Wafflecone: Measuring Bias through the Latent Space Exploration

Anonymous ACL submission

Abstract

The increasing role of large language models (LLMs) in critical social functions, from influencing decision making to shaping public opinion, highlights the urgent need for effective mechanisms to detect bias within these systems. Ensuring the impartiality of LLMs is essential to prevent perpetuation of existing social biases. Although strides have been made in identifying and quantifying biases, particularly those related to gender and race, much of the research concentrates on binary biases, and some of the works often rely on downstream tasks to evaluate bias. Our paper contributes to this field by proposing a novel, computationally efficient method "Wafflecone" for bias evaluation that utilizes the latent space of the model, and does not require tailored adjustments for specific models or extensive lists of biased terminology. This method is adept at assessing a range of biases, both binary and non-binary, thereby expanding our understanding of how biases manifest in different language model embeddings.

1 Introduction

The quest for fairness in artificial intelligence has become a critical issue as large language models (LLMs) increasingly intersect with vital areas of society, such as influencing decision-making processes and public opinion. The risk of LLMs perpetuating and even amplifying social biases cannot be overstated, and this threatens to deepen social disparities and establish prejudicial norms. Our research introduces "Wafflecone", a methodology for bias detection that leverages latent space for analysis without the need for specific model tuning or exhaustive lists of bias indicators.[1]

Our proposed methodology advances bias detection and evaluation beyond the conventional binary approach to capture a more nuanced spectrum of bias manifestations within language model embeddings. The Wafflecone framework is designed to assess bias across a broad spectrum of language, and particularly focuses on words typically expected to be free from bias. Unlike previous studies that predominantly examined bias in professional titles, this framework extends the scope of evaluation to include seemingly neutral tokens, such as "a", "is", "good", "the" and ".", because these tokens are expected as bias-free. By incorporating these additional elements, Wafflecone is able to detect more subtle forms of bias.

Furthermore, we introduce an innovative metric termed "ideal similarity", which acts as a normative benchmark for gauging bias. This metric is instrumental in standardizing bias quantification across different bias-related groups. It is important to note that, in our experimental setup, embeddings can be derived from any layer of the model, including contextualized layers, where they represent contextual embeddings. At the core of Wafflecone analysis is the examination of disparities in similarity between biased token embeddings and those deemed neutral. Our approach is based on the concept that embeddings from bias-specific sentences (e.g., "He is an Engineer" vs. "She is an Engineer") should be impartial. For example, an unbiased scenario is one where the similarity between "Engineer" and "He" equals that between "Engineer" and "She," as shown in Panel (a) of Fig. **(a)** in Fig. 1. In contrast, scenarios where the similarity between "Engineer" and "He" is disproportionately greater than that between "Engineer" and "She," evidencing an underlying bias are presented in Panels **(b)** in Fig. 1.

In addition, a prevalent challenge in current bias detection frameworks is the dependency on downstream tasks for evaluating biases in language models, often necessitating additional performance-oriented tasks such as sentiment analysis, machine translation, and gender classification. The conven-

[1]The code is available on GitHub at Anonymous.

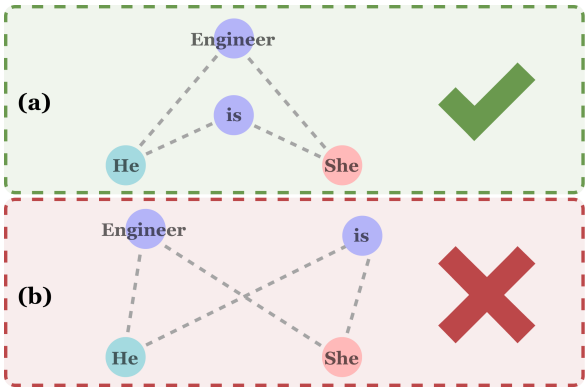

Figure 1: The Wafflecone framework is built on the assertion that the unbiased semantic representation of professions should be unprejudiced with respect to gender. This is visually articulated in the embeddings from the sentences "He is an Engineer" and "She is an Engineer".

tional approach requires the construction of additional machine learning models specifically for task specific bias assessment, as noted in the literature (Stanczak and Augenstein, 2021; Jentzsch and Turan, 2022; Bhardwaj et al., 2020). Such a requirement not only demands significant computational resources, but also limits the scope of bias detection to the tasks for which the model has been fine-tuned. For instance, a model calibrated for binary gender bias detection may prove ineffective in identifying racial biases. Moreover, each added model introduces greater potential for imprecision. However, our proposed methodology diverges from this paradigm by avoiding the necessity for task-specific fine-tuning and additional modeling. Instead, our method operates directly within the high-dimensional embedding space, shortening the process and broadening the detectable range of biases.

Moreover, although some current techniques use embedding to evaluate bias, such as the Word Embedding Association Test (WEAT) and Log Probability Bias Score (Kurita et al., 2019), they require a long list of pre-defined words. For example, to evaluate bias in the context of binary gender using the WEAT score, four sets of words must be defined, including two sets of contrasting gender items and two sets of contrast attributes (Nemani et al., 2023). In contrast, well tailored for users aiming to pinpoint specific biases, our approach requires only keyword sets associated with the bias-group-related subspace. For instance, by identifying and segregating all male-related and female-related words within the corpus, two distinct subspaces can be constructed. This approach simplifies the process by eliminating the need for complex steps like fine-tuning or constructing new models, focusing solely on the predefined small number of tokens for each subspace related to the type of bias they want to evaluate.

In short, the contributions of our approach, Wafflecone, can be summarized as follows: **(a) Increased Computational Efficiency:** It does not require computational resources for model finetuning. It operates within the high-dimensional embedding space of the experimental dataset, eliminating the need to construct support models or specific model adaptations. **(b) Inclusive Non-Binary Bias Evaluation:** It is equipped to address both binary and non-binary biases, enabling more complex bias analyses and exploration of language models. **(c) Multi-Dimensional Bias Consideration:** This approach enables the examination of bias not only across different groups but also between all bias-related groups and unrelated tokens. **(d) Broader Bias Comparisons:** It offers the capability to compare biases across different models and layers.

2 Related Works

The study of binary bias detection in transformer-based large language models or in pre-transformer works, particularly focusing on gender bias, has garnered significant attention in recent years. In works such as Caliskan et al. (2022), they examined gender bias with GloVe and fastText embeddings. The works of Jentzsch and Turan (2022), who utilized the IMDB dataset and sentiment ratings in downstream classification tasks to measure and analyze biases. It is important to note that the detection and analysis of biases in these studies are tightly linked to the nature of the downstream tasks, requiring additional models and model fine-tuning. In addition, some survey studies, such as the one conducted by Stanczak and Augenstein (2021), have highlighted that much of the research in this area has focused on detecting gender bias within a strictly binary framework.

