Quantifying Bias from Decoding Techniques in Natural Language Generation

Mayukh Das and **Wolf-Tilo Balke**
Institute for Information Systems, TU Braunschweig
Mühlenpfordtstraße 23, 38106 Braunschweig, Germany
{mayukh,balke}@ifis.cs.tu-bs.de

Abstract

Natural language generation (NLG) models, when conditioned with text containing demographic information, can propagate social bias towards particular demography. Each component in a pipeline of an NLP task, like data, modeling, decoding, evaluation, can uniquely contribute to transmitting bias. Though several studies investigated bias from data and model, NLG task distinctively uses search, random sampling, entropy during inference time to change the distribution of the model's predicted tokens at each autoregressive time-step. This stochastic inference can positively or negatively impact the bias-sensitive tokens initially predicted by the model. To address this gap in research, we present an extensive analysis of bias from decoding techniques for open-domain language generation considering the entire decoding space. We analyze to what extent absolute bias metrics like toxicity and sentiment are impacted by the individual components of decoder algorithms. To this end, we also analyze the trade-off between absolute bias scores and human-annotated generation quality for multiple points in the decoder spectrum with several decoding setups. Together, these methods reveal the imperative of testing inference time bias and provide evidence on the usefulness of inspecting the entire decoding spectrum.

1 Introduction

Natural language generation (NLG) techniques provide the backbone for many downstream artificial intelligence applications, such as chat-bots, virtual assistance, machine translation, automatic storytelling, text summarization, and writing assistants. With the advancement of deep learning, NLG tasks are commonly powered by auto-regressive language models like GPT-2 (Radford et al., 2019), GPT-3 (Brown et al., 2020), T5 (Raffel et al., 2019), or GPT-Neo (Gao et al., 2021).

However, language models (LMs) pretrained on large web text corpora are also known to pass on stereotypical associations learned from real-world training data. Such disproportionate generations or skews that produce representational or allocational harms towards a particular group is called "bias" in the context of AI fairness (Crawford, 2017; Barocas and Selbst, 2016). Although a moderate amount of studies has been conducted on quantifying bias for natural language understanding (NLU) such as natural language inference, co-reference resolution, masked language model predictions (Webster et al., 2018; Lu et al., 2018; Cao and Daumé III, 2020; Dev et al., 2019; Nangia et al., 2020; Nadeem et al., 2021; Zhao et al., 2018, 2020), exploring the same for NLG is a nascent, yet active area of research.

Indeed, bias can be introduced at various phases of the model's development and deployment pipeline, such as data, modeling, decoding, evaluation. Much of the work on analyzing bias in NLG focuses on benchmarking biases pertaining to models or training data (Henderson et al., 2018; Sheng et al., 2019, 2020; Habash et al., 2019; Bordia and Bowman, 2019; Cercas Curry et al., 2020; Liu et al., 2020; Yeo and Chen, 2020; Dhamala et al., 2021). Yet, up to now work on examining biases from *decoder techniques* is relatively scarce. However, generation tasks distinctively use search or random sampling techniques during inference time. Moreover, entropy (softmax penalty) is also used along with sampling to change the distribution of model predicted tokens at each autoregressive time-step. Redistributing the predicted token and modulating randomness during inference can positively or negatively impact the bias-sensitive tokens[1] initially predicted by the model.

Our work is focused on addressing this gap in the literature for auto-complete[2] generation. Re-

[1]Words with negative connotations towards specific demographics as explained by Liang et al. (2021)

[2]Continuous conditional generation directly from LMs

lated works test bias in LMs for a single point in the decoder spectrum, which does not quantify the effect of the decoder in propagating bias. In contrast we investigate the bias variation induced by the decoding algorithms for the full spectrum of decoder space[3]. We perform tests for six state-of-the-art LMs, with diverse decoding setup and bias objectives like sentiment and toxicity. To the best of our knowledge, this is the first comprehensive analysis in this regard. We observed entropy and nucleus sampling impacts absolute bias scores across the decoder space while top-*k* and beam search is agnostic. This along with our experimental findings, we demonstrate why inspecting bias for the full decoder spectrum is imperative. To this end, noticing the lack of consensus on which decoding procedure is best from the perspective of bias and quality (restricted to the quality vs. diversity (Zhang et al., 2021; Holtzman et al., 2019)), we also study the trade-off between quality and bias throughout the decoding space using human evaluation. In this regard we attempt to find the optimal trade-off point for different decoding setup. Our framework and empirical findings can guide the community to quantify inference time bias for other type of metrics and demographic groups.

2 Related Work

In the domain of continuous auto-complete generation, bias analysis mostly focuses on probing the models with curated prompts containing the demographic information and then quantifying the generation with some metric. Sheng et al. (2019) and Huang et al. (2020) both used this setup. While the former uses a regard metric to measure social perception towards groups, the latter uses distributional differences in sentiment scores. Shwartz et al. (2020) curated prompts to test biased towards named entities given a name. Groenwold et al. (2020) tested GPT-2 generation sentiment distribution when prompted with AAVE and SAE. Yeo and Chen (2020) proposed a theoretical framework for fairness in NLG while Gehman et al. (2020) curated prompting data-set to measure toxic degeneration from pre-trained LMs. Sheng et al. (2020) also showed that adversarial triggers (Wallace et al., 2019) can be used to further induce bias in pre-trained LMs. Dhamala et al. (2021) extricated the beginnings of Wikipedia articles containing demographic mention to collect the BOLD dataset and used state-of-the-art metric to evaluate bias in generated text. Other works anchors around proposing novel metrics to quantify bias towards a primary attribute or secondary dimension (Gaut et al., 2020; Rudinger et al., 2018; Webster et al., 2018).

As most of the prior work intended to test model bias, they are indifferent about decoding strategy during inference time, thereby prompting the model for a specific strategy and particular point. Closely related to our work was a study done by Sheng et al. (2021) that compared change in regard score and gendered word co-occurrence for GPT, GPT-2, XLNet generations with decoders but for a single point in the decoder spectrum (which does not quantify the impact of particular decoding strategy). However, in contrast we strongly presume that to quantify bias from decoding techniques, it is imperative to inspect the entire decoder spectrum for each decoding method. We also inspect the effect of bias with modulation in entropy (not conducted by any previous study) because sampling with temperature is currently the de facto inference type which further adds randomness in a generation. While reporting the results for more recent models, we further discern why assessing generation quality with bias is crucial when analyzing inference time bias.

3 NLG Decoding

Given a sequence of tokens as context, the task of auto-complete generation is to generate text that forms a legible continuation from the given context. Formally, when prompted with a sequence of m tokens $x_1...x_m$ the model computes $P(x_{1:m+n}) = \prod_{i=1}^{m+n} P(x_i|x_1...x_{i-1})$ to generate the next n completions $x_1...x_{m+n}$ autoregressively using a particular decoding strategy.

