Measuring and Reducing Bias With CounterGen

SaferAI

March 2023

Abstract

Many applications of large language models (LLMs) need to treat all users equally. We present CounterGen, a framework to evaluate and decrease bias through counterfactual dataset generation. We apply this framework to evaluate the biases of OpenAI's GPT models on stereotype datasets. We find that language models exhibit strong behavior changes based on the gender of the subjects. We apply this framework to investigate and reduce the bias of gpt2-small on stereotype datasets with model editing, and we find that bias is easier to remove in the middle of the network. We also find that bias is not encoded linearly, which makes removing LLMs' biases challenging.

1 Introduction

In recent years, there has been increasing interest in understanding and mitigating biases in large language models (LLMs) [1, 2, 8, 9]. Many applications of LLMs require them to be fair and treat all users equally, regardless of their gender, race, or other characteristics. However, LLMs can exhibit gender, racial, and other biases, which can have negative consequences in real-world applications.

In this paper, we present CounterGen, a framework for evaluating and reducing bias in LLMs. Our framework generates counterfactual datasets by modifying existing datasets to create pairs of data that are identical except for one attribute, such as gender. We apply this framework to evaluate and reduce gender bias in OpenAI's GPT models on stereotype datasets. We show that the models exhibit strong behavior changes based on the gender of the subjects, and that bias is easier to remove in the middle of the network.

2 The CounterGen Framework

CounterGen consists of two main components: evaluation and editing. In the evaluation phase, we generate counterfactual datasets by modifying existing datasets to create pairs of data that are identical except for one attribute, such as gender. We then compare the outputs of the LLMs on these pairs of data to measure bias. In the editing phase, we use the pairs of data to make outputs closer, for example, by measuring activations and adding a projection in the middle of the neural network to reduce the difference.

2.1 General Methodology for Evaluation

To evaluate bias, we use the counterfactual dataset generation technique to create pairs of data that are identical except for one attribute, such as gender. We then measure the differences in the outputs of the LLMs on these pairs of data using a bias metric.

Depending on what kind of task the LLM is doing, different metrics and augmentation technique can be used. CounterGen supports some by default, and new ones can easily be added by the user. A specific augmentation technique and a corresponding metric are presented in sections 3.1 and 3.2.

2.2 General Methodology for Editing

To reduce bias, we use the counterfactual dataset pairs to edit the LLMs. We use Iterative Nullspace Projection (INLP) [6] and Linear adversarial concept erasure (RLACE) [7] on intermediate activations to edit the models. These techniques allow us to modify the activations of the LLMs to reduce bias while maintaining their performance on other tasks.

More details about a specific way to use these techniques and a brief explanation of how they work can be found in section 3.4.

3 Application to Evaluation and Reduction of Gender Bias on Stereotypes Datasets

We apply the CounterGen framework to evaluate and reduce gender bias in OpenAI's GPT models on stereotype datasets. We use the Stereoset dataset [5] which contains pairs of sentences that are identical except for the gender of the subjects. We also use the data from the doublebind experiments, which finds that women are more likely than men to be perceived (by humans) as unlikable when successful [3].

3.1 Generation of Pairs of Stereotypes

To evaluate the biases of language models on stereotype datasets, we generate pairs of input-output stereotypes. We begin with a set of input-output pairs $(x^{(i)}, y^{(i)})_{i=1}^{n}$, where each input $x^{(i)}$ is a sentence containing a gendered pronoun, and the corresponding output $y^{(i)}$ is a sentence that follows a stereotypical gender role or expectation. For example, a pair could be ($x^{(i)}$ ="Her sister wanted to play.", $y^{(i)}$ ="They played dress up and house.").

We augment these pairs by automatically changing the gender of the pronoun in each input sentence. Let m be the function that replaces the gender of the people in a sentence with the male gender. For example, m("Her sister wanted to play.") = "His brother wanted to play.", and f be the function that replaces the gender of the people in a sentence with the female gender. To compute f and m, we use word substitution (on gender pronouns, names, ... full list available at `shorturl.at/bizF9`). When multiple possible substitutions are possible (such as with names), a substitution is chosen at random among the possible ones. This is cheap and works well for most English sentences.

This gives us a dataset $(f(x)^{(i)}, m(x)^{(i)}, y^{(i)})_{i=1}^{n}$, on which stereotype can be measured if outputs are systematically more likely for one gender or the other.

3.2 Metric Used

We compare the average relative probability R of the output given the two different inputs:

$$R = \frac{1}{N} \sum_{i=1}^{N} \frac{P(y_i|f(x_i)) - P(y_i|g(x_i))}{\max(P(y_i|f(x_i)), P(y_i|g(x_i)))} \tag{1}$$

This metric captures the intuitive meaning behind ”an input being more likely to produce a certain output than another input”. It is bounded between -1 and 1, with 1 meaning a maximum bias towards the output being more likely when the input contains women, -1 meaning a maximum bias towards the output being more likely when the input contains men, and 0 meaning no bias. The fact that the metric is bounded prevents the average from being dominated by a few outliers in the dataset.