With the increase of the importance of evaluating bias in non-binary contexts, like gender-inclusive research, there is an increasing trend in the academic community toward expanding the scope of bias detection methodologies. The study by Felkner et al. (2023) presents a benchmarking of various models, using their dataset, WinoQueer, to

evaluate the biases that are harmful to the LGBTQ+ community. This trend encompasses not only gender bias but also other forms of bias, such as ethnic bias, as explored in the research by Ahn and Oh (2021). The work of Esiobu et al. (2023) introduces the *BiasScore* metric, a percentage of subgroups within a dataset for which it cannot be conclusively established whether their exposure to negative sentiment or toxicity surpasses a specified standard level, and they benchmarked various prompt-based bias and toxicity metrics across multiple demographic axes and evaluates the effectiveness of bias and toxicity mitigation techniques in LLMs. Also, Barikeri et al. (2021) have developed a conversational dataset, RedditBias, grounded in actual human conversations from Reddit. This dataset allows for the measurement and mitigation of non-binary bias across four critical dimensions: gender, race, religion, and queerness.

3 Methodology

In this section, we provide a detailed description of the methodology.

3.1 Data Preparation

In data preparation, the focus is on creating a dataset of template sentences and anchor word lists for various bias groups or attributes, based on the bias evaluation task.

Every template sentence should have m versions instantiated, each illustrating a different bias group where m is the number of categories of bias under examination, and $m \geq 1$, $m \in N$. For example, in the Panel **(a)** of Fig. 2, each template sentence is expressed as three unique variants ($m = 3$).

The anchor word list A_j is a list of words related to the jth bias-related group, and there are no shared words among anchor word lists i.e. $A_1 \cap A_2 \cap \ldots A_m = 0$. Moreover, all anchor word lists are non-empty and contain the same number of anchor words, i.e. $|A_1| = |A_2| \cdots = |A_m| \geq 1$. The anchor word lists ($A_1 \ldots A_m$) are selected from the corpus ($corpus = A_1 \cup A_2 \cup \ldots A_m \cup \ldots W$, where W refers to all other tokens in the corpus other than anchor words, which are expected to be bias-free). Panel **(b)** of Fig. 2 shows the token lists for A_1, A_2, A_3 and W.

It is important to note that the representation of each token is uniquely defined by a combination of the token itself, its line number, and its position number. This ensures that identical tokens have

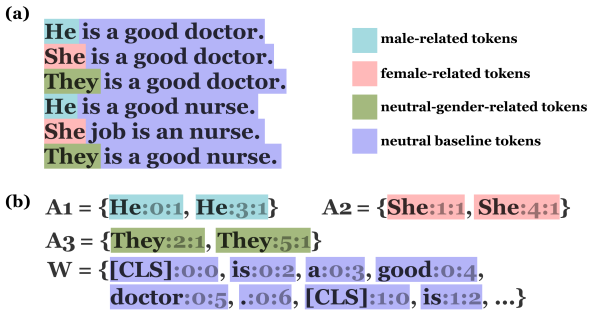

Figure 2: An example of token categorization. **(a)** Tokens associated with male identifiers are marked in blue, female identifiers in pink, and gender-neutral or baseline tokens in purple. The neutral tokens serve as a benchmark for unbiased context and do not correspond to gendered categories. **(b)** Contextual variability of token embeddings is depicted. Tokens are indexed and differentiated by their content, line number, and sequence position within a given context.

distinct vector embedding in varying contexts. For example, the token "He" in the first sentence has a different representation, and therefore a different contextual embedding than the token "He" in the fourth sentence.

Definition 1 (Space)**.** The high-dimensional space S encompasses the vector representations of the set of all tokens W present within the corpus.

Definition 2 (Bias-related Group)**.** The m classes of bias, as defined by the users, contribute collectively to the comprehensive bias. Each class represents a specific aspect, with examples like the male and female groups together constituting the broader category of gender bias.

Definition 3 (Anchor Word)**.** The ith word from the anchor word list A_j, specifically selected to represent the characteristics or attributes of the jth bias-related group within a corpus. It has embedding e_{ij}.

3.2 Subspace and Subspace Centers' Generation

This subsection discusses generating the subspace and, hence, the respective center for each bias group for bias evaluation. If the subspace S_j for the jth bias-related group is a subset of the high-dimensional space (S) containing all $|A_j|$ tokens of the anchor word list A_j, and each token i in S_j can be represented as an embedding vector e_{ji}, then the center C_j of the subspace S_j is calculated as the average of all the embeddings e_{ji}.

Hence, the centers of all m subspaces are calculated based on Eq. 1. Subsequently, new subspace S_{center} is generated, which consists of all the center points $C_1, C_2, \ldots, C_m$. Its center, C_{center}, is represented as the average of all these center points.

For the proper evaluation of bias, C_{center} should have an identical distance to each subspace center C_j in $S_{centers}$, as it allows us to compare the distance between each token and the subspace centers C_j to determine the bias of the token.

Revisiting the earlier example with three bias categories; The embedding vectors of the anchor word lists, A_1, A_2, and A_3, are used to generate three subspaces, as illustrated in Fig. 3. The first subspace, designated as S_1, is associated with the male gender group and contains the embeddings of all tokens in A_1. Subspace S_2 analogously, contains the embeddings of all tokens in A_2 and pertains to the female gender group, while subspace S_3 comprises embeddings for A_3 and corresponds to the neutral gender group. In Fig.3, Panel **(a)** and Panel **(b)** visualize the subspaces, **(c)** facilitate the representation of token aggregates through their respective mean vectors.

Definition 4 (Subspace)**.** A high-dimensional space, denoted as S_j, formed by the vector representations of the tokens in A_j, and specifically associated with the jth bias-related group.

Definition 5 (Subspace Center)**.** The center, donated by C_j, of the high-dimensional space S_j, is determined by calculating the average of all vector representations of tokens in A_j.

$$C_j = \frac{1}{|A_j|} \sum_{i=1}^{|A_j|} e_{ji} \tag{1}$$

Definition 6 (Center of the Subspace Centers)**.** The center of the high-dimensional space is defined by the center of all the subspace centers $C_1, C_2, \ldots, C_m$.

$$C_{center} = \frac{1}{m} \sum_{j=1}^{m} C_j \tag{2}$$

3.3 Ideal Similarity Calculation

This step aims to establish a theoretical baseline for similarity in an unbiased context. This is achieved by measuring the cosine similarity between each token in W and the adjusted idealized norm C_{center}. This measure provides a standard against which we can normalize bias measurements, allowing for consistent comparisons across different models and embedding layers. This concept is visually represented by yellow dashed lines in Panel **(d)** of Fig. 3 and is a part of the "Ideal Similarity Calculation" subcomponent within the Bias Detector, as illustrated in Fig. 6.

Definition 7 (Ideal Similarity)**.** A theoretical value, denoted as IS_j and calculated by the cosine similarity from C_{center} to C_j, is utilized as the bias-free value of cosine similarity for subspace S_j with respect to all bias-unrelated tokens.

$$IS_j = \cos(C_j, C_{center}) \tag{3}$$

Where IS_j represents the ideal similarity between the subspace center C_j and all tokens in W, and $\cos(C_j, C_{center})$ computes the cosine similarity between vectors C_j and C_{center}.

3.4 Bias Evaluation for Single Group

In this step, we will evaluate and report the bias quantitatively. Bias evaluation is performed using the adjusted subspace centers and ideal similarities from previous steps. This evaluation entails a comparison of the actual similarity between each token within W and the centroid of the adjusted subspace centers C_j, against the theoretically derived ideal similarity values. The bias for the bias-related group j, denoted as $bias_j$, is calculated as the average cosine similarity between C_{center} and each W_i.

Keeping the positive and negative signs when calculating the difference between the adjusted subspace centers and ideal similarities may be beneficial in a binary bias evaluation task when the bias direction is important. For example, evaluating binary-gender bias in a model. However, if the bias direction is not important in the context of the evaluation task, or if the bias evaluation task is not binary, the bias value can instead be calculated as an absolute value.

This particular step is the subcomponent "Bias evaluation for a single group" in the Bias Detector diagram illustrated in Fig. 6.

Definition 8 (Non-normalized Bias for Single Group)**.** The non-normalized bias related to the jth bias-related group, donated as $\widehat{bias}_j$, is quantified as the sum of all differences between the actual cosine similarity from the subspace center C_j to all

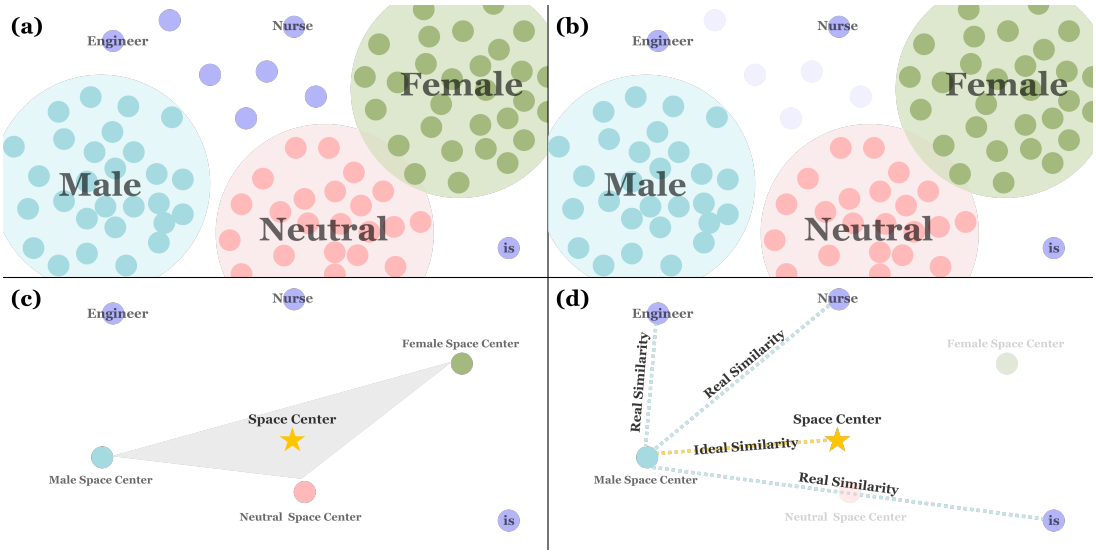

Figure 3: Illustrates of the methodology for assessing bias in binary and nonbinary gender categories within a linguistic framework. Panel **(a)** introduces three semantic subspaces, each color-coded: blue for male-associated terms, green for female-associated terms, and pink for nonbinary-associated terms. The gender-unrelated terms "Engineer", "Nurse", and "is" are presented as purple nodes. Panel **(b)** the centers of the three gender-related subspaces (male, female, and non-binary) are used to construct a comprehensive subspace. The center of this composite subspace is marked by a yellow star, as shown in panel **(c)**. Panel**(d)** quantifies bias through cosine similarity measures: the "ideal similarity" (yellow dashed line) represents a theoretical unbiased standard, while the "real similarity" (blue dashed line) reflects actual term associations.

bias-unrelated tokens, and its ideal similarity IS_j.

$$\widehat{bias}_j = \frac{1}{|W|}\sum_{i=1}^{|W|}|\cos(W_i, C_j) - IS_j| \quad (4)$$

If the direction of the bias is critical to the analysis, the calculation will adjust to include the sign of each bias value.

$$\widehat{bias}_j^{+/-} = \frac{1}{|W|}\sum_{i=1}^{|W|}[\cos(W_i, C_j) - IS_j] \quad (5)$$

Where $|W|$ is the number of tokens in W.

3.5 Normalization

Given that one of the main tasks of our approach is to compare the bias between different models, we also need to normalize bias values to make them comparable. Although bias values are bounded between $[-1, 1]$, due to our use of cosine similarity, these values cannot be compared directly between different models because of potential variations in embedding scales. To address this, we normalize the bias values.

In order to avoid changing the original embedding space, as well as to take spatial scale into account, the normalization process is not carried out on the embedding vectors. Instead, we normalize the bias values by dividing the bias value for the jth bias group by the average cosine similarity between its adjusted center C_j and the embedding vectors of all non-anchor words in W.

After normalization, the calculated bias values range from 0 to infinity, where 0 indicates the absence of bias and values greater than 1 indicate a bias intensity higher than the average level of bias across the corpus.

Definition 9 (Bias for Single Group)**.** The normalized bias for the jth bias-related group, denoted as $bias_j$, is calculated by dividing the non-normalized bias value $\widehat{bias}_j$ by the average cosine similarity between the subspace center C_j and the embedding vectors of all non-anchor words in W.

$$bias_j = \frac{\widehat{bias}_j}{\frac{1}{|W|}\sum_{w \in W}\cos(w, C_j)} \quad (6)$$

3.6 Bias interpretation

Bias interpretation is performed on the bias values and normalized bias values that are generated in the preceding stages. Here, the user has the flexibility to tailor the final bias value based on user-specific requirements, ensuring a personalized and relevant evaluation. It serves as the capstone of our bias

assessment by translating numerical data into actionable insights.

