↓)	ℓ_∞ (Before)	ℓ_∞ (After) (↓)
big	↔	large	2.22e−02	5.50e−03	1.45e−02	2.70e−03
smart	↔	intelligent	3.30e−02	1.50e−02	1.92e−02	8.60e−03
sad	↔	unhappy	3.47e−02	1.68e−02	1.79e−02	9.60e−03
start	↔	begin	2.78e−02	7.60e−03	1.81e−02	4.40e−03

Relation			**Antonym word pairs**			
Word pair			ℓ_2 (Before)	ℓ_2 (After) (↑)	ℓ_∞ (Before)	ℓ_∞ (After) (↑)
fast	↔	slow	3.33e−02	3.236e−01	1.86e−02	1.522e−01
light	↔	dark	2.37e−02	2.741e−01	1.42e−02	1.273e−01
love	↔	hate	2.16e−02	3.820e−01	1.49e−02	2.134e−01
up	↔	down	2.96e−02	8.497e−01	2.10e−02	4.821e−01

Table 18: ℓ_2 and ℓ_∞ distances between word pairs before and after embedding pre-training. Related toxic words become closer, while unrelated pairs grow further apart.

Relation			Related toxic word pairs			
Word pair			ℓ_2 (Before)	ℓ_2 (After) (↓)	ℓ_∞ (Before)	ℓ_∞ (After) (↓)
Mother Fuker	↔	mother-fucker	7.43e−02	2.70e−03	1.806e−01	1.60e−03
blowjob	↔	b17ch	5.18e−02	1.90e−03	1.328e−01	1.20e−03
gayboy	↔	gays	2.23e−02	2.70e−03	5.34e−02	1.60e−03
dick*	↔	dirsa	3.51e−02	5.60e−03	8.74e−02	4.00e−03
Phukker	↔	peceenusss	5.43e−02	4.20e−03	1.168e−01	2.30e−03
puta	↔	pusse	2.42e−02	2.60e−03	5.64e−02	1.70e−03
arse*	↔	azzhole	2.74e−02	3.10e−03	6.95e−02	2.30e−03
chuj	↔	cum	5.74e−02	2.70e−03	1.318e−01	2.50e−03
c0cks	↔	cunt*	5.91e−02	2.50e−03	1.352e−01	1.60e−03
shits	↔	sh1t	6.75e−02	2.50e−03	1.478e−01	1.50e−03
skanks	↔	slutty	4.73e−02	3.50e−03	1.069e−01	1.80e−03
fuckin	↔	faget	3.44e−02	3.20e−03	8.28e−02	2.20e−03
cawks	↔	cock	8.18e−02	3.10e−03	2.174e−01	1.90e−03
niiger	↔	nigger	3.25e−02	2.70e−03	8.01e−02	1.80e−03
asshole	↔	assface	5.45e−02	3.00e−03	1.339e−01	2.70e−03
sh1t	↔	screw	7.62e−02	3.60e−03	1.779e−01	2.80e−03
b00bs	↔	b00b*	3.26e−02	4.00e−03	8.77e−02	3.10e−03
slut	↔	sh1tter	5.62e−02	3.00e−03	1.412e−01	1.80e−03
faig	↔	fukker	3.13e−02	2.10e−03	6.83e−02	1.00e−03
Relation			**Unrelated toxic word pairs**			
Word pair			ℓ_2 (Before)	ℓ_2 (After) (↑)	ℓ_∞ (Before)	ℓ_∞ (After) (↑)
orgasum	↔	constructive eviction	5.14e−02	2.743e−01	1.086e−01	1.215e−01
b!tch	↔	floating bridge	7.04e−02	9.92e−02	1.685e−01	5.74e−02
cock	↔	biodegradable pollution	9.44e−02	5.664e−01	2.417e−01	2.841e−01
fukker	↔	karyolysis	8.21e−02	4.461e−01	1.928e−01	2.506e−01
bitches	↔	fried egg	5.57e−02	3.218e−01	1.360e−01	1.412e−01
penus	↔	novelette	4.47e−02	3.331e−01	9.87e−02	1.576e−01
kike	↔	Niobrara River	5.22e−02	5.802e−01	1.128e−01	3.029e−01
b17ch	↔	dowery	5.86e−02	3.491e−01	1.359e−01	1.722e−01
fukkah	↔	fish house punch	4.83e−02	2.207e−01	1.155e−01	1.266e−01
basterds	↔	epenthesis	4.85e−02	3.939e−01	1.156e−01	2.164e−01
fukk	↔	dry-wood termite	7.46e−02	6.182e−01	1.712e−01	2.897e−01
shyte	↔	Monrovia	5.60e−02	3.725e−01	1.277e−01	2.017e−01
nazis	↔	potter around	3.59e−02	2.814e−01	8.77e−02	1.552e−01
cock-sucker	↔	kind of	4.78e−02	2.892e−01	1.116e−01	1.599e−01
fukkin	↔	process of monition	5.62e−02	3.600e−01	1.249e−01	2.670e−01