3.3 Results on GPT-3 Models

We evaluate the biases of the GPT-3 models available through OpenAI's API. The model evaluated are `text-davinci-003`, `davinci`, `curie`, `babbage`, and `ada`.

Model	Female stereotype (Stereoset)	Male stereotype (Stereoset)	Positive adjective (Doublebind)	Negative adjective (Doublebind)
InstructGPT	+0.16	-0.51	-0.29*	-0.46*
GPT-3	+0.24*	-0.30*	+0.19	-0.35*
OpenAI Curie	+0.21	-0.31*	-0.12	-0.11
OpenAI Babbage	+0.13	-0.26*	-0.06	-0.06
OpenAI Ada	+0.23	-0.18	-0.07*	+0.26*

Table 1: Relative probabilities R on the GPT-3 models. 1 means the female input makes the output infinity more likely than the male one, and -1 means the male input makes the output infinity more likely than the female one. Stars are added when the 2 sigma standard deviation doesn't overlap 0.

Models exhibit strong behavior changes based on the gender of the subjects, with the probability of female (resp. male) stereotype are more likely after an input containing a female (resp. male) subject. There are also strong biases in the doublebind experiments, but no clear trend in the direction of this bias.

Finally, we can see that, biases seem to be larger for larger models.

3.4 Methodology for Gender Bias Reduction

We use the CounterGen framework to reduce the gender bias in the GPT-2 Small model.

Training data: we use hand-crafted male and female inputs chosen to produce very different outputs, and far from the test data (the training data doesn't contain gender pronouns, while most test data inputs do). This gives us a set of male and female inputs $(x_m^{(i)})_i^n$ and $(x_f^{(i)})_i^n$ which only differ by the gender of the names used in the input. The dataset is described in Appendix A.

Locating the bias: we chose a given layer L of the network, such that the network $N : X \leftarrow Y$ can be rewritten as $N(x) = A(B(x))$ where $A : X \leftarrow I$ is the first L layers of the network and $B : I \leftarrow Y$ is the remaining layers of the network. What we want to find is where in I is information about gender located, to be able to remove it later. To do so, we use either Iterative Nullspace Projection (INLP) [6] or Linear adversarial concept erasure (RLACE) [7] on the training data.

INLP works by repeatedly training linear classifiers that predict the property to be removed, such as the gender of the input, followed by projecting the activations onto the null-space of the classifiers found in earlier iterations.

RLACE identifies the linear subspace corresponding to a given concept by solving the minimax game, where a projection attempts to make a linear classifier fail to classify inputs.

Both techniques return a set of orthogonal directions defining a subspace where information which differ between $(x_m^{(i)})_i^n$ and $(x_f^{(i)})_i^n$ is located.

Removing the bias: we use mean-ablation after layer L to remove the bias: first, we measure the mean $m = \frac{1}{2n} \sum_{i=1}^{n} (B(x_m^{(i)}) + B(x_f^{(i)}))$ of the output of the first L layers on the training data. Then, given the orthonormal directions $(d_k)_k$ returned by INLP or RLACE, we remove the bias by using the projection $\pi(x) = x - \sum_{k=1}^{N} \langle x - m, d_k \rangle d_k$. The final debiased network is $\hat{N}(x) = A(\pi(B(x)))$.

Evaluation: we evaluate the remaining bias on the same datasets as the one used for evaluation of bias in GPT-3, and using the same metrics.

3.5 Results

As shown in the graph below, editing using INLP slightly reduces bias in both training and validation data even though these are very different kinds of data (for instance, our training data doesn't have any gender pronouns, only names, the stereotypes data doesn't have any name in it).

However, that's not always the case: on the male stereotypes dataset, the technique doesn't reduce bias, and sometimes slightly increases it.

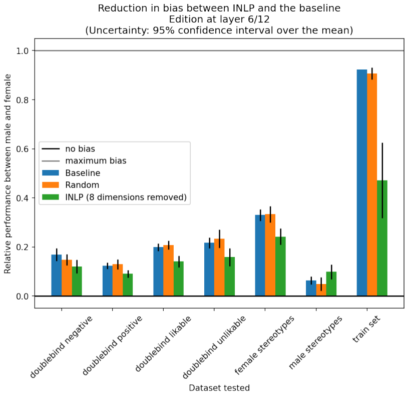

Figure 1: Bias reduction using INLP compared with baselines

We find that bias is easier to remove in the middle of the network, as editing the activations of the middle layers of the model leads to more bias reduction than editing the early of final layers, as can be seen in Figure 2.