Delving deeper into this process, we consider two distinct dimensions of bias. The first dimension involves an inter-group bias analysis, where we examine the disparities between different groups, such as male versus female. The second dimension shifts focus to the comparison between all bias-related groups and unrelated tokens, exemplified by contrasting all gender-related tokens with those unrelated to gender. The standard deviation over the groups represents the inter-group bias, and the magnitude of the numerical value signifies the extent of bias between all bias-related groups and unrelated tokens. The final bias value is derived by integrating these two dimensions.

The final bias value is calculated as the weighted average, which combines the sum of normalized bias values across all bias-related groups and the standard deviation of these normalized bias values, focusing only on the magnitude and ignoring the direction of bias by default. The weights, as hyperparameters, assigned to each component are tailored to signify their respective significance. For consistency, all experiments presented in this paper will treat these components as equally important.

The entire model pipeline is detailed in Fig. 6.

Definition 10 (Bias)**.** The bias for the language model regarding all the bias-related groups is defined as the sum of normalized bias values across all bias-related groups, and the standard deviation:

$$bias = \frac{\lambda_1 \cdot \sum_{j=1}^{m} bias_j + \lambda_2 \cdot \sigma(bias_1, \ldots, bias_j)}{\lambda_1 + \lambda_2} \tag{7}$$

Where λ_1 and λ_2 are two hyperparameters, λ_1 defines the importance of bias with respect to different groups, and λ_2 determines the significance of all bias-related tokens in comparison to other tokens. $\sigma(bias_1, \ldots, bias_j)$ represents the standard deviation of all individual-group bias.

4 Experiments

This section introduces the experiments we conducted to evaluate the bias in different embedding-based language models. It is important to note that in this section, "bias" exclusively denotes "normalized bias" unless otherwise specified.

	Bias in Language Model		
Layer No.	bert-base-uncased	mabel-bert-base-uncased	Bias Difference
0	1.5821	1.6884	-0.1064
1	1.1790	1.0067	0.1723
2	0.8200	0.6401	0.1798
3	0.9072	0.6992	0.2080
4	0.7552	0.5593	0.1958
5	0.5970	0.4406	0.1565
6	0.6241	0.4328	0.1913
7	0.6560	0.4427	0.2132
8	0.6060	0.3651	0.2409
9	0.6868	0.3746	0.3122
10	0.5836	0.3555	0.2281
11	0.4115	0.2472	0.1643
12	0.5224	0.3139	0.2084

Table 1: Assessment of bias values for bert-base-uncased and mabel-bert-base-uncased using the WinoBias dataset.

4.1 Models

As the major part of our experiments, we will evaluate the bias in the following embedding-based language models: albert-base-v2 (albertB; Lan et al., 2020), albert-large-v2 (albertL; Lan et al., 2020), albert-xxlarge-v2 (albertXXL; Lan et al., 2020), bert-base-cased (bertB; Devlin et al., 2019), bert-large-cased (bertL; Devlin et al., 2019), distilbert-base-cased (distilbert; Sanh et al., 2020), roberta-base (robertaL; Liu et al., 2019), roberta-large (robertaL; Liu et al., 2019), xlm-roberta-base (xlmB; Conneau et al., 2020), xlm-roberta-large (Conneau et al., 2020, xlmL;). All these models are available as pre-trained models from HuggingFace Transformers.

4.2 Datasets

We use three datasets for bias evaluation, namely WinoBias (Zhao et al., 2018), Winogender (Rudinger et al., 2018), and Equity Evaluation Corpus (EEC) (Kiritchenko and Mohammad, 2018) dataset. WinoBias offers 3,160 sentences designed to test semantic and syntactic challenges, while the Winogender dataset, with its 720 sentences derived from 120 templates, focuses on identifying gender biases in coreference resolution. The EEC, which encompasses 8,640 sentences, aims to detect biases in sentiment analysis, particularly in relation to race and gender.

4.3 Results

It is essential to demonstrate the effectiveness of our approach in discerning bias disparities between

a base model and a debiased version before conducting bias evaluations across various models. For this purpose, we employ the bert-base-uncased model (Devlin et al., 2019), along with its gender-debiased version, mabel-bert-base-uncased (He et al., 2022). Comparative analysis, as defined in Table 1, substantiates the efficacy of our pipeline in detecting mitigated bias in the variant of the debiased model.

4.3.1 Bias Comparisons Across Different Models

The results of the bias calculation for the different language models are shown in Fig. 4 and Table 2. The bias value is calculated based on Eq. 7. We found that xlmB has the lowest bias value among all the models evaluated on the three datasets; In contrast, bertL and albertXXL have the highest bias.

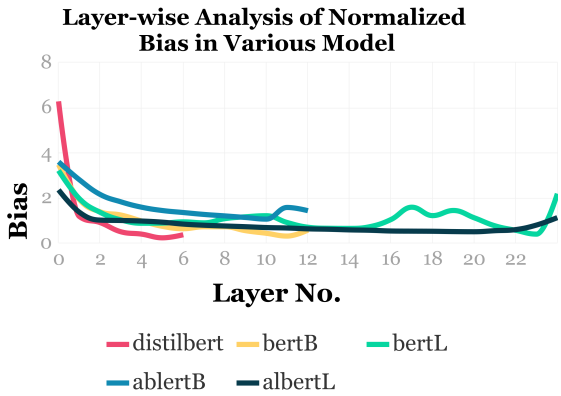

Figure 4: The graph provides an analysis of bias magnitudes within various computational models with the WinoBias dataset. Sequential layers are plotted along the x axis, and their corresponding normalized bias metrics are charted on the y axis.

Language Model	WinoBias	Winogender	EEC
albertB	0.7245 (9)	0.8076 (8)	0.6286 (8)
albertL	0.5639 (8)	0.6193 (7)	0.5541 (7)
albertXXL	**0.9079 (10)**	0.8526 (9)	0.6373 (9)
bertB	0.2833 (5)	0.2921 (6)	0.2440 (6)
bertL	0.3679 (7)	**1.0653 (10)**	**1.2833 (10)**
distilbert	0.3582 (6)	0.1928 (5)	0.1805 (5)
robertaB	0.0605 (4)	0.0581 (4)	0.0549 (4)
robertaL	0.0163 (3)	0.0151 (3)	0.0083 (2)
xlmB	0.0062 (1)	0.0066 (1)	0.0073 (1)
xlmL	0.0111 (2)	0.0111 (2)	0.0111 (3)

Table 2: Consolidated normalized spaces and biases for gender terms across different language models. Rank values are enclosed in brackets. Bold indicates the highest bias, and italic indicates the lowest bias.