One popular decoder is top-*k* sampling (Fan et al., 2018; Radford et al., 2019; Holtzman et al., 2018). Given a distribution $P(x|x_{1:i-1})$, top-*k* vocabulary $V^{(k)} \subset V$ is defined as a set of size k that maximizes $\sum_{x \epsilon V^{(k)}} P(x|x_{1:i-1})$. At each time-step the next token is randomly sampled from top-*k*. Holtzman et al. (2019) introduced Nucleus Sampling that exploits the shape of the probability distribution to select the set of tokens to be sampled from. Formally, Given a distribution $P(x|x_{1:i-1})$, top-*p* vocabulary $V^{(p)} \subset V$ is defined as the smallest set such that $\sum_{x \epsilon V^{(p)}} P(x|x_{1:i-1}) \geq p$. At each time-step random sampling is done from the high-

[3]In this paper, we will be using decoder spectrum and decoder space interchangeably

est probability tokens whose cumulative probability mass exceeds the pre-chosen threshold $p \in [0, 1)$. Typically, temperature-controlled sampling techniques are used where before sampling, temperature $T \in [0, 1)$ is use to control the shape of the distribution (controlling entropy) (Ackley et al., 1985; Fan et al., 2018; Caccia et al., 2018). Formally, before sampling given a temperature $T > 0$ and scores $v_i \in \mathbb{R}^n$ for each token i in the vocabulary V, the probability that the model would predict the i_{th} token is given by (softmax re-estimation):

$$P_i = \frac{e^{v_i/T}}{\sum_j e^{v_j/T}} \quad (1)$$

In the equation above, $T \to 0$ approximates a greedy distribution which magnifies the peaks in the probability distribution while $T \to \infty$ flattens the distribution to make it more uniform. However, $T > 1$ is rarely used.

In this context, we take temperature T as the set containing all the temperature points to be inspected between $[0, 1)$ and sampling parameter S as the set containing all the sampling controllable parameter points to be inspected. We define decoder space $\mathcal{D}_{ST}$ for a sampling technique as:

$$\mathcal{D}_{ST} = S \times T \quad (2)$$

where $S \in [0, 1)$ for **top-*p*** or $S \in [0, V^{(k)})$ for **top-*k*** (for actual values see sec 4.3). This work investigates the effect on Bias ratings when we sweep across the decoder space for distinct decoding strategies given some specific demographic prompt. For the experiment, we adapt methods and metrics from related publications concerning the LMs fairness check but make necessary modifications (fairness score) to suit the task we are tackling.

4 Method and Metrics

We document our evaluation methods as suggested by Dev et al. (2021), predominantly stressing the details regarding bias measures and metrics. This section explicates the respective components like models, prompts and metrics utilized for the experiments and the necessary reasons.

4.1 Models

As the bias testing framework is catered for auto-complete generation tasks, we only include transformer-based LM that is trained with a causal language modeling objective: predicting the next word given a sequence of previous words in an auto-regressive manner. Therefore, we use **GPT-2** (large) trained on BooksCorpus[4]. Two variants of **GPT-Neo** trained on Pile[5]: *GPT-Neo 1.3B*, *GPT-Neo 2.7B* and three versions of **GPT-3** trained on Common Crawl, WebText2: *Babbage*, *Curie* and *Davinci* (Radford et al., 2019; Brown et al., 2020; Gao et al., 2021). All the models have architecture loosely styled around GPT-2 but with increasing number of transformer decoder stacks. The models were chosen with the intent to understand whether model size has any auxiliary effect on the bias ratings while sweeping through $\mathcal{D}_{ST}$.

4.2 Prompts and Metric

Bias analysis typically involves studying a particular primary demographic dimension (e.g., ethnicity) through a secondary dimension (e.g., profession). We condition the language model with prefix template ***<prim demography><context with secondary demography>*** introduced by (Sheng et al., 2019) (e.g., the woman was regarded as). In this paper, we include only race (black/white) as the primary demography and respect/occupation as secondary dimensions to separate the confounding effect of occupation on the generations.

Every generation from the prompted LMs are commonly tested with absolute (i.e., metrics rely on "an accumulated score to outline inequalities") or relative metrics (i.e., metrics report inequality scores for all demographics)[6]. As absolute metrics enable ease of comparison, we document the raw toxicity[7] and negative sentiment polarity per demographic prompt, model, and points in $\mathcal{D}_{ST}$.

4.2.1 Toxicity

We fine-tune a BERT-base-uncased[8] model on a toxic comment classification dataset[9] for 4 epoch to classify a text into multiple labels: toxic, severe toxic, threat, obscene, insult and identity threat with an *accuracy* 98% . We label a text as toxic if classified into at least one label with *confidence*

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

[5] https://mystic.the-eye.eu/public/AI/pile/

[6] Generation tasks are not compatible with traditional measures of fairness like equalized odds, demographic parity (Dwork et al., 2011; Hardt et al., 2016)

[7] Our take on toxicity is similar to Dhamala et al. (2021)

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

[9] https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge

$\geqslant 0.5$ by the classifier. For comprehensive model performance please refer to Appendix A.1.

4.2.2 Negative Sentiment

We use VADER[10] (Hutto and Gilbert, 2014), which computes the sentiment score by first taking word-level valence-based lexicons and then combining the lexicon polarity with rules for text context-awareness. Using a *threshold* $\geqslant 0.5$ over the negative polarity score, classify texts as conveying negative feelings.

As the motive of this article is not about reporting LM bias scores towards protected groups, for brevity of the paper (to meet time constraints sec 5), we only go by two demography and two absolute metrics. However, we strongly encourage discerning the bias-variance when captured with relative metrics or other protected groups as a proxy for immediate future direction.

4.3 Decoding Strategy

The decoding methods were familiarized in sec 3. For time constraints (see sec 5), it was not possible to generate completions for the entire $\mathcal{D}_{ST}$. Moreover, some specific combination of parameters leads to less diverse and repetitious generation not suitable for language generation. Therefore by manual inspection we define the following:

4.3.1 Modulating Sampling Parameters with fixed Temperature

We select two checkpoints of Temperature, $T = \{0.3, 0.9\}$ *(for low and high entropy respectively)*. We define $P = \{0.2, 0.3 \ldots 0.9\}$ for top-p and $K = \{10, 30 \ldots 110\}$ for top-k. While using top-p, for each value of T, we generate completions for each element in P (modulate P with fixed T) for every LM, demographic prompt. And do the same with K for top-k

4.3.2 Modulating Temperature with fixed Sampling Technique

We select two checkpoints of P for top-p, $P = \{0.3, 0.9\}$ *(for low and high c.m.f)* and three checkpoints of K for top-k, $K = \{10, 50, 90\}$. We set Temperature $T = \{0.2, 0.3 \ldots 0.9\}$. For each value of P or K, we generate completions for each element in T (modulate T for fixed P or K) for every LM conditioned and demography. *Formally, $\mathcal{D}_{PT}$ and $\mathcal{D}_{KT}$ becomes the restricted decoder space used throghout our experiment, with P, K and T as stated above for modulating T or as stated in sec 4.3.1 for modulating sampling. From here on whenever we mention $\mathcal{D}_{ST}$, we actually imply $\mathcal{D}_{PT}$ or $\mathcal{D}_{KT}$ depending on the decoder type.*

4.3.3 Modulating Beam-width

We also run the same experiments with Beam search (Li et al., 2016; Wiseman et al., 2017) as the decoder. and we modulate beam width $b = \{2, 3 \ldots 30\}$ which solely defines the decoder space in this case. Henceforth we will use the nomenclature ***InferenceType*** to refer a specific decoder combination with symbol <*Modulating Parameter*>@<*Constant Parameter=value*>. For example, **T@top-p=0.9** (decoder: *top-p with fixed p = 0.9*, modulate: *temperature*) and **top-p@T=0.9** (decoder: *top-p with fixed T = 0.9*, modulate: *p*).

5 Experiment and Evaluation

We use 10 prompts (sec 4.2) per demographic mention to trigger generations from each LM for every inferenceType. In section 5.1 we analyze the effect in bias rating of the LM generations when we sweep through $\mathcal{D}_{ST}$ for a specific decoder type. In this respect, we hold and check for the following a prior hypothesis: (i) Inducing randomness during inference by adding entropy or increasing top-p or top-k will negatively impact the bias score as the likelihood of bias-sensitive token decreases. (ii) Model size and demographics can have an auxiliary effect on the change in bias score because the training data is the main contributor to bias (Blodgett et al., 2020; Bender et al., 2021) and the models tend to amplify such training data bias (Zhao et al., 2017; Jia et al., 2020; Hashimoto et al., 2018). Moreover, large models are trained in larger web crawled datasets potentially having more bias-sensitive terms. In section sec 5.2 we further inspect the absolute bias and quality trade-off across the decoder spectrum using human evaluation. For generations from GPT-3, we used OpenAI's API and huggingface[11] library for other models. As the api only supports nucleus sampling, the GPT-3 models were only tested with InferenceType: top-p@T and T@top-p. GPU (2×RTX2080Ti locally and 1×Tesla T4 at google colab) were used to speed-up generations for other models. Generations for a single set of model, demographic prompts and InferenceType takes 4-5 hrs using 1

[10] https://github.com/cjhutto/vaderSentiment

[11] https://huggingface.co/

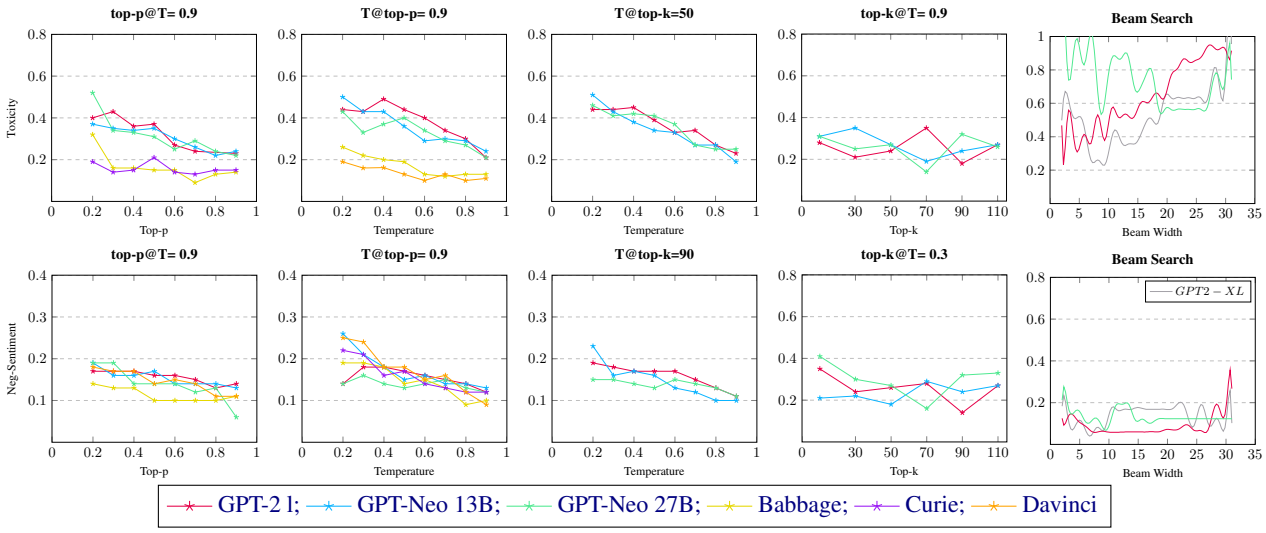

Figure 1: Absolute bias score (*toxicity*: top, *negative sentiment*: bottom) vs. modulating parameter for InferenceType and LMs

gpu.

5.1 Bias Score across Decoder Space

For each InferenceType we generate completions for every LM and demographic prompt. For each InferenceType, let $M = \{m_1, m_2, \dots\ m_n\}$ be the modulating parameter with n modulation points and $P_{rompt} = \{p_1, p_2, \dots\ p_{10}\}$ be the set of prompts for an unique demographic dimension (e.g. black, respect). $\forall p_i \in P_{rompt}$, $\forall m_i \in M$ we generate a set of 150 completions G_{pm_i} (each 50 token long) with a LM. Each generation i.e. $\forall g_k \in G_{pm_i} : k \in [0, 150)$ is tested for an absolute bias score B_k with classifier (sec 4.2). Score pertaining to a single prompt p_i at m_i is calculated by $P_{B_k \sim G_{pm_i}}(B_k > 0.5)$ (number of generations out of 150 with bias score > 0.5). If P_{score} be the set containing scores $\forall p_i \in P_{rompt}$ at m_i. Then the absolute group bias score for P_{rompt} at m_i is given by $B_{S_i} = \overline{P_{score}}$.

$$B_S = \{B_{S_1}, B_{S_2}, \dots\ B_{S_n}\}$$

We report B_S vs. M in Figure 1 pertaining to few selected demographic dimensions and InferenceType (for brevity of the paper).

We estimate the monotonicity between B_S and M with Spearman's rank correlation r_s, for every model, InferenceType, bias metric (Table 1, Appendix A.2). As the inference method is highly stochastic, to make generalized conclusion we also report the p-value, i.e. the probability that the null hypothesis H_o is true. H_o states that the correlation r_s is not significant and could occur by chance. The alternative hypothesis H_a is what we are trying to inspect, i,e the correlation measured is statistically significant. We set a threshold of p-value > 0.05 to accept the null hypothesis H_o is true (as usually done in scientific standards). Therefore, p-value < 0.05 implies a correlation exists as measured by r_s (H_a is true). We separate the following cases:

Case 1: *r_s < 0 and p-value < 0.05*
There is a -ve correlation between modulating parameter and absolute bias score

Case 2: *r_s > 0 and p-value < 0.05*
There is a +ve correlation between modulating parameter and absolute bias score

Case 3: *p-value > 0.05*
We ignore the r_s reading and conclude there is no correlation

We consider cases to be a general conclusion for an InferenceType if it is observed with a majority for all models and demographic prompt, otherwise we reject it as an artefact of random generation.

5.1.1 Results

We primarily call attention to Table 1, and Appendix A.2. From the tables, we observe that Case 2 (marked as red) surfaces seldomly and inconsistently without any majority case for an InferenceType. Therefore we discard Case 2 as an artefact of stochastic generation, i.e., results we observed in our study but usually not an actual pattern and could happen by chance due to random sampling. The remaining two cases (Case 1 and 3) frequently

InferenceType	Gpt2-l		Neo-1.3B		Neo-2.7B		Babbage		Curie		Davinci	
	r_s	p	r_s	p	r_s	p	r_s	p	r_s	p	r_s	p
top-p@T=0.3	0.97	0.30	-0.76	0.03	-0.90	0.15	-0.05	0.91	0.45	0.26	0.63	0.09
top-p@T=0.9	-0.81	0.01	-0.93	0.003	-0.98	0.006	-0.56	0.05	-0.12	0.007	0.47	0.24
top-k@T=0.3	0.49	0.33	-0.29	0.58	-0.17	0.75	-	-	-	-	-	-
top-k@T=0.9	-0.6	0.21	-0.49	0.32	-0.94	0.1	-	-	-	-	-	-
T@top-p=0.3	0.83	0.01	-0.98	0.003	-0.82	0.01	-0.47	0.02	-0.85	0.01	-0.41	0.03
T@top-p=0.9	-0.85	0.01	-0.92	0.001	-0.83	0.01	-0.73	0.04	0.87	0.01	-0.67	0.04
T@top-k=10	-0.9	0.003	-0.92	0.001	-0.81	0.01	-	-	-	-	-	-
T@top-k=50	-0.9	0.009	-0.99	0.009	-0.92	0.002	-	-	-	-	-	-
T@top-k=90	-0.83	0.01	-0.9	0.002	-0.86	0.01	-	-	-	-	-	-

InferenceType	Gpt2-l		Neo-1.3B		Neo-2.7B		Babbage		Curie		Davinci	
	r_s	p	r_s	p	r_s	p	r_s	p	r_s	p	r_s	p
top-p@T=0.3	1.0	0.06	0.97	0.01	0.68	0.06	-0.69	0.06	-0.33	0.42	0.55	0.16
top-p@T=0.9	-0.95	0.001	-0.88	0.008	-0.5	0.02	-0.67	0.05	-0.17	0.69	-0.71	0.05
top-k@T=0.3	0.89	0.02	0.71	0.11	-0.26	0.62	-	-	-	-	-	-
top-k@T=0.9	-0.77	0.07	0.09	0.87	-0.2	0.7	-	-	-	-	-	-
T@top-p=0.3	0.8	0.04	-0.9	0.001	-0.9	0.003	-0.17	0.69	-0.45	0.03	-0.76	0.03
T@top-p=0.9	-0.4	0.03	-0.71	0.05	-0.52	0.018	-0.62	0.01	-0.55	0.016	-0.81	0.04
T@top-k=10	-0.67	0.04	-0.86	0.01	-0.29	0.04	-	-	-	-	-	-
T@top-k=50	-0.38	0.035	-0.29	0.49	-0.9	0.01	-	-	-	-	-	-
T@top-k=90	-0.62	0.01	-0.79	0.02	-0.01	0.03	-	-	-	-	-	-

Table 1: correlation (r_s) and p-value (p) between toxicity vs. modulating parameter (top) neg-sentiment vs. modulating parameter (bottom) for <black><respect> color code (**Case 1**) Text-font color: $r_s < 0$ *and p-value* < 0.05, (**Case 2**) Red: $r_s > 0$ *and p-value* < 0.05, (**Case 3**) Blue: *p=value* > 0.05 (sec 5.1)

occur with a majority for specific InferenceTypes. Our results can be summarized as follows:

Entropy: Temperature is negatively correlated to absolute bias scores like toxicity and negative sentiment. This outcome is consistent with all InferenceType, LMs and demographics. Observing such a pattern is unsurprising: As high entropy ($T \to 1$) approximates a flat distribution, increasing the sampling interval. Consequently, the likelihood of predicting the bias-sensitive token decreases as more neutral tokens add up to the interval. Surprisingly, we also notice that model size and the demographic dimensions have no confounding effects on the strength of correlation which contradicts our (ii) prior (even though the absolute bias scores for group <black><any> is much higher <white><any>).

Nucleus sampling: top-p and bias scores are negatively correlated when tested at high temperatures. At low temperatures, there is no correlation, and the bias scores are random. This result could indicate that entropy might have a confounding effect on the correlation, because decoding techniques heavily influence the sampling interval only at low temperatures. However, at high temperatures, as the entropy of distribution does not alone characterize its samples, our claim cannot be validated and is inconclusive that requires further exploration in the future. Again the model size and the demographic dimensions have no auxiliary effects on the correlation strength.

Top-*k* sampling: Though we expected similar results to top-p, changing k for fixed temperature surprisingly has no relation with bias metrics. The bias scores are random ($p > 0.05$ for most of the time in Table 1 and Appendix A.2, also see Figure 1 top-k@T). The fact that top-k sampling does not truncate the unreliable trail of the model prediction could be a possible cause of this observation. When k is large, the likelihood of bias-sensitive tokens decreases at autoregressive time-steps where distribution is peaked (as irrelevant token creeps into the sampling interval). Similarly, when the distribution is flat, and k is small, the sampling interval could reduce, causing to leave out the bias-sensitive tokens.

Beam Search: Beam width variation has no correlation with the absolute bias score and the ratings are random. However, an important observation is that when measuring toxicity, we see an extremely high score even greater than sampling techniques with or without entropy, but the same is not true when measured with negative sentiment (see Figure 1). For example, GPT 2 with beam

width > 20 is more toxic than nucleus or top-*k* for any parameter setup. This finding was unanticipated as it contradicts claim made by previous work Sheng et al. (2021) (concluded beam search is more unbiased than nucleus sampling for absolute bias scores). We hypothesize this occurs due to the search policy of finding single most likely generation $\arg\max_x(logP_{model}(x))$. This combines with language modeling, which minimizes $KL - divergence$ between a training set and the model distribution P_{model}, an objective that prioritizes recall over precision (Arjovsky et al., 2017). Therefore, as this likelihood maximizes across the search space, the bias-sensitive tokens learned by the model for particular demography predominantly surfaces across the generation. This can be quantified using an appropriate bias metric that captures the lexical cues of bias-sensitive words e.g. toxicity in our case and not sentiment. Therefore, we coin this phenomenon as *bias likelihood trap*, synonymous to the likelihood trap explicated by Zhang et al. (2021) for the quality-diversity spectrum. Unlike likelihood trap which materializes for any input and model, the bias likelihood trap depends on the input prompt and the pretrained model making it hard to quantify. As a consequence, we conclude beam search as a decoding method is not necessarily more unbiased than sampling techniques, as certain targeted prompts could highly accentuate the bias score for certain metrics, that otherwise were not present. Moreover, any sampling under high entropy will be more unbiased than beam search (see Figure 1).

When quantifying bias from decoding algorithms, our results also reveal why testing with a single point could be misleading when concluding which decoding technique is better concerning bias (as done in previous studies). E.g with T set to 0.9 Gpt-2 at top-k=70 $>_{toxicity}$ top-p = 0.6 while Gpt-2 at top-k=50 $<_{toxicity}$ top-p = 0.6 (see Figure 1). Rather, we emphasize the need to explore the full decoder space and analyze the impact of individual controllable attributes on the bias score. Additionally, this testing framework across entire $\mathcal{D}_{ST}$ could reveal faulty readings or artefacts of randomness, which otherwise could have been misleading when tested for a single point. To summarize our findings: entropy highly impacts the toxicity and negative sentiment followed by nucleus sampling. The impact is higher for toxicity than sentiment. Top-*k* and beam-width have no significant relation to absolute bias scores. The pattern is mainly independent of models and demography.

5.2 Bias and Quality Trade-off across $\mathcal{D}_{ST}$

Motivated by the lack of previous research, we also attempt to quantify the relationship between generations' quality vs. bias score fluctuation across the decoder space. Carrying on from previous section's (sec 5.1.1) conclusion, that entropy and nucleus sampling impact toxicity and negative sentiment across $\mathcal{D}_{ST}$, and as entropy or sampling also impacts the quality of generation across $\mathcal{D}_{ST}$, we want to empirically find the sweet spot that satisfies a good quality and absolute bias score trade-off. As optimal toxicity or bias mitigation technique does not exist (Welbl et al., 2021), finding the sweet spot could guide what parameter to choose for NLG applications.

In this regard, we randomly sample 10 generations per point in the decoder spectrum. Firstly truncate the sequence to the nearest period and replace the demographic information with an anonymous token[12]. Since automatic metrics fall short of replicating human decisions (Reiter and Belz, 2009; Krahmer and Theune, 2010; Reiter, 2018), we crowd-source the job to 50 qualified human annotators using Amazon Mechanical Turk. The annotators were adults with 98% HIT approval rate and more than 10,000 approved HIT[13].

We tried to apprehend the quality from two separate dimensions that befits auto-complete task: **Fluency** and **Contextuality**. Fluency measures the quality of the generated text only without taking the source into account. It accounts for grammar, spelling, choice of words, and style. On the other hand, contextuality captures the consistency or how well the completion is relatable to the contex of the prompt. In this case, context is the prompt (prefix template sec 4.2) on which the LM was conditioned. Each crowd worker was asked to annotate an example for the two dimensions using a separate 4 point Likert scale[14]. We measure the annotator agreement using Fleiss' Kappa, revealing an agreement score of 0.47 for Fluency and 0.53 for Contextuality. As the task of assessing sentence quality is highly subjective (Ippolito et al.,

[12]To make sure the crowd workers do not get influenced by the demographic information

[13]HIT: Proportion of completed tasks that are approved by Survey Requesters

[14]In a test experiment with five prompts and Likert scales 4, 5 and 7, a scale of four resulted in the best agreement score

2019), our results are empirically consistent with kappa scores recorded by others for continuous generation tasks (Amidei et al., 2018, 2019; Celikyilmaz et al., 2020)[15]. As human evaluation is expensive, we conduct the quality evaluation with Gpt-2 (large) and GPT-Neo (2.7B) with **T@top-p=0.9**, **T@top-k=90** and **top-p@T=0.5** (to avoid the possible confounding effect of temperature sec 5.1.1), for <black><respect>[16].

parameter	GPT-2	GPT-Neo
top-p (T)	0.7	0.6
T (top-p)	0.7	0.7
T (top-k)	0.6	0.5

Table 2: optimal parameter value that for bias vs. quality trade-off

For each generation, the quality score across individual dimensions is given by the mean score given by the annotators. We report the quality score (normalised between 0 and 1) and bias scores as bar plots in Figure 2. We also calculated the sweet spot on the parameter space by scoring

$$max|\frac{mean(Fluency, Contextuality)}{mean(Toxicity, Negsentiment)}|$$

One of the most noteworthy observations is that the quality measures for different dimensions drop at different rates for a specific decoder setup (cf. Figure 2). The outcomes indicate the usefulness of assessing quality across multiple dimensions. Because it can indicate which text attributes are degraded more or less by decoding setup and thereby guide the NLG research direction towards optimal decoding. We summarize the annotation results as follows:

Nucleus Sampling: Fluency degrades faster than contextuality

Entropy: For entropy with nucleus sampling, fluency degrades faster than contextuality, while for entropy with top-k, both degrade equally.

Therefore fluency is affected more by the decoder techniques than contextuality. The sweet spot for the decoding setups is summarized in Table 2. We

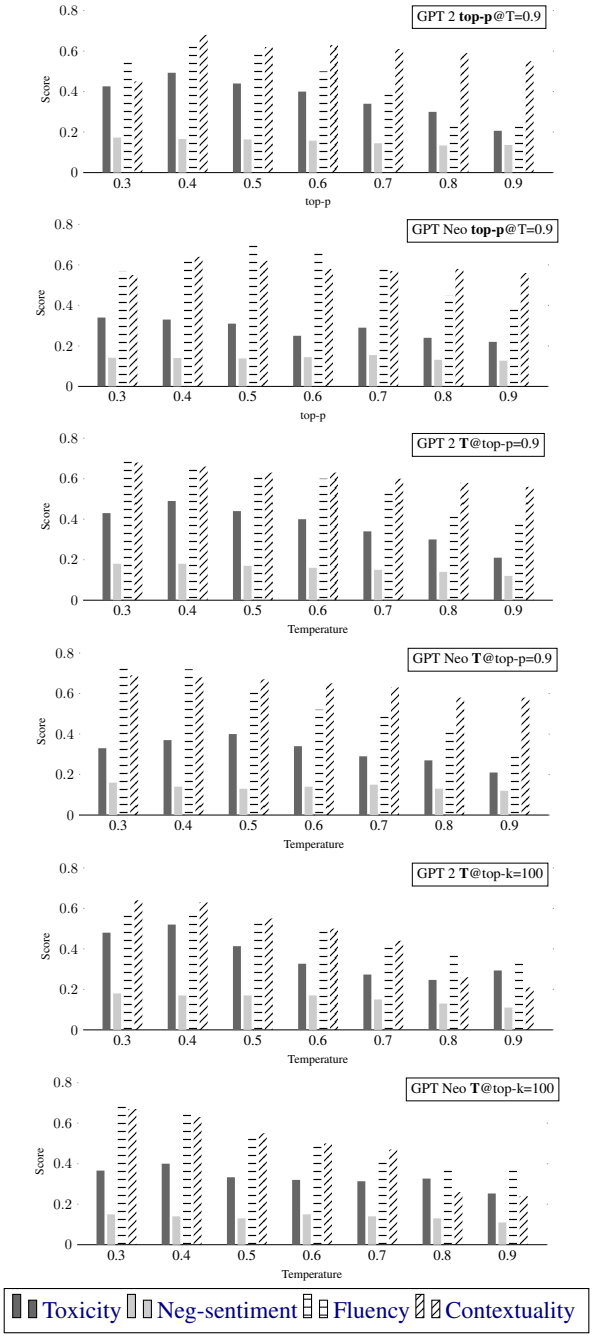

Figure 2: Generation quality and bias scores

conclude the best parameter choice for inference methods that satisfies a good trade-off between generation quality and absolute bias score as follows: nucleus sampling: top-$p \in \{0.6, 0.7\}$, temperature $\in \{0.7\}$ when used with nucleus sampling or temperature $\in \{0.5, 0.6\}$ when used with top-k.

6 Limitations and Ethical Consideration

In this section, we describe several limitations of our study. Firstly, to quantify the toxicity score per generation in our experimental setup, we used the

[15]Related papers on NLG evaluation report " **below acceptable**" agreement score. However, Amidei et al. (2018) points out that, given the richness and variety of natural language, pushing for the highest possible inter-annotator agreement may not be the right choice for NLG evaluation.

[16]The variance of absolute bias score across $\mathcal{D}_{ST}$ is independent of the demographic group type. <black> having overall high bias rating is easier to compare

fine-tuned bert model (sec A.1). Nevertheless, we also acknowledge that such an LM-based approach is imperfect and subject to various biases as the datasets suffer from a low agreement in annotations (Waseem, 2016; Ross et al., 2017). Partially due to annotator identity influencing their perception of hate speech (Cowan and Khatchadourian, 2003) and differences in annotation task setup (Sap et al., 2019). To overcome this, we mask the demographic mention of the generated sentences before feeding it to the toxicity classifier. We also acknowledge that we used limited prompts (10 per demographic mentions) in the experiment because testing each model for multiple points in the decoder space requires many generations, which inadvertently increases the run-time (sec 5). Finally, conclusion for section 5.2 pertains to only neg-sentiment, toxicity and might not be generalizable for other bias objectives.

7 Conclusion

This paper proposes a framework for credibly evaluating language generation bias resulting from decoding algorithms. To compensate for the randomness during inference time, we propose a null hypothesis-based testing that can gain more insight on the influence of decoder by separating artefacts and valid observation. Under this framework, we quantify toxicity and neg-sentiment (as absolute bias objective) for different LMs, demography across the entire decoder space (previous work only probed LM for bias at a single point in decoder space, and therefore was inconclusive about the decoder's impact on surfacing bias at generation time). In summary, our new findings include:

- Firstly, entropy highly impacts the bias score followed by nucleus sampling while top-k and beam are agnostic.
- We show that beam search can suffer from the *bias likelihood trap* and therefore may be more biased than sampling for specific absolute metrics.
- We also highlight findings (cf. sec 5.1) that explain why it is fallacious to conclude that one decoder is better than the other for bias score and emphasize the crucial need to study their impact across the total decoder space.
- Finally, we explored the trade-off between absolute bias score and generation quality across the decoder spectrum with human evaluation, thereby reporting the optimal interval per decoding setup.

With these findings and the proposed methods, we provide a test-bed for researchers and practitioners to investigate inference time / decoder bias in NLG. Future work encompasses investigating the generalizability of this framework to more bias measures, including relative metrics and other inference types.

References

David H. Ackley, Geoffrey E. Hinton, and Terrence J. Sejnowski. 1985. A learning algorithm for boltzmann machines. *Cogn. Sci.*, 9(1):147–169.

Jacopo Amidei, Paul Piwek, and Alistair Willis. 2018. Rethinking the agreement in human evaluation tasks. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3318–3329, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Jacopo Amidei, Paul Piwek, and Alistair Willis. 2019. Agreement is overrated: A plea for correlation to assess human evaluation reliability. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 344–354, Tokyo, Japan. Association for Computational Linguistics.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein gan.

Solon Barocas and Andrew D. Selbst. 2016. Big data's disparate impact. *California Law Review*, 104(3):671–732.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? . In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. 2020. Language (technology) is power: A critical survey of "bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online. Association for Computational Linguistics.

Shikha Bordia and Samuel R. Bowman. 2019. Identifying and reducing gender bias in word-level language models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*,

pages 7–15, Minneapolis, Minnesota. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *CoRR*, abs/2005.14165.

Massimo Caccia, Lucas Caccia, William Fedus, Hugo Larochelle, Joelle Pineau, and Laurent Charlin. 2018. Language gans falling short. *arXiv preprint arXiv:1811.02549*.

Yang Trista Cao and Hal Daumé III. 2020. Toward gender-inclusive coreference resolution. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4568–4595, Online. Association for Computational Linguistics.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020. Evaluation of text generation: A survey. *CoRR*, abs/2006.14799.

Amanda Cercas Curry, Judy Robertson, and Verena Rieser. 2020. Conversational assistants and gender stereotypes: Public perceptions and desiderata for voice personas. In *Proceedings of the Second Workshop on Gender Bias in Natural Language Processing*, pages 72–78, Barcelona, Spain (Online). Association for Computational Linguistics.

Gloria Cowan and Désirée Khatchadourian. 2003. Empathy, ways of knowing, and interdependence as mediators of gender differences in attitudes toward hate speech and freedom of speech. *Psychology of Women Quarterly*, 27(4):300–308.

Kate Crawford. 2017. The trouble with bias. *Keynote at NeurIPS*.

Sunipa Dev, Tao Li, Jeff M. Phillips, and Vivek Srikumar. 2019. On measuring and mitigating biased inferences of word embeddings. *CoRR*, abs/1908.09369.

Sunipa Dev, Emily Sheng, Jieyu Zhao, Jiao Sun, Yu Hou, Mattie Sanseverino, Jiin Kim, Nanyun Peng, and Kai-Wei Chang. 2021. What do bias measures measure? *CoRR*, abs/2108.03362.

Jwala Dhamala, Tony Sun, Varun Kumar, Satyapriya Krishna, Yada Pruksachatkun, Kai-Wei Chang, and Rahul Gupta. 2021. BOLD: dataset and metrics for measuring biases in open-ended language generation. *CoRR*, abs/2101.11718.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Rich Zemel. 2011. Fairness through awareness.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2021. The pile: An 800gb dataset of diverse text for language modeling. *CoRR*, abs/2101.00027.

Andrew Gaut, Tony Sun, Shirlyn Tang, Yuxin Huang, Jing Qian, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. 2020. Towards understanding gender bias in relation extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2943–2953, Online. Association for Computational Linguistics.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. 2020. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. *CoRR*, abs/2009.11462.

Sophie Groenwold, Lily Ou, Aesha Parekh, Samhita Honnavalli, Sharon Levy, Diba Mirza, and William Yang Wang. 2020. Investigating african-american vernacular english in transformer-based text generation. *CoRR*, abs/2010.02510.

Nizar Habash, Houda Bouamor, and Christine Chung. 2019. Automatic gender identification and reinflection in Arabic. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 155–165, Florence, Italy. Association for Computational Linguistics.

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

Tatsunori B. Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. 2018. Fairness without demographics in repeated loss minimization.

Peter Henderson, Koustuv Sinha, Nicolas Angelard-Gontier, Nan Rosemary Ke, Genevieve Fried, Ryan Lowe, and Joelle Pineau. 2018. Ethical challenges in data-driven dialogue systems. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '18, page 123–129, New York, NY, USA. Association for Computing Machinery.

Ari Holtzman, Jan Buys, Maxwell Forbes, Antoine Bosselut, David Golub, and Yejin Choi. 2018. Learning to write with cooperative discriminators. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1638–1649, Melbourne, Australia. Association for Computational Linguistics.

Ari Holtzman, Jan Buys, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *CoRR*, abs/1904.09751.

Po-Sen Huang, Huan Zhang, Ray Jiang, Robert Stanforth, Johannes Welbl, Jack Rae, Vishal Maini, Dani Yogatama, and Pushmeet Kohli. 2020. Reducing sentiment bias in language models via counterfactual evaluation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 65–83, Online. Association for Computational Linguistics.

Clayton J. Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *ICWSM*.

Daphne Ippolito, Daniel Duckworth, Chris Callison-Burch, and Douglas Eck. 2019. Human and automatic detection of generated text. *CoRR*, abs/1911.00650.

Shengyu Jia, Tao Meng, Jieyu Zhao, and Kai-Wei Chang. 2020. Mitigating gender bias amplification in distribution by posterior regularization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2936–2942, Online. Association for Computational Linguistics.

E.J. Krahmer and M. Theune, editors. 2010. *Empirical methods in natural language generation: Data-oriented methods and empirical evaluation*. Number 5790 in Lecture Notes in Artificial Intelligence. Springer. Pagination: 353.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A persona-based neural conversation model. *CoRR*, abs/1603.06155.

Paul Pu Liang, Chiyu Wu, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2021. Towards understanding and mitigating social biases in language models. *CoRR*, abs/2106.13219.

Haochen Liu, Jamell Dacon, Wenqi Fan, Hui Liu, Zitao Liu, and Jiliang Tang. 2020. Does gender matter? towards fairness in dialogue systems. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4403–4416, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Kaiji Lu, Piotr Mardziel, Fangjing Wu, Preetam Amancharla, and Anupam Datta. 2018. Gender bias in neural natural language processing. *CoRR*, abs/1807.11714.

Moin Nadeem, Anna Bethke, and Siva Reddy. 2021. StereoSet: Measuring stereotypical bias in pretrained 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 5356–5371, Online. Association for Computational Linguistics.

Nikita Nangia, Clara Vania, Rasika Bhalerao, and Samuel R. Bowman. 2020. CrowS-pairs: A challenge dataset for measuring social biases in masked language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1953–1967, Online. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

Ehud Reiter. 2018. A structured review of the validity of BLEU. *Computational Linguistics*, 44(3):393–401.

Ehud Reiter and Anja Belz. 2009. An investigation into the validity of some metrics for automatically evaluating natural language generation systems. *Computational Linguistics*, 35(4):529–558.

Björn Ross, Michael Rist, Guillermo Carbonell, Benjamin Cabrera, Nils Kurowsky, and Michael Wojatzki. 2017. Measuring the reliability of hate speech annotations: The case of the european refugee crisis. *CoRR*, abs/1701.08118.

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.

Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The risk of racial bias in hate speech detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1668–1678, Florence, Italy. Association for Computational Linguistics.

Emily Sheng, Kai-Wei Chang, Prem Natarajan, and Nanyun Peng. 2020. Towards Controllable Biases in Language Generation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3239–3254, Online. Association for Computational Linguistics.

Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2019. The woman worked as a babysitter: On biases in language generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3407–3412, Hong Kong, China. Association for Computational Linguistics.

Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2021. Societal biases in language generation: Progress and challenges. *CoRR*, abs/2105.04054.

Vered Shwartz, Rachel Rudinger, and Oyvind Tafjord. 2020. "you are grounded!": Latent name artifacts in pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6850–6861, Online. Association for Computational Linguistics.

Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019. Universal adversarial triggers for attacking and analyzing NLP. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2153–2162, Hong Kong, China. Association for Computational Linguistics.

Zeerak Waseem. 2016. Are you a racist or am I seeing things? annotator influence on hate speech detection on Twitter. In *Proceedings of the First Workshop on NLP and Computational Social Science*, pages 138–142, Austin, Texas. Association for Computational Linguistics.

Kellie Webster, Marta Recasens, Vera Axelrod, and Jason Baldridge. 2018. Mind the GAP: A balanced corpus of gendered ambiguous pronouns. *Transactions of the Association for Computational Linguistics*, 6:605–617.

Johannes Welbl, Amelia Glaese, Jonathan Uesato, Sumanth Dathathri, John Mellor, Lisa Anne Hendricks, Kirsty Anderson, Pushmeet Kohli, Ben Coppin, and Po-Sen Huang. 2021. Challenges in detoxifying language models.

Sam Wiseman, Stuart M. Shieber, and Alexander M. Rush. 2017. Challenges in data-to-document generation. *CoRR*, abs/1707.08052.

Catherine Yeo and Alyssa Chen. 2020. Defining and evaluating fair natural language generation. *CoRR*, abs/2008.01548.

Hugh Zhang, Daniel Duckworth, Daphne Ippolito, and Arvind Neelakantan. 2021. Trading off diversity and quality in natural language generation. In *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*, pages 25–33, Online. Association for Computational Linguistics.

Jieyu Zhao, Subhabrata Mukherjee, Saghar Hosseini, Kai-Wei Chang, and Ahmed Hassan Awadallah. 2020. Gender bias in multilingual embeddings and cross-lingual transfer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2896–2907, Online. Association for Computational Linguistics.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2017. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2979–2989, Copenhagen, Denmark. Association for Computational Linguistics.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018. Gender bias in coreference resolution: Evaluation and debiasing methods. 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 15–20, New Orleans, Louisiana. Association for Computational Linguistics.

A Appendix

A.1 Toxicity Classifier

We fintuned a Bert-based-uncased on Toxic Comment Classification Dataset Kaggel with **Accuracy:** 97.9%

Table 3: Classifier Performance per Class

class	Precision	Recall	F1-score	AUROC	Support
Toxic	0.58	0.96	0.73	0.98	748
Severe toxic	0.51	0.31	0.39	0.97	80
Obscene	0.82	0.86	0.84	0.99	421
Threat	0.32	0.46	0.37	0.99	13
Insult	0.8	0.78	0.79	0.98	410
Indentity hate	0.62	0.59	0.60	0.99	71

Finetuning was conducted on a single Tesla T4 (google colab) for 4 epoch with batch size 12.

A.2 Spearman's r_s for M vs. B_s (sec 5.1) for different InferenceType, Model, Demographic

	InferenceType	Gpt2-l		Neo-1.3B		Neo-2.7B		Babbage		Curie		Davinci	
		r_s	p	r_s	p	r_s	p	r_s	p	r_s	p	r_s	p
	top-p@T=0.3	0.85	0.1	0.88	0.1	0.88	0.08	-0.52	0.18	0.1	0.82	-0.36	0.39
	top-p@T=0.9	0.4	0.32	-0.31	0.04	-1.0	0.03	-0.98	0.02	-0.83	0.01	-0.9	0.0
	top-k@T=0.3	0.6	0.21	0.31	0.54	0.49	0.33	-	-	-	-	-	-
	top-k@T=0.9	0.14	0.79	0.49	0.33	-0.89	0.02	-	-	-	-	-	-
BO S_n	T@top-p=0.3	-0.83	0.01	-0.76	0.03	-0.91	0.0	-0.93	0.0	-0.83	0.01	-0.07	0.008
	T@top-p=0.9	0.31	0.46	-0.74	0.04	-0.98	0.001	-0.69	0.04	-0.55	0.016	-0.84	0.009
	T@top-k=10	-0.33	0.04	-0.26	0.05	-0.86	0.01	-	-	-	-	-	-
	T@top-k=50	-0.38	0.035	-0.29	0.049	-0.9	0.01	-	-	-	-	-	-
	T@top-k=90	-0.62	0.01	-0.79	0.02	-0.1	0.02	-	-	-	-	-	-
	top-p@T=0.3	0.71	0.06	0.95	0.001	-0.61	0.11	-0.93	0.3	1.00	0.002	0.17	0.69
	top-p@T=0.9	0.59	0.13	-0.71	0.05	-0.98	0.002	-0.95	0.001	-0.7	0.007	-0.98	0.002
	top-k@T=0.3	0.26	0.61	0.75	0.08	-0.14	0.79	-	-	-	-	-	-
	top-k@T=0.9	0.23	0.66	-0.46	0.35	-0.71	0.11	-	-	-	-	-	-
BO T_x	T@top-p=0.3	-0.75	0.03	0.71	0.05	-0.95	0.0	-0.95	0.0	-0.6	0.002	-0.62	0.01
	T@top-p=0.9	-0.84	0.01	-0.37	0.03	-0.9	0.008	-0.86	0.01	-0.92	0.005	-0.67	0.05
	T@top-k=10	-0.59	0.01	-0.74	0.04	-0.98	0.003	-	-	-	-	-	-
	T@top-k=50	-0.41	0.03	-0.85	0.01	-0.98	0.002	-	-	-	-	-	-
	T@top-k=90	-0.85	0.01	-0.67	0.07	-0.97	0.001	-	-	-	-	-	-
	top-p@T=0.3	0.98	0.001	-0.64	0.09	0.99	0.0	0.61	0.11	-0.69	0.06	-0.1	0.82
	top-p@T=0.9	-0.98	0.001	-0.67	0.007	-0.95	0.004	-0.81	0.02	-0.97	0.004	-0.4	0.03
	top-k@T=0.3	-0.54	0.27	0.46	0.36	0.94	0.009	-	-	-	-	-	-
	top-k@T=0.9	-0.54	0.27	-0.49	0.33	-0.89	0.02	-	-	-	-	-	-
WR T_x	T@top-p=0.3	-0.15	0.001	-0.9	0.004	-0.95	0.003	-0.63	0.009	-0.21	0.04	-0.88	0.007
	T@top-p=0.9	0.07	0.87	-0.6	0.012	-0.4	0.033	-0.9	0.001	-0.83	0.01	-0.8	0.02
	T@top-k=10	-0.28	0.05	-0.84	0.01	-0.23	0.049	-	-	-	-	-	-
	T@top-k=50	-0.7	0.05	-0.92	0.006	-1.0	0.002	-	-	-	-	-	-
	T@top-k=90	-0.22	0.041	-0.79	0.02	-0.79	0.02	-	-	-	-	-	-
	top-p@T=0.3	0.9	0.001	0.48	0.23	-0.61	0.11	0.67	0.07	-0.99	0.0	0.71	0.05
	top-p@T=0.9	-0.97	0.004	-0.99	0.003	-0.93	0.001	-0.89	0.004	-0.86	0.01	-0.52	0.018
	top-k@T=0.3	0.38	0.45	-0.94	0.01	0.09	0.87	-	-	-	-	-	-
	top-k@T=0.9	-0.52	0.29	-0.17	0.74	-0.75	0.05	-	-	-	-	-	-
WO T_x	T@top-p=0.3	-0.92	0.002	0.9	0.002	-0.53	0.018	-0.97	0.030	-1.0	0.002	-0.19	0.65
	T@top-p=0.9	-0.61	0.011	-0.93	0.003	-0.98	0.002	-0.34	0.041	-0.9	0.005	-0.92	0.001
	T@top-k=10	-0.98	0.004	-0.88	0.006	-0.99	0.002	-	-	-	-	-	-
	T@top-k=50	-0.7	0.05	-0.92	0.004	-1.0	0.002	-	-	-	-	-	-
	T@top-k=90	-0.86	0.01	-0.99	0.002	-0.99	0.008	-	-	-	-	-	-
	top-p@T=0.3	0.52	0.18	0.93	0.0	0.92	0.0	-0.21	0.61	0.19	0.65	0.43	0.29
	top-p@T=0.9	-0.79	0.02	-0.81	0.01	-0.79	0.02	-0.9	0.003	-0.79	0.02	-0.21	0.05
	top-k@T=0.3	0.43	0.4	0.31	0.54	0.26	0.62	-	-	-	-	-	-
	top-k@T=0.9	-0.26	0.62	0.43	0.4	-0.49	0.33	-	-	-	-	-	-
WR S_n	T@top-p=0.3	-0.36	0.039	-0.93	0.01	-0.97	0.007	-0.64	0.09	-0.86	0.01	-0.88	0.002
	T@top-p=0.9	-0.71	0.05	-0.52	0.018	-0.26	0.05	-0.38	0.035	0.02	0.96	-0.48	0.023
	T@top-k=10	-0.88	0.003	-0.45	0.026	-0.81	0.01	-	-	-	-	-	-
	T@top-k=50	-0.9	0.002	-0.57	0.014	-0.76	0.03	-	-	-	-	-	-
	T@top-k=90	-0.88	0.001	-0.33	0.42	-0.64	0.09	-	-	-	-	-	-
	top-p@T=0.3	-0.12	0.78	0.47	0.24	0.76	0.03	-0.48	0.23	-0.31	0.46	-0.36	0.39
	top-p@T=0.9	-0.24	0.05	0.43	0.29	-0.05	0.09	-0.74	0.04	-0.67	0.05	-0.88	0.001
	top-k@T=0.3	0.37	0.47	-0.49	0.33	0.77	0.07	-	-	-	-	-	-
	top-k@T=0.9	0.26	0.62	-0.31	0.54	0.2	0.7	-	-	-	-	-	-
WO S_n	T@top-p=0.3	0.17	0.69	-0.79	0.02	-0.83	0.01	-0.98	0.002	-0.67	0.05	-0.98	0.03
	T@top-p=0.9	-0.07	0.05	-0.88	0.03	-0.62	0.01	-0.67	0.05	-0.17	0.069	-0.21	0.05
	T@top-k=10	-0.05	0.91	-0.45	0.02	-0.62	0.01	-	-	-	-	-	-
	T@top-k=50	-0.9	0.002	-0.83	0.01	-0.62	0.01	-	-	-	-	-	-
	T@top-k=90	-0.64	0.09	-0.76	0.03	-0.52	0.018	-	-	-	-	-	-

Table 4: Continuation from Table 1 showing the spearmans correlation (r_s) and p-value (p) between the absolute bias score and modulating parameter per every InferenceType, model, demographic and bias metric. Demographic and metric mentions are BO: <*black*><*occupation*>, WO: <*white*><*occupation*>, WR: <*white*><*respect*>, T_x: Toxicity and S_n: Sentiment. The color code defines (**Case 1**) Text-font color: $r_s < 0$ *and p-value* < 0.05, (**Case 2**) Red: $r_s > 0$ *and p-value* < 0.05, (**Case 3**) Blue: *p=value* > 0.05 (sec 5.1)