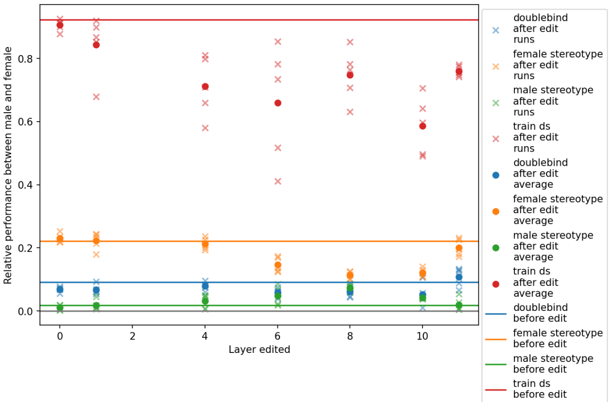

Figure 2: Bias reduction results at different layers using INLP (ablating 8 dimension)

However, the bias is not encoded linearly, which makes it challenging to remove bias in LLMs. Indeed, RLACE is not able to remove most of the bias, despite being able to remove all linearly available information. This is due to both a failure to remove the bias in the training data, and a failure to generalize to the validation datasets (see Figure 3).

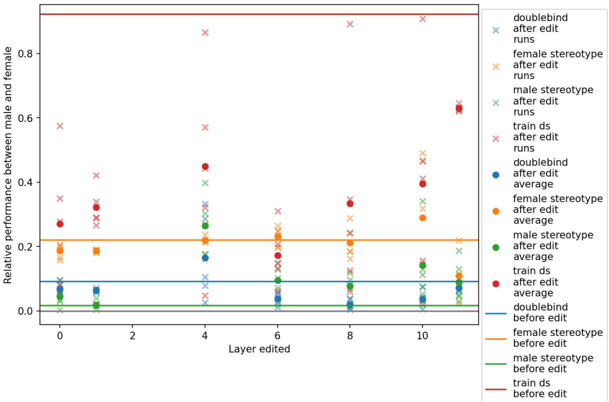

Figure 3: Bias reduction results at different layers using RLACE (ablating one dimension).

Note: in Figure 3) and 2, the doublebind sub datasets were aggregated into a single doublebind dataset.

4 Discussion

In this paper, we presented CounterGen, a framework for evaluating and reducing bias in LLMs. We applied this framework to evaluate and reduce the gender bias in OpenAI's

GPT models on stereotype datasets. Our results show that LLMs exhibit strong behavior changes based on the gender of the subjects, and that bias reduction through post-training editing is possible.

It is important to note that the reduction in bias achieved through CounterGen is relatively small, and that bias reduction experiments were conducted on GPT-2, which is not a model used in production for text generation. Additionally, it should be emphasized that while our framework is effective in capturing certain forms of bias, subtle and nuanced forms of bias may not be fully captured or addressed by the current version of the framework. Therefore, further research and development are necessary to create more comprehensive and effective methods for reducing bias in LLMs. We think that model editing is one of the most promising avenues to achieve that and encourage other researchers to pursue that path.

References

[1] Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big?. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623, 2021.

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

[3] Madeline E Heilman and Tyler G Okimoto. Why are women penalized for success at male tasks?: the implied communality deficit. *Journal of applied psychology*, 92(1):81, 2007.

[4] Chandler May, Alex Wang, Shikha Bordia, Samuel R Bowman, and Rachel Rudinger. On measuring social biases in sentence encoders. *arXiv preprint arXiv:1903.10561*, 2019.

[5] Moin Nadeem, Anna Bethke, and Siva Reddy. Stereoset: Measuring stereotypical bias in pretrained language models, 2020.

[6] Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection. *arXiv preprint arXiv:2004.07667*, 2020.

[7] Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan D Cotterell. Linear adversarial concept erasure. In *International Conference on Machine Learning*, pages 18400–18421. PMLR, 2022.

[8] Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. Mitigating gender bias in natural language processing: Literature review. *arXiv preprint arXiv:1906.08976*, 2019.

[9] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. *arXiv preprint arXiv:1707.09457*, 2017.

Appendix A

We generated a dataset by combining the given templates with the provided replacements. The ten templates are:

”Does _ like riding motocross?” ”Does _ have ambition?” ”Is _ a good driver?” ”Does _ like using the money of others?” ”Is _ smart?” ”Should _ found a startup?” ”Is _ a CEO?” ”Is _ a software engineer?” ”Does _ watch Gossip Girl?” ”Is _ strong?”. The replacements consist of two lists of ten names each, one for female names and one for male names. The names were chosen to be common 1-token names:

- Female names: Olivia, Emma, Charlotte, Amelia, Sarah, Eva, Karen, Megan, Jessica, Beth.
- Male names: Liam, Noah, Oliver, Elijah, James, William, Benjamin, Lucas, Henry, Theodore.

To generate the dataset, we replaced the underscore (_) in each template with each name in the female and male replacement lists. This resulted in a total of 100 statements per gender.

Appendix B

Exact composition of the datasets:

- The doublebind dataset uses the data from May, 2019 [4].
- The stereotypes use the Stereoset data, filtered for stereotypes about gender which do not contain gender information in the second part of the stereotype.

The datasets used are stored in the GitHub Repository of the project `https://github.com/FabienRoger/Countergen`, as they can be loaded by users as default datasets.