In our analysis, we observed that models with greater complexity and size, exemplified by bertL, albertL, and albertXXL, manifested more substantial bias scores compared to their smaller counterparts, such as bertB, albertB, and distilbert. This trend suggests a correlation between the size of the model and the extent of encoded biases, aligning with previous empirical findings, such as by Jentzsch and Turan (2022), that larger models may amplify underlying biases present in their training data.

We also observed that the bias rank of the models was not always consistent across the different datasets, regardless of whether the bias values are normalized from Table 2 and Table 3. Such a pattern suggests the distinct characteristics inherent in each dataset, aligning with the pattern reported by Vig et al. (2020) in their study.

In the normalization process, it became apparent that this procedure does not significantly alter the ranking scores among the models under scrutiny, a fact that is corroborated by the data presented in Table 2 and Table 3. However, it's noteworthy that the actual bias values exhibited marked variations pre- and post-normalization, especially in models with higher ranks in both tables.

Language Model	WinoBias	Winogender	EEC
albertB	0.2777 (9)	0.2889 (8)	0.2463 (8)
albertL	0.2424 (8)	0.2511 (7)	0.2341 (7)
albertXXL	**0.3191 (10)**	0.3101 (9)	0.2664 (9)
bertB	0.1730 (5)	0.1760 (6)	0.1528 (6)
bertL	0.2178 (7)	**0.3106 (10)**	**0.3579 (10)**
distilbert	0.2037 (6)	0.1322 (5)	0.1216 (5)
robertaB	0.0532 (4)	0.0515 (4)	0.0488 (4)
robertaL	0.0158 (3)	0.0147 (3)	0.0082 (2)
xlmB	0.0060 (1)	0.0064 (1)	0.0069 (1)
xlmL	0.0109 (2)	0.0108 (2)	0.0109 (3)

Table 3: Consolidated non-normalized spaces and biases for gender terms across different language models. Rank values are enclosed in brackets. Bold indicates the highest bias, and italic indicates the lowest bias.

More detailed results for bias change over the models are presented in Table 5, Table 6, and Table 7.

4.3.2 Bias Comparisons Across Different Layers of a Model

The progression of binary gender bias values across successive layers for the five models is shown in Fig. 4 using the WinoBias dataset. The data reveals a pronounced decrease in bias with increasing layer

levels; moreover, the inaugural layer is characterized by a notably augmented bias compared to its subsequent counterparts. This pattern suggests a concentration of bias encoding within the initial layer, and potential mitigation or modification of such biases occurring in the deeper strata of the model architectures. Another interesting observation is that the bias increases in the last layer of the model. Specifically, the final layer, known as the prediction layer, shows an increase in bias compared to the previous layer.

Rank	Language Model	Bias	Rank	Language Model	Bias
1	xlmB	0.0064	6	bertB	0.2926
2	xlmL	0.0109	7	albertL	0.5974
3	robertaL	0.0148	8	albertB	0.7378
4	robertaB	0.0568	9	albertXXL	0.7901
5	distilbert	0.1933	10	bertL	1.0109

Table 4: Non-binary bias comparison results for different language models with WinoBias dataset. The lower the ranking, the lower bias.

The Winogender dataset categorically encapsulates gender with a binary classification of "female" and "male," occasionally extending to a non-binary "neutral" category. This provides a straightforward way to assess the bias of more than two classes and gender biases that transcend conventional binary distinctions. The detailed bias results for each bias-related group are shown in Table 4. Models are ranked in increasing order according to their bias value. Comparing these findings with those in Table 2 shows consistency in bias rankings, implying that the biases inherent in these models may extend beyond binary gender constructs to the non-binary framework. Furthermore, this alignment evidences the effectiveness of our methodology in assessing non-binary bias.

The results of in-depth experiments, showcasing how bias varies across the model's various layers, are presented in Tables 8 and 9.

4.3.3 Bias Evaluation with PCA

Principal Component Analysis (PCA) is a widely applied technique for reducing dimensionality in high-dimensional datasets. We performed experiments with PCA to reduce the dimensionality of data and explore its impact on bias evaluation results.

After performing PCA in the WinoBias dataset, the bias results are illustrated in Fig. 5. For each model under investigation, we reduced the dimensionality in steps from the original number of dimensions down to two dimensions. Our findings reveal a notable pattern: as dimensionality decreases, the normalized bias initially exhibits a significant increase, which is followed by a decreasing trend. Moreover, we observed that the ranking order of models in terms of bias becomes inconsistent. This phenomenon suggests the influence of the PCA technique on bias evaluation is likely attributable to the loss of information that typically accompanies dimensionality reduction. This outcome aligns with expectations, considering that PCA simplifies the embedding space, thereby rendering the previously richly detailed dataset less capable of representing the targeted information in a condensed embedding space.

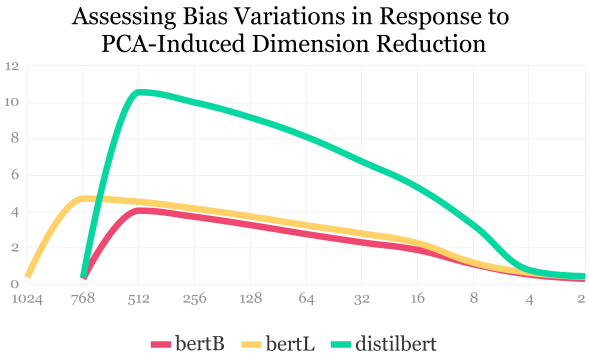

Figure 5: Analysis of Bias Trends in Language Models Following PCA-Based Dimensionality Reduction.

A few more experiments are shown in Fig. 7 and detailed in Table 10.

5 Conclusion

In this work, we present Wafflecone, an innovative bias evaluation framework for probing biases within embedding-based language models. The versatility of Wafflecone lies in its capacity to perform comprehensive bias assessments across a spectrum of models—including, but not limited to, BERT, RoBERTa, ALBERT, and DistilBERT, without necessitating fine tuning or constructing supplementary models. Through rigorous experimentation, we explore the dynamics of bias across various model layers and conduct comparative analyses of different models. Developed in Rust for robust performance, Wafflecone is readily accessible as an open-source Python library on GitHub, and we invite further collaboration and innovation in the field.

6 Limitations

A major limitation of our current research is that our approach, Wafflecone, has not been extensively tested on a wide range of non-binary tasks. Within the scope of future work, we intend to extend our investigation into bias by conducting a wider series of experiments focused on non-binary tasks, including various gender and racial identities. This initiative aligns with our commitment to accurately reflect the rich diversity of genders in our research. Our focus on "male," "female," and neutral identifiers such as "they" was driven by the limitations of available datasets, not by our research philosophy. We are actively working on methodologies that will allow our studies to encompass the entire spectrum of gender identities. Ultimately, our goal is to develop a language model that embodies the inclusivity and diversity of human society, honoring the unique gender identity of each individual.

More models and datasets should also be tested to explore the necessity of normalization, since we tentatively found that normalization only affected the assessed bias value and not the ranking. Moreover, in our current experiments, we have specifically excluded certain tokens, such as [CLS], <SEP>, <s>, and </s>, during the bias detection phase. These tokens may have an understated but significant role in bias evaluation. Investigating their influence in future research is crucial, and could potentially provide more profound insights.

It is also imperative to further validate our approach's efficacy in identifying bias reduction in debiased models. Here, we only evaluated mabel-bert-base-uncased, a debiased iteration of the bert-base-uncased model. While our findings indicate a lower bias level in mabel-bert-base-uncased compared to its predecessor, further testing with a broader range of debiased models is crucial.

In future research, we plan to conduct a comprehensive series of experiments to assess bias in more complex models, including, but not limited to GPT-2, GPT-3, T5, and Llama. In particular, our pipeline design facilitates straightforward assessment and comparison of biases across different models, allowing for a consistent benchmarking process. As we expand our testing to include a greater number of embedding-based models, the results of these efforts will be aggregated and released as an established benchmark for the assessment of bias.

References

Jaimeen Ahn and Alice Oh. 2021. Mitigating Language-Dependent Ethnic Bias in BERT. ArXiv:2109.05704 [cs].

Soumya Barikeri, Anne Lauscher, Ivan Vulić, and Goran Glavaš. 2021. RedditBias: A Real-World Resource for Bias Evaluation and Debiasing of Conversational Language Models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1941–1955, Online. Association for Computational Linguistics.

Rishabh Bhardwaj, Navonil Majumder, and Soujanya Poria. 2020. Investigating Gender Bias in BERT. ArXiv:2009.05021 [cs].

Aylin Caliskan, Pimparkar Parth Ajay, Tessa Charlesworth, Robert Wolfe, and Mahzarin R. Banaji. 2022. Gender Bias in Word Embeddings: A Comprehensive Analysis of Frequency, Syntax, and Semantics. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '22, pages 156–170, New York, NY, USA. Association for Computing Machinery.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. ArXiv:1911.02116 [cs].

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. ArXiv:1810.04805 [cs].

David Esiobu, Xiaoqing Tan, Saghar Hosseini, Megan Ung, Yuchen Zhang, Jude Fernandes, Jane Dwivedi-Yu, Eleonora Presani, Adina Williams, and Eric Michael Smith. 2023. ROBBIE: Robust Bias Evaluation of Large Generative Language Models. ArXiv:2311.18140 [cs].

Virginia K. Felkner, Ho-Chun Herbert Chang, Eugene Jang, and Jonathan May. 2023. WinoQueer: A Community-in-the-Loop Benchmark for Anti-LGBTQ+ Bias in Large Language Models. ArXiv:2306.15087 [cs].

Jacqueline He, Mengzhou Xia, Christiane Fellbaum, and Danqi Chen. 2022. MABEL: Attenuating Gender Bias using Textual Entailment Data.

Sophie Jentzsch and Cigdem Turan. 2022. Gender Bias in BERT - Measuring and Analysing Biases through Sentiment Rating in a Realistic Downstream Classification Task. In *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, pages 184–199, Seattle, Washington. Association for Computational Linguistics.

Svetlana Kiritchenko and Saif M. Mohammad. 2018. Examining Gender and Race Bias in Two Hundred Sentiment Analysis Systems. ArXiv:1805.04508 [cs].

Keita Kurita, Nidhi Vyas, Ayush Pareek, Alan W Black, and Yulia Tsvetkov. 2019. Measuring Bias in Contextualized Word Representations. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 166–172, Florence, Italy. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. ArXiv:1909.11942 [cs].

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. ArXiv:1907.11692 [cs].

Praneeth Nemani, Yericherla Deepak Joel, Palla Vijay, and Farhana Ferdousi Liza. 2023. Gender Bias in Transformer Models: A comprehensive survey. ArXiv:2306.10530 [cs].

Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. 2018. Gender Bias in Coreference Resolution. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 8–14, New Orleans, Louisiana. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. ArXiv:1910.01108 [cs].

Karolina Stanczak and Isabelle Augenstein. 2021. A Survey on Gender Bias in Natural Language Processing. ArXiv:2112.14168 [cs].

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Simas Sakenis, Jason Huang, Yaron Singer, and Stuart Shieber. 2020. Causal Mediation Analysis for Interpreting Neural NLP: The Case of Gender Bias. ArXiv:2004.12265 [cs].

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018. Gender Bias in Coreference Resolution: Evaluation and Debiasing Methods. ArXiv:1804.06876 [cs].

A Supporting Figures

Wafflecone pipeline illustration is illustrated in Fig. 6.

B Experiments Figures

The comprehensive experiment results of bias changing with PCA-Based Dimensionality Reduction are visually presented in Fig. 7.

C Experiments Tables

Additional results of the experiments are presented in Tables 5, 6, 7, 8, 9 and 10. Within some of these tables, the highest bias values across each column are highlighted in bold, while the lowest bias values are emphasized with underlining.

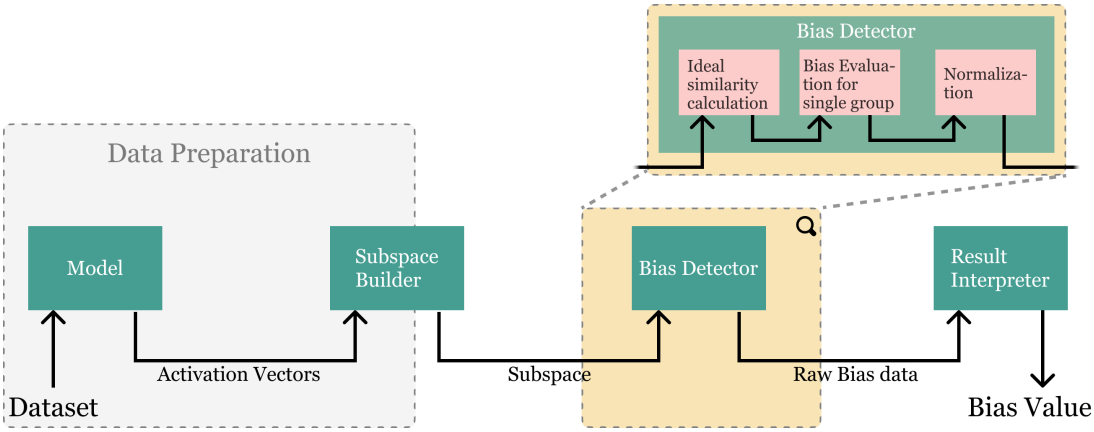

Figure 6: Wafflecone Overview. **Dataset**: The corpus of textual data input to the model. **Model**: The primary language representation model provides word embeddings, such as BERT-based-cased. **Activation Vectors**: High-dimensional token representations that encapsulate semantic relationships within the text corpus. These vectors collectively define the embedding space. **Subspace Builder**: Constructs subspaces from processed token pairs to perform nuanced bias analyses and then formulates a centralized point in the embedding subspaces, streamlining the bias detection process. This central point is integrated into the corresponding subspace to form the **Modified Subspace**. **Bias Detector**: Evaluates the modified subspace to detect and measure embedded biases, yielding **raw bias data** indicative of potential problems. **Result Interpreter**: Converts raw bias data to a normalized **bias value**, providing a quantifiable measure of bias across various layers or models.

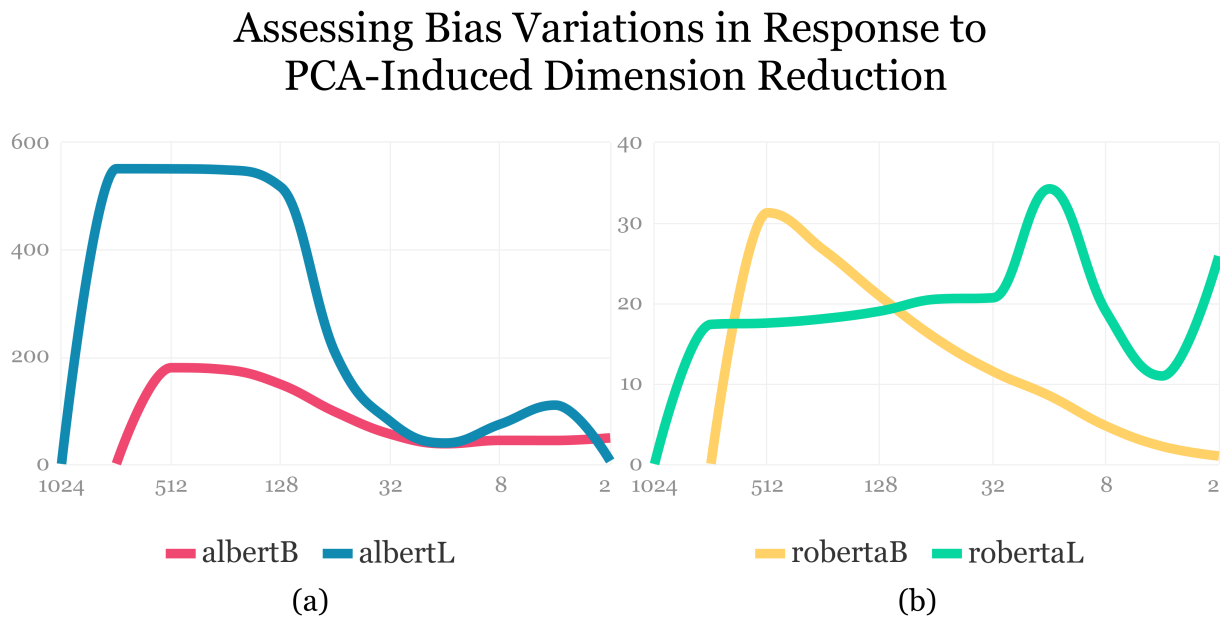

Figure 7: Analysis of Normalized Bias Trends in Language Models Following PCA-Based Dimensionality Reduction. This figure is divided into three panels, each illustrating the relationship between the number of dimensions retained after PCA and the bias evaluated in different language models. Panel **(a)** focuses on the albert-base-v2 and albert-large-v2 models, while Panel **(b)** examines the roberta-base and roberta-large models. This segmentation into three distinct graphs is intentional, as it addresses the varying bias ranges across these models, ensuring clarity and preventing the loss of crucial details. The graphs reveal a characteristic pattern in which the normalized evaluated bias initially increases significantly with reduced dimensions, followed by a subsequent decrease. This observation suggests that the PCA process impacts the evaluation of bias, possibly due to the loss of information inherent in dimensionality reduction techniques. This phenomenon underscores the complexities involved in interpreting bias within language models and highlights the need to carefully consider the impacts of dimensionality reduction in bias analysis.

	Norm. Bias Value for Female			Norm. Bias Value for Male			Normalized Bias		
Language Model	WinoBias	Winogender	EEC	WinoBias	Winogende	EEC	WinoBias	Winogender	EEC
albertB	1.3990	1.5623	1.2361	1.4403	1.6061	1.2535	0.7245	0.8076	0.6286
albertL	1.1235	1.2125	1.0977	1.1028	1.2341	1.1064	0.5639	0.6193	0.5541
albertXXL	**1.7339**	1.6327	1.2728	**1.8018**	1.6927	1.2639	**0.9079**	0.8526	0.6373
bertB	0.5652	0.5829	0.4652	0.5579	0.5766	0.4841	0.2833	0.2921	0.2440
bertL	0.7246	**2.1305**	**2.4555**	0.6705	**2.1306**	**2.5476**	0.3679	**1.0653**	**1.2833**
distilbert	0.7067	0.3850	0.3430	0.6595	0.3820	0.3578	0.3582	0.1928	0.1805
robertaB	0.1196	0.1159	0.1079	0.1207	0.1161	0.1094	0.0605	0.0581	0.0549
robertaL	0.0325	0.0300	0.0167	0.0320	0.0301	0.0165	0.0163	0.0151	0.0083
xlmB	0.0123	0.0132	0.0137	0.0121	0.0128	0.0144	0.0062	0.0066	0.0073
xlmL	0.0220	0.0221	0.0190	0.0213	0.0217	0.0217	0.0111	0.0111	0.0111

Table 5: Consolidated normalized spaces and biases for gender terms across different language models. WinoBias, Winogender, and EEC represent values from the respective datasets. A lower bias value indicates a lower bias. Notably, bert-large-cased stands out with the highest bias value, whereas xlm-roberta-base exhibits the lowest bias value among the models.

	Bias Value for Female			Bias Value for Male			Non-normalized Bias		
Language Model	WinoBias	Winogender	EEC	WinoBias	Winogender	EEC	WinoBias	Winogender	EEC
albertB	0.5362	0.5590	0.4843	0.5521	0.5747	0.4912	0.2777	0.2889	0.2463
albertL	0.4830	0.4916	0.4638	0.4741	0.5003	0.4675	0.2424	0.2511	0.2341
albertXXL	**0.6094**	0.5938	0.5321	**0.6332**	0.6156	0.5284	**0.3191**	0.3101	0.2664
bertB	0.3451	0.3512	0.2914	0.3407	0.3475	0.3032	0.1730	0.1760	0.1528
bertL	0.4290	**0.6212**	**0.6847**	0.3969	**0.6212**	**0.7104**	0.2178	**0.3106**	**0.3579**
distilbert	0.4019	0.2640	0.2311	0.3751	0.2620	0.2411	0.2037	0.1322	0.1216
robertaB	0.1052	0.1027	0.0959	0.1061	0.1030	0.0972	0.0532	0.0515	0.0488
robertaL	0.0316	0.0292	0.0164	0.0311	0.0293	0.0162	0.0158	0.0147	0.0082
xlmB	0.0120	0.0128	0.0129	0.0117	0.0124	0.0136	0.0060	0.0064	0.0069
xlmL	0.0216	0.0216	0.0187	0.0209	0.0212	0.0213	0.0109	0.0108	0.0109

Table 6: Consolidated non-normalized spaces and biases for gender terms across different language models. WinoBias, Winogender, and EEC represent values from the respective datasets. A lower bias value indicates a lower bias. Notably, bert-large-cased and albert-xxlarge-v2 stand out with the highest bias value, whereas xlm-roberta-base exhibits the lowest bias value among the models.

	Rank for Bias Value			Rank for Norm. Bias Value		
Language Model	WinoBias	Winogender	EEC	WinoBias	Winogender	EEC
albertB	9	8	8	9	8	8
albertL	8	7	7	8	7	7
albertXXL	**10**	9	9	**10**	9	9
bertB	5	6	6	5	6	6
bertL	7	**10**	**10**	7	**10**	**10**
distilbert	6	5	5	6	5	5
robertaB	4	4	4	4	4	4
robertaL	3	3	2	3	3	2
xlmB	1	1	1	1	1	1
xlmL	2	2	3	2	2	3

Table 7: Rankings for Bias (Non-normalized) and Bias Norm (Normalized) across different language models for the datasets WinoBias, Winogender, and EEC. A lower rank indicates a lower bias. The rank values for normalized bias are consistent with the rank values for non-normalized bias. In terms of bias assessment, the xlm-alberta-base model demonstrates the lowest bias values in both normalized and non-normalized bias evaluations. Conversely, the bert-base-large and albert-xxlarge-v2 stand out with the highest value in both bias evaluations.

	Non-normalized Bias for Gender Groups			
Language Model	Female	Male	Neutral Gender	Non-normalized Bias
xlmB	0.0125	0.0123	0.0120	0.0062
xlmL	0.0213	0.0210	0.0204	0.0107
robertaL	0.0284	0.0286	0.0269	0.0145
robertaB	0.1013	0.0999	0.0997	0.0506
distilbert	0.2575	0.2588	0.2670	0.1331
bertB	0.3400	0.3408	0.3556	0.1771
albertL	0.4548	0.4906	0.4875	0.2487
albertB	0.5382	0.5476	0.5163	0.2751
albertXXL	0.5784	0.5949	0.5616	0.2975
bertL	**0.6003**	**0.3033**	**0.6020**	0.6031

Table 8: Non-binary Bias (non-normalized) Comparison Results for Different Language Models with WinoBias Dataset. The lower the ranking, the lower bias. Overall, the model xlm-roberta-base has the lowest bias value, and bert-large-cased has the highest bias value.

	Bias for Gender Groups			
Language Model	Female	Male	Neutral Gender	Bias
xlmB	0.0128	0.0127	0.0124	0.0064
xlmL	0.0217	0.0214	0.0208	0.0109
robertaL	0.0292	0.0294	0.0277	0.0148
robertaB	0.1137	0.1121	0.1119	0.0568
distilbert	0.3739	0.3759	0.3877	0.1933
bertB	0.5616	0.5629	0.5874	0.2926
albertL	1.0924	1.1784	1.1709	0.5974
albertB	1.4437	1.4690	1.3849	0.7378
albertXXL	1.5362	1.5801	1.4916	0.7901
bertL	**2.0009**	**2.0231**	**2.0067**	1.0109

Table 9: Non-binary Bias (normalized) Comparison Results for Different Language Models with WinoBias Dataset. The lower the ranking, the lower bias. The normalized bias values are calculated based on Eq. 7. Overall, the model xlm-roberta-base has the lowest bias value, and bert-large-cased has the highest bias value.

	Models						
Dimension	bertB	bertL	distilbert	albertB	albertL	robertaB	robertaL
1024	-	0.6975	-	-	1.1131	-	0.0323
768	0.5615	**8.9859**	0.6831	1.4197	**1079.6957**	0.1202	34.6198
512	**8.0015**	8.6470	**19.8404**	**357.3488**	1079.2734	**62.3124**	34.9535
256	7.3255	7.9287	18.7775	349.7006	1075.0957	53.0488	36.0275
128	6.4250	7.0844	17.1960	295.4346	1011.9271	41.5281	37.9229
64	5.4317	6.1386	15.1571	192.6671	400.9505	30.9722	40.8956
32	4.5103	5.2444	12.5751	111.0657	155.1969	22.8938	41.2816
16	3.6683	4.1911	9.8005	75.9347	77.6041	16.8120	**68.2724**
8	2.1618	2.3379	3.8296	88.6019	145.1582	9.2294	37.3822
4	1.0187	1.2037	1.4740	88.3973	209.8812	4.3610	21.6785
2	0.5483	0.7579	0.8311	95.6846	10.1703	2.0217	50.7239

Table 10: The details of the bias values for different language models with different dimensions after PCA. The lower the value, the lower the bias. After the first PCA dimension reduction, the evaluated bias values have a significant increase. Specifically, the bias values for albert-base-v2 and albert-large-v2 increase significantly. After that, the bias values start to decrease.