Reproducibility Report: Understanding the Effect of Model Compression on Social Bias in Large Language Models

Jessica Colleran, Nicolas Hitosis, Aidan Jay

Spring 2024

Reproducibility Summary

Motivation

Given the current popularity of Large Language Models (LLMs), the need to mitigate social bias in LLMs and reduce computational costs is becoming increasingly important. While we want LLMs to have the best language modeling capabilities possible, they should not exhibit biased behavior and should not perpetuate harmful stereotypes. Gonçalves and Strubell [2023] studies the effects quantization and distillation have on LLMs and how these techniques may remedy these issues.

Scope of Reproducibility

The original paper claims that longer pretraining and larger models elicit higher social bias, but that quantization and/or distillation can reduce this. We mainly reproduce results pertaining to model compression.

Methodology

We leveraged the repository provided in the original paper and conducted experiments using the BERT and Pythia LLM families on 2 GPUs and 2 CPUs. In all, producing the results took about 24 hours. For quantization, we used PyTorch's built in post-training quantization (PTQ) functionality. However, torch's PTQ is CPU-only, thus all the quantized experiments ran on CPU exclusively.

Results

Our study reproduced the StereoSet and CrowS-Pairs bias scores in the paper to within 1% of the original values, which strongly supports the paper's hypothesis that model compression decreases bias. From additional testing outside of the paper, we find the same results with regards to model compression. However, the effect of model size and pretraining on bias does not seem as strong as the paper suggests. This is evidenced by experiments like CrowS-Pairs where BERT Large was generally less biased than BERT.

What was Easy

After cloning and setting up the author's repository, executing the experiments was straightforward due to the repository's file structure and provision of bash scripts designed to run each experiment. All the quantization code was also included and worked fine with correct dependencies which saved us a lot of time as well. Note that the datasets were included in the repository which made running the experiments relatively easy as well.

What was Difficult

The most challenging aspect of the reproduction was ensuring the reliability and functionality of the provided code. This involved several complexities, such as diagnosing issues, selectively commenting out portions of the code for specific experiments, and installing specific dependency versions. Additionally, certain scripts failed to perform as expected, such as checking for already computed results.

Communication with Original Authors

No contact was made with the authors of the original paper over the course of our research.

1 Introduction

Large Language Models (LLMs) are rapidly increasing in size and popularity as interest in Natural Language Processing (NLP) grows. However, training models of these large proportions require copious datasets, which subsequently contain human bias. While we want models to have the best language modeling capabilities possible, they should not exhibit biased behavior that perpetuate harmful stereotypes. Gonçalves and Strubell [2023] examines how a model's bias correlates with another issue facing LLMs: model size. Given the size of LLMs right now, the need to reduce the computational cost of these models is becoming increasingly important. As such, techniques like quantization and distillation have arisen. Quantization refers to when a model's weights and activations are reduced to a smaller bit space. The paper uses 8-bit integer quantization, meaning the model's original 32-bit floating point parameters are shrunk down to 8-bit integers. On the other hand, distillation refers to using a large model to train a smaller model and using the smaller model for inference. Thus, the paper examines how model compression can reduce computational cost and potentially debias models.

2 Scope of Reproducibility

Our scope includes reproducing the results of Gonçalves and Strubell [2023], surrounding the effect of model compression, specifically quantization and distillation, on reducing bias.

For quantization, we run all quantized experiments with int8 dynamic Post Training Quantization (PTQ), meaning the model weights are reduced to 8-bit integers before inference and the activations are reduced to 8-bit integers dynamically during inference. Specifically, we are testing how int8 dynamic PTQ affects bias on BERT, RoBERTA, and the Pythia models by evaluating them on the StereoSet, CrowS-Pairs, and SEAT datasets. To test distillation, we evaluate distilBERT and distilRoBERTa on the same bias benchmarks for consistency. To validate lower bias, we expect the bias scores of the compressed models to be "less biased" than of the regular models. For StereoSet, CrowS-Pairs, and SEAT, we expect the bias scores for compressed models to be closer to 50 (the gold score for these benchmarks) than non-compressed models.

Outside of the experiments run in the original paper, we test the effects of quantization on bias further by evaluating the Pythia models on the Professions and WinoBias benchmarks. For Professions and WinoBias, we expect the bias scores to be lower than non-compressed models, as the scores are more loosely defined for this benchmark.

Note that while the original paper extensively tests the effects of pretraining length on bias, longer pretraining (more datapoints used in pretraining the model) is expected to increase bias. However, this requires running experiments on multiple model checkpoints, which we do not possess sufficient computational resources to do. Thus, we reproduce only a select few experiments from the paper to test this hypothesis and do not test for it in our newly introduced experiments.

2.1 Addressed Claims from the Original Paper

1. Longer pretraining and larger LLMs elicit higher social bias by generating stereotypical continuations with a higher probability than non-stereotypical continuations.

2. Quantization and distillation reduce bias in LLMs by causing the LLMs to generate stereotypical and non-stereotypical continuations with more equal probabilities than their non-compressed counterparts.

3 Methodology

In our replication, we aimed to reproduce the results for a select few of the experiments performed in the original paper. To do this, we leveraged the repository provided by the authors, in addition to an Nvidia RTX 4090, an Nvidia RTX 2080 Ti, and an Intel Xeon Gold 6132 Processor.

3.1 Model Descriptions

In our reproduction, our learning objective and models were similar to that of the original paper. As such, we used knowledge distillation and quantization as our model compression techniques. We were also interested in understanding the relationship between pretrained model size and measures of social bias, and how social bias changed as a function of how well the model fits the data. We worked with three base LLMs: BERT, RoBERTa, and Pythia. BERT comes in 2 sizes: 110M and 336M. RoBERTa also comes in 2 sizes: 125M and 355M. Pythia has model sizes ranging from 70M parameters to 6.9B parameters. All of these models are pretrained, thus we are simply loading in those published weights from HuggingFace.

The BERT models were trained on BookCorpus and Wikipedia. The RoBERTa models were trained on BookCorpus, Wikipedia, CC-News, OpenWebText, and the Stories datasets. All of the Pythia models were trained on the exact same deduped version of the Pile dataset. Additionally, we worked with pre-training lengths ranging from 1K to 143K by using checkpoints of the Pythia models (although our reproduction focuses less on this aspect). Also note that our extension experiments not in the original paper were run exclusively using the set of Pythia models. Refer to Table 1 to see a full list of all models used in this paper.

LLM Name	LLM Size	Training Objective	Quantization Used
BERT Base	110M	Masked LM	int8
BERT Large	336M	Masked LM	None
DistilBERT	67M	Masked LM	None
RoBERTa Base	125M	Masked LM	int8
RoBERTa Large	355M	Masked LM	None
DistilRoBERTa	82.8M	Masked LM	None
Pythia	70M	Causal LM	int8
Pythia	160M	Causal LM	int8
Pythia	410M	Causal LM	int8
Pythia	1.4B	Causal LM	int8
Pythia	2.8B	Causal LM	int8
Pythia	6.9B	Causal LM	int8

Table 1: All models used in testing

3.2 Datasets

We used five datasets to evaluate the models, three of which came from the original paper - CrowS-Pairs [Nangia et al., 2020] and StereoSet [Nadeem et al., 2020]- and two of which our team added to further examine the claims from the paper on different types of bias benchmarks - WinoBias [Zhao et al., 2018] and Professions Dataset [Vig et al., 2020]. The original datasets were obtained from the repository provided by the authors and the datasets added to expand the experiments were obtained from a repository for a paper on identifying gender bias in LLMs [Chintam et al., 2023].

CrowS-Pairs evaluates social bias in an LLM. Each sample in the CrowS-Pairs dataset consists of pairs of sentences where one is more stereotyping than the other. The sentences are almost the same with only the subject of the sentence being changed from a person/race/group/etc that is associated with the common stereotype addressed in the sentence's context to a person/race/group/etc that is not stereotypically associated

with the context. A model that is unbiased will ideally score 50% on the dataset. CrowS-Pairs consists of 1,508 crowd-sourced examples such that samples are each annotated with what bias category it belongs to (gender, race, etc.). Although CrowS-Pairs has many bias types, we focus on the Gender (262 examples), Race (516 examples), and Religion (106 examples) categories of the dataset.

StereoSet measures social bias in addition to the language modeling capabilities of an LLM. Thus, models evaluated on the StereoSet benchmark produce two scores: a bias score and a Language Model (LM) score. Each sample is human-annotated and consists of a context sentence and a stereotypical, anti-stereotypical, and unrelated answer. An unbiased model will yield a bias score of 50% between stereotypical and anti-stereotypical answers. If a model chooses the unrelated answer a high percentage of the time, then the model is considered to have low language modeling capabilities and will have a low LM score. The main score used here is the bias score and the LM score is a mere reference point. For our experiments, we focus on the Gender, Race, and Religion bias types and the intrasentence context type. This portion of StereoSet has 5290 total examples with 1026 examples in Gender bias (19.4%), 3996 examples in Race bias (75.5%), and 623 examples in Religion bias (11.8%).

WinoBias evaluates gender bias in an LLM. The WinoBias dataset is split into two test types. Type 1 is formatted "[entity1] [interacts with] [entity2] [conjunction] [pronoun] [circumstances]" while Type 2 is formatted "[entity1] [interacts with] [entity2] and then [interacts with] [pronoun] for [circumstances]". Type 1 is more difficult for models to decipher the pronoun's corresponding entity because they contain no syntactic cues whereas Type 2 has linguistic clues that make it more clear of the pronoun's entity. In total, the WinoBias dataset contains 3,160 sentences, split equally for development and test. Since each example consists of two sentences, the dataset split includes 395 examples for each: Type 1 dev, Type 1 test, Type 2 dev, Type 2 test. Bias is measured by how the model assigns probability of the stereotypical sentence compared to the anti-stereotypical sentence. In our experiment, we calculate the proportion of examples where the model assigns higher probability to the pro-stereotypical version.

Professions measures social bias, specifically gender bias, in LLMs. Each example in Professions consists of a prompt with the subject being a profession known to have a major gender gap, such as "the profession said that", and the model must assign the probability of the following pronoun being either she or he. The dataset uses 17 templates ("the [profession] [verb] that") and 299 professions for a total of 5083 examples. The dataset is split into 70 female-stereotypical professions and 229 male-stereotypical professions. Bias is computed as the probability of the anti-stereotypical pronoun given the profession divided by the probability of the stereotypical pronoun given the profession. In our experiment, we calculate the percentage of times the model gives a higher probability to the stereotypical pronoun. Thus the lower the percentage, the less bias the model has.

3.3 Hyperparameters

The hyperparameter values were set to the same as those from the original paper. We did not do any pre-training or hyperparameter tuning. The experiments we run all consist of running forward passes on the bias benchmarks to essentialy compute likelihoods. As such, hyperparameters like temperature also do not apply as no generation is done in the experiments.

3.4 Implementation

We utilized the existing repository provided by the authors, subsequently adapting the code to be in accordance with our hardware configuration.

Our repository contains all the data required so no download is required. The code is mainly written in Python and Bash and leverages packages such as PyTorch, SciPy, and the Hugging Face Transfomers library,

to name a few. The full list of dependencies can be retrieved by executing "pip show" in the root directory of the repository. To execute experiments without a cluster, ensure the corresponding Bash script has the sbatch command(s) disabled and execute the script.

For the additional experiments, we transformed the repository used in a different paper on reducing gender bias in LLMs. Since this paper used the different bias benchmarks that we wanted to experiment on, we only needed to input the Pythia models into the code. Our code transformation is in this updated repository and the experiments can be run by running the evaluate.py file after inputting. The parameters required by the experiment can be found in Tables 5-8. The quantized version of the models can be acquired by uncommenting the quantization lines in evaluate.py.

3.5 Experimental Setup

We ran our experiments on either an Nvidia RTX 4090, an Nvidia RTX 2080 Ti, an Intel Xeon Gold 6132 Processor, or an Intel i7 14700k. All non-quantized LLM experiments were run one one of the two GPUs. All quantized experiments were run on one of the two CPUs. This is becuase PyTorch's implementation of dynamic PTQ that we used is CPU-only.

3.6 Computational Requirements

StereoSet experiments for non-quantized LLMs were expected to take a total of 10 hours and another 15 for the quantized LLMs. In reality, this took 20 GPU hours on the 2080Ti for the non-quantized experiments and 11.5 CPU hours on the 14700k for the quantized experiments.

Crow-S experiments were estimated to take around 24 hours on the Intel Xeon CPU. In actuality, performing the Crow-S experiments required a total of 31 hours.

WinoBias experiments for non-quantized LLMs took 2 GPU hours on the Nvidia RTX 4090 and 0.37 CPU hours (22.5 minutes) on the Intel i7 14700k for the quantized experiments.

Professions experiments for non-quantized LLMs took 6 GPU hours on the Nvidia RTX 4090 and 2 CPU hours on the Intel i7 14700k for the quantized experiments.

Note that for all experiments, running the large models (1B) requires a lot of RAM. The 6.9B model required upwards of 50gb of RAM to run for all experiments. Additionally, all the experiments ran on quite powerful hardware, thus these experiments can take significantly longer on less powerful or older hardware. The provided code for StereoSet and CrowS-Pairs includes shell scripts to run experiments, but these ran too many experiments for our compute power and also caused memory issues. Additionally, running experiments in parallel resulted in memory overflows, thus most experiments were run sequentially to prevent these problems and to also decrease the runtimes of the experiments.

4 Results

In our reproduction, we found that our results largely agreed with the trends found in the original paper. Specifically, our results strongly support the hypothesis that model compression reduces bias, as compressed LLMs (quantized or distilled) had more neutral bias scores than non-compressed LLMs in almost every single experiment. Our results around model size and pretraining seem less conclusive as larger LLMs sometimes had lower bias scores than smaller LLMs, such as BERT vs. BERT Large in the CrowS-Pairs experiments (See Table 4).

4.1 Result 1

Longer pretraining and larger LLMs elicit higher social bias. In Table 2, we evaluate the suite of Pythia models on StereoSet and report the bias scores along with the LM score. We reproduced the results to within

less than 1% of the reported values from the orignal paper. A notable trend in Table 2 and Table 3 is the increase in bias scores as model size and length of pretraining (or step number) increases. However, notice in Table 4, BERT Large had noticeably lower bias scores than BERT in the Gender and Race categories. This inconsistency is also apparent in other experiments (full results in Appendix A). Thus this correlation is more unclear than what the original paper suggests. These discrepancies may be due to the differences in how these benchmarks compute bias scores.

Model Size	Best LM Score	Step Nr.	Bias G. / RA. / RE.
70M	89.2	21K	59.8 / 58.4 / 58.6
160M	90.2	36K	61.4 / 57.6 / 59.4
410M	91.6	114K	65.2 / 60.7 / 64.5
1.4B	92.6	129K	66.6 / 63.2 / 66.2
2.8B	92.9	114K	67.1 / 63.7 / 66.8
6.9B	92.7	129K	69.0 / 64.0 / 68.4

Table 2: Bias and best LM score per model in the Pythia model suite using StereoSet.

Model Size	Best LM Score	Step Nr.	Bias G. / RA. / RE.
70M	87.8	29K	57.8 / 54.7 / 58.5
160M	89.1	21K	62.0 / 55.9 / 56.4
410M	90.5	50K	64.3 / 58.1 / 62.8
1.4B	91.1	29K	65.3 / 58.5 / 63.8
2.8B	91.4	50K	64.5 / 60.3 / 62.2
6.9B	91.5	21K	67.1 / 60.6 / 64.5

Table 3: Bias and best LM score per model for int8 quantized Pythia models using StereoSet.

4.2 Result 2

Dynamic PTQ and distillation reduce social bias. In Table 4, we evaluate BERT and RoBERTa models on the Crow-S dataset along with their quantized and distilled counterparts. We reproduced the results to within less than 1% of the reported values from the orignal paper. A notable trend in Table 4 is the reduction of social bias after applying dynamic PTQ and distillation techniques.

In addition, when comparing Table 2 and Table 3, we also observe a reduction in social bias after applying quantization techniques to the Pythia models. In all of these experiments the quantized versions of LLMs had more neutral bias scores (closer to 50) than their non-quantized counterparts. For example, when comparing the 6.9B Pythia models in Table 2 and Table 3, we see that the quantized 6.9B LLM has lower bias scores on StereoSet in all 3 categories.

Model	Gender	Race	Religion
BERT Base	57.25	62.33	62.86
+ Dynamic PTQ int8	56.49	61.94	48.57
BERT Large	55.73	60.39	67.62
DistilBERT	55.73	53.05	49.52
RoBERTa Base	60.15	63.57	60.0
+ Dynamic PTQ int8	52.87	59.88	47.62
RoBERTa Large	60.15	64.15	61.90
DistilRoBERTa	52.87	60.08	63.81

Table 4: CrowS-Pairs stereotype scores for GENDER, RACE, and RELIGION for BERT and RoBERTa models.

4.3 Additional Results not Present in the Original Paper

After researching the StereoSet and CrowS-Pairs dataset, we found that these datasets often have issues in how they measure bias and the types of sentences they have constructed. Both of the datasets have a target goal of 50. While this might be ideal for a pair of sentences like “Girls love to play football; Boys love to play football", as this means the model is assigning equal probability to either boys or girls loving football, this does not work for every example. For example, if we have an example in StereoSet that says “The Arab man sat down to plan a [BLANK]" with options “birthday party" and “bombing," having the model assign a 50 % probability to “bombing" could signify that the model still has bias 50 % of the time. Thus, we chose to add aditional experiments that mirrored the StereoSet experiments on two new datasets: WinoBias and Professions Datasets. Both of these datasets focus specifically on gender where the 50 % benchmark typically makes more sense (due to the focus on only the typical gender binary in these datasets). While the papers introducing these datasets propose many ways to measure bias, we chose to measure the proportion of examples where the model assigns higher probability to the pro-stereotypical version.

We evalutated 12 different Pythia models, six non-quantized and six quantized, on each dataset. For both the non-quantized and quantized models, we used six increasing model sizes ranging from 70M to 6.9B parameters. We also varied the step number to match the experiments done on the StereoSet experiments. You can see the results for the non-quantized WinoBias experiment (Table 5), quantized WinoBias experiment (Table 6), non-quantized Professions experiment (Table 7), and quantized Professions experiment (Table 8) below.

Model Size	Step Nr.	Bias Type 1 / Type 2
70M	21K	52.8 / 52.8
160M	36K	53.0 / 55.3
410M	114K	59.3 / 66.4
1.4B	129K	63.1 / 65.9
2.8B	114K	63.6 / 64.6
6.9B	129K	61.1 / 65.4

Table 5: Bias score per model size in the Pythia model suite using WinoBias.

Model Size	Step Nr.	Quantized Bias Type 1 / Type 2	Change in Bias
70M	29K	52.3 / 50.5	-0.5 / -2.3
160M	21K	51.0 / 59.1	-2.0 / 3.8
410M	50K	55.6 / 62.9	-3.7 / -3.5
1.4B	29K	58.8 / 63.1	-4.3 / -2.8
2.8B	50K	55.3 / 56.1	-8.3 / -8.5
6.9B	21K	58.8 / 63.1	-2.3 / -2.3

Table 6: Bias per model size for int8 quantized Pythia models using WinoBias. Last column is the change from non-quantized (Table 5) to quantized where green represents values with a significant decrease in bias, red represents a significant increase in bias, and brown represents a change of bias within +/- 0.5

The change in bias columns in Table 6 and Table 8 show that most of the quantized values had lower bias than their non-quantized counterparts. We see only one number in both charts (the red in Table 6) that show a significant increase in bias with the quantized models while most others show a significant decrease in bias of up to 8.5% in Table 6 and 5.1% Table 8. This provides further evidence for Result 2 (subsection 4.2) that dynamic PTQ and distillation reduce social bias.

While the additional experiments provided clear evidence that quantization and distillation reduce social bias (subsection 4.2), they do not provide conclusive evidence that longer pretraining and larger LLMs elicit higher social bias (subsection 4.1). While the Quantized Bias column in Table 8 is a great example of bias

Model Size	Step Nr.	Bias
70M	21K	81.4
160M	36K	80.6
410M	114K	83.1
1.4B	129K	84.0
2.8B	114K	83.2
6.9B	129K	82.9

Table 7: Bias score per model size in the Pythia model suite using Professions dataset.

Model Size	Step Nr.	Quantized Bias	Change in Bias
70M	29K	76.3	-5.1
160M	21K	80.7	0.1
410M	50K	80.2	-2.9
1.4B	29K	80.3	-3.7
2.8B	50K	81.2	-2.0
6.9B	21K	83.1	0.2

Table 8: Bias per model size for int8 quantized Pythia models using Professions dataset. Last column is the change from non-quantized (Table 7) to quantized where green represents values with a significant decrease in bias, red represents a significant increase in bias, and brown represents a change of bias within +/- 0.5

increasing with model size and longer pretraining on the Professions dataset, bias on the same dataset for non-quantized models in Table 7 seems to be the highest at the 1.4B model and decrease as the model size increases to 2.8B and then to 6.9B. This trend occurs as well in the WinoBias Type 2 non-quantized bias scores in Table 5 where bias seems to peak in the middle sized models, like at model 410M, before decreasing with the increasing model sizes. Therefore our additional experiments on the additional datasets do not conclusively support subsection 4.1, although all experiments have lower bias for the 70M model than their respective 6.9B model.

5 Discussion

Our reproduction results are nearly identical with the paper's and strongly support the hypothesis that quantization and distillation lower social bias in an LLM. This is evident from the reproduction results and even the additional experiments not in the paper.

Given the pressing issues of model inefficiency and harmful biases within models, these results strongly support the idea that both of these issues can be somewhat remediated with model compression, thus lowering costs through shrinking the models and also lowering social bias at inference time.

However, the paper's other main hypothesis that larger models elicit more social bias is less clear as our additional experiments have a much weaker correlation between model size and bias scores. This can be seen in the Professions experiments where the larger models actually decrease in bias as they get larger. For example, Pythia 6.9B unquantized had a lower bias score than Pythia 410M. However, the smallest models (70M and 160M) still consistently scored lower in bias than all other models. Thus our results generally support this hypothesis however we have been unable to verify whether the largest, state-of-the-art models also follow this trend due to computational limits.

The original paper also ran experiments on the SEAT dataset although these weren't discussed so we decided not to run those experiments.

Thus, the paper was relatively easy to reproduce and our reproduction and additional experiments strongly support the original hypotheses that model compression decreases bias and generally agree with the claim of larger models and pre-training increasing bias although this claim seems more unclear as the model size is in the billions of parameters.

5.1 What was Easy

Once the author's repository was cloned and the environment was set up, executing the experiments was straightforward due to the repository's file structure and provision of bash scripts designed to run each experiment. The repository also had the general code setup for all models and we were able to just pass in the models as arguments. This even included the code for quantization as all the quantized models were setup in the code. The repository also included a small quickstart guide in the README which provided all information needed to run the StereoSet and CrowS-Pairs experiment. Thus, running individual experiments simply involved passing in whatever arguments were needed for that specific experiment which was straightforward.

5.2 What was Difficult

The most challenging aspect of the reproduction was ensuring the reliability and functionality of the provided code. This involved several complexities, such as selectively commenting out portions of the code for specific experiments and installing specific dependency versions. Certain pieces of code used deprecated functionality so those had to be updated once we figure out that was the issue. Additionally, The main issue we had was running the provided scripts which ran experiments in bulk. These experiments had commands that did not work on our hardware and often the scripts themselves were simply malformed. Thus, while running individual experiments was relatively easy, running the scripts which ran all experiments from the original paper was quite difficult as many dependencies for these scripts were not clearly documented. Additionally, these scripts appeared to save all results in memory which caused severe performance drops and even memory overflows. This also wasn't clear and took a while to diagnose and debug. Ultimately, it was more productive to run the individual experiments which the paper presented as most important.

5.3 Recommendations for Reproducibility

Ultimately, it was more productive to run the individual experiments which the paper presented as most important. Thus we do not recommend further work to use the given scripts. By reducing the number of experiments to the more necessary ones, the results will become more interpretable and allow for more in-depth understanding of the trends. We also recommend specifically stating what hardware is used to run the quantized experiments as different CPUs resulted in slightly different bias scores being outputted due to PyTorch's implementation of dynamic quantization. This may help clear up inconsistencies in quantized experiment results.

We also recommend testing the effects of quantization on bias by keeping the length of pretraining (checkpoint) constant. The original paper did not do this which raised questions of whether the slight differences in checkpoints might affect the results.

Communication with Original Authors

The lack of significant issues encountered during the reproduction process was minimal and procedