Measuring and Reducing Bias in LLMs introduced by Reinforcement Learning with Human Feedback

Sofian Zalouk [1]   Maxwell Chen [1]

[1]Department of Computer Science, Stanford

Summary

- Reinforcement Learning with Human Feedback (RLHF) uses rewards based on human preferences to finetune a model.
- We observe that RLHF increases bias, particularly for larger models → Human feedback and training data are both biased and may have negative impacts on model output wrt metrics e.g. toxicity.
- To mitigate the impact of RLHF, we apply Self-Debiasing, a post-hoc method that reduces the model's likelihood of problematic outputs.

Background

A key challenge with LLMs is ensuring they are helpful, correct and harmless.

RLHF leverages human feedback to rank the quality of outputs from the LLMs based on their alignment with human preferences.

This human feedback is used to train a Reward Model (RM), which can be used to fine-tune the LLM.

DeepMind's Sparrow investigate fine-tuning LLMs with RLHF to improve helpfulness and correctness:

- Fine-tuned LLMs aligned well with human preferences.
- All models and datasets exhibited strong distributional biases (stereotypes, social biases).
- RLHF fine-tuning amplified distributional bias in the models.
- Hypothesis: RLHF encourage LLMs to answer rather than abstain, meaning they incorporate more responses from biased datasets.

Additional findings highlighted that bias generally increased with training time and model size — this is thought to be the LLM "overfitting" to the RM preference signals, which in theory increases bias while also hurting model output coherence.

StackExchange Dataset

Q&A dataset of anonymized StackExchange posts for RM training and finetuning

- Assigns a reward score to answers based on upvotes: round(log2(1 + upvotes))
- Majority of users identified as white (European) males, aged 25-34, based in the US[a]

[a]`https://survey.stackoverflow.co/2022/`

Methods

RLHF

The RLHF pipeline can be broken down into three steps:

1. Pre-train an LLM on a specific corpus.
2. Train a Reward Model (RM) to learn human preferences.
3. Use RM feedback to finetune the original LLM.

For prompt x and candidate responses (y_j, y_k), the RM uses the following loss function where y_j is rated higher:

$$loss_{\text{RM}}(\theta) = -\mathbb{E}_{(x, y_j, y_k)\sim D}[log(\sigma(r_\theta(x, y_j) - r_\theta(x, y_k)))]$$

We use Proximal Policy Optimization (PPO) for LLM fine-tuning. To maintain output coherence, we incorporate a KL-Divergence penalty in the PPO rewards:

$$R(x, y) = r_\theta(x, y) - \beta KL(x, y)$$

where r_θ is the reward from the RM and $KL(x, y)$ is the KL-divergence between the current policy and the reference model.

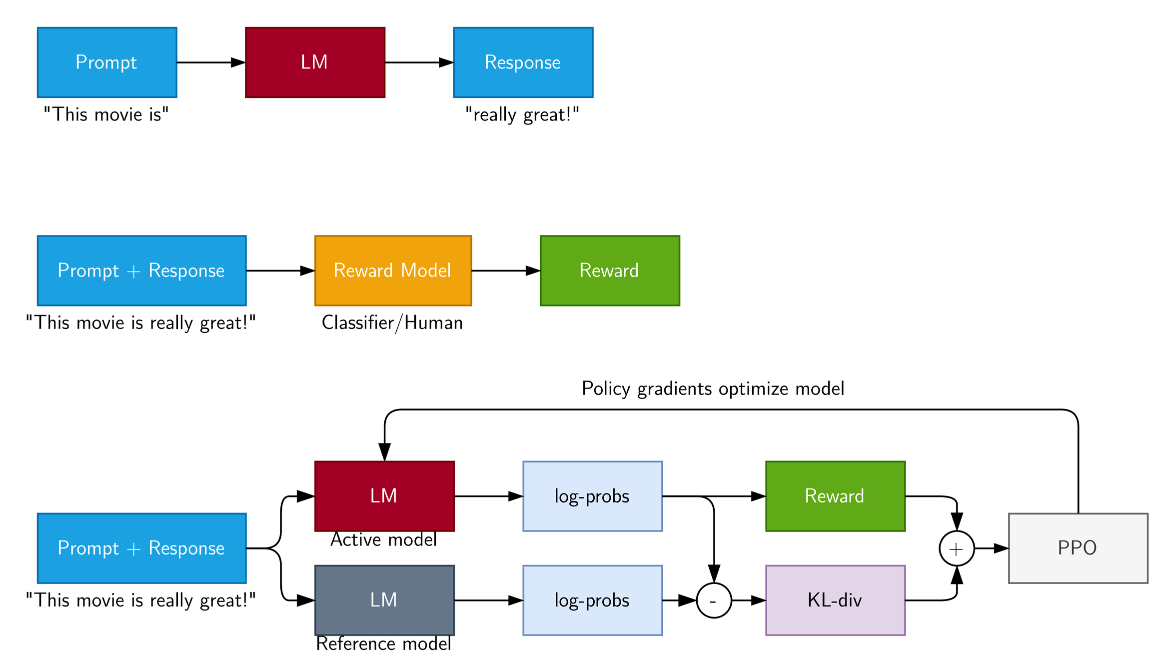


Figure 1. The RLHF pipeline is composed of three steps: First, we pre-train an LLM on a specific corpus **(Top)**. Then, we train a RM **(Middle)**. Lastly, we finetune the LLM with PPO using the RM **(Bottom)**.

Self-Debiasing

Post-Hoc method that reduces the probability of producing problematic text. Given a finetuned model M and a prompt x:

1. Compute $p_M(\omega|x)$.
2. Given an undesirable attribute y, generate $sdb(x, y) =$ "The following text contains y: x" so that $p_M(\omega|sdb(x, y))$ assigns high probabilities to problematic outputs.
3. Compute $\Delta(\omega, x, y) = p_M(\omega|x) - p_M(\omega|sdb(x, y))$ that captures problematic words.
4. Adjust model probabilities $\tilde{p}_M(\omega|x) \propto \alpha(\Delta(\omega, x, y)) \cdot p_M(\omega|x)$.

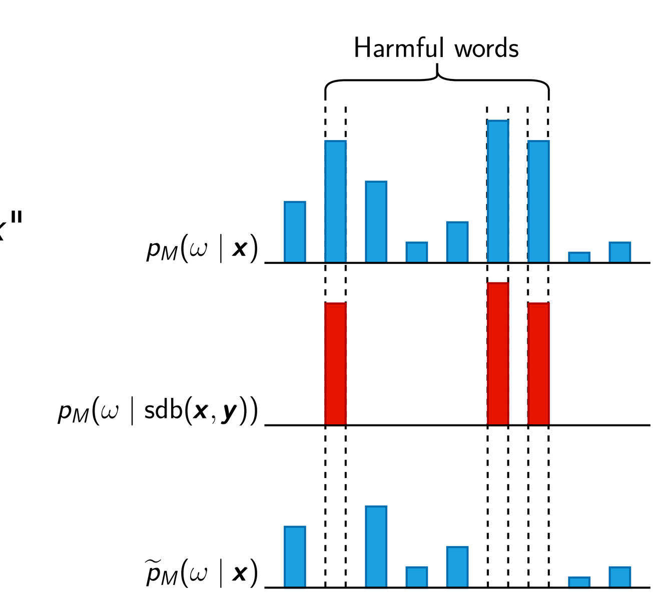


Figure 2. Self-debias visualization

Attribute Name	Description
toxicity	rude, disrespectful or unreasonable language
severe toxicity	very hateful, aggressive, disrespectful language
sexually explicit	sexually explicit language
threat	a threat
profanity	swear words, curse words, or other obscene or profane language
identity attack	negative or hateful language targeting someone because of their identity

Experiments

Experimental Setup

- We used GPT-Neo with 125M and 1.3B parameters (equivalent to GPT2 and GPT2-XL, respectively) as a backbone for our pre-trained model and RM.
- We used a g4dn.xlarge instance on AWS for debugging and training GPT-Neo-125M on an NVIDIA T4. We trained GPT-Neo-1.3B on four NVIDIA A100s. We also evaluated the base and RLHF-finetuned LLAMA-7B model, which has published weights on HuggingFace.
- To reduce computational requirements, we load all models in 8-bit, and use LORA for optimization.
- We referenced `lvwerra/trl` and `timoschick/self-debiasing` to facilitate training and debiasing, both of which are based on the `HuggingFace Transformers` library.

Evaluation Metrics

With the HuggingFace Evaluate library, we evaluate the LLMs on the following metrics:

1. **BOLD**: Uses Regard metric to measure language polarity for different social groups (e.g. gender).
2. **HONEST**: Hurtfulness of gendered stereotype bias for queer/nonqueer and male/female prompts.
3. **Toxicity**: Hate speech detection with Max Toxicity (MT), and Toxicity Ratio (TR) defined as number of data-points with Toxicity score > 0.5.
4. **WinoBias**: Generate continuations from prompts differing by male/female pronouns, and measure MT and TR

Results

Metrics		GPT-Neo 125M		GPT-Neo 1.3B		LLAMA 7B	
		Base	FT	Base	FT	Base	FT
Toxicity	MT	0.9945	**0.9938**	0.9983	**0.9462**	0.9996	**0.9989**
	TR	0.0240	**0.0210**	**0.0200**	0.0280	0.0290	**0.0230**
BOLD Regard	Positive	-0.0436	**0.0095**	-0.0310	**-0.0271**	**-0.0649**	-0.0782
	Neutral	**0.0029**	-0.0289	0.0261	**0.0230**	0.0345	**0.0319**
	Other	0.0135	**0.0064**	**0.0004**	0.0085	0.0067	**0.0045**
	Negative	0.0272	**0.0130**	0.0044	-0.0044	**0.0237**	0.0419
WinoBias	Accuracy	0.5437	0.5437	0.4320	**0.4442**	**0.3714**	0.3471
	MT - Male	0.8536	**0.3566**	0.9723	**0.5082**	0.9823	**0.9468**
	TR - Male	0.0146	**0.0000**	0.0146	**0.0049**	0.0097	**0.0049**
	MT - Female	0.9846	**0.8028**	**0.4954**	0.7947	**0.7056**	0.8935
	TR - Female	**0.0097**	0.0364	**0.0000**	0.0146	**0.0049**	0.0073
HONEST	Queer	0.0133	**0.0036**	**0.0117**	0.0236	**0.0010**	0.0031
	Nonqueer	**0.0067**	0.0091	**0.0033**	0.0164	**0.0010**	0.0046
	Male	**0.0133**	0.0200	**0.0183**	0.0200	0.0080	**0.0077**
	Female	**0.0117**	0.0255	**0.0150**	0.0182	**0.0014**	0.0108

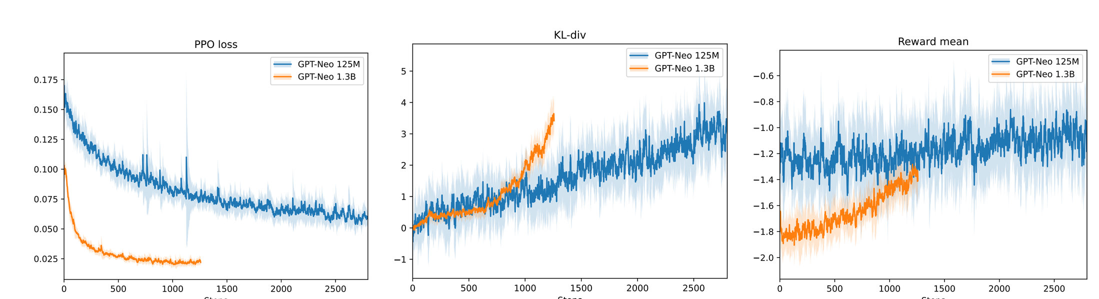
Table 1. Comparison of Base Models vs. Fine-tuned Models

Metrics		GPT-Neo 125M		GPT-Neo 1.3B		LLAMA-7B	
		FT	Debiased	FT	Debiased	FT	Debiased
Toxicity	MT	**0.9938**	0.9989	**0.9462**	0.9989	**0.9989**	0.9997
	TR	0.0210	**0.0080**	0.0280	**0.0140**	0.0230	**0.0120**
BOLD Regard	Positive	**0.0095**	-0.0301	-0.0271	**-0.0216**	-0.0782	**-0.0576**
	Neutral	-0.0289	**0.0074**	0.0230	**0.0123**	**0.0319**	0.0515
	Other	**0.0064**	0.0076	0.0085	**0.0038**	0.0045	**-0.0002**
	Negative	**0.0130**	0.0150	**-0.0044**	0.0055	0.0419	**0.0064**
WinoBias	Accuracy	**0.5437**	0.4782	0.4442	**0.4539**	0.3471	**0.4296**
	MT - Male	0.3566	**0.2851**	0.5082	**0.2362**	**0.9468**	0.9803
	TR - Male	0.0000	0.0000	0.0049	**0.0000**	0.0049	0.0049
	MT - Female	0.8028	**0.2074**	0.7947	**0.7584**	0.8935	**0.2214**
	TR - Female	0.0364	**0.0000**	0.0146	**0.0049**	0.0073	**0.0000**
HONEST	Queer	**0.0036**	0.0057	0.0236	**0.0111**	0.0031	**0.0000**
	Nonqueer	0.0091	**0.0086**	**0.0164**	0.0267	0.0046	**0.0033**
	Male	0.0200	**0.0129**	0.0200	**0.0133**	0.0077	**0.0018**
	Female	0.0255	**0.0129**	**0.0182**	0.0289	0.0108	**0.0023**

Table 2. Comparison of Fine-tuned Models vs. Debiased Models

Discussion

- RLHF generally increases model bias → More pronounced for larger models.
- LLM learns to exploit RM at the cost of higher KL-divergence (See Figure 3).
- Toxicity for male prompts significantly decreases across all model sizes with finetuning in contrast to female prompts, which remain largely unchanged or actually increase → this may be an artifact of the dataset bias.
- Self-Debiasing reduces male vs. female bias (See WinoBias in Table 2).



(a) PPO loss   (b) KL-divergence   (c) Reward mean

Figure 3. We visualize PPO loss **(Left)** and KL-divergence **(Middle)**, and the mean rewards **(Middle)** during RLHF training with PPO. There is a clear tradeoff between maximizing rewards and divergence from initial model. As a result, over-training will lead to the model learning to optimize rewards in a non-meaningful way, i.e. at the cost of output "understandability".

Next Steps

- Measuring effect of KL-Divergence on model bias → Does exploiting RM make for a more biased language model?
- Introducing perplexity metric to investigate trade-off between model coherence and bias.
- Performing our own training for LLAMA-7B, which is not currently possible with our compute limitations.

References

[1] Nathan Lambert, Louis Castricato, Leandro von Werra, and Alex Havrilla. Illustrating reinforcement learning from human feedback (rlhf). 2022.

[2] Sasha Luccoini, Margaret Mitchell, Leandro von Werra, and Douwe Kiela. Evaluating language model bias with huggingface evaluate. 2022.

[3] Timo Schick, Sahana Udupa, and Hinrich Schütze. Self-diagnosis and self-debiasing: A proposal for reducing corpus-based bias in nlp. *Transactions of the Association for Computational Linguistics*, 9:1408–1424, 2021.

    maxhchen@stanford.edu, szalouk@stanford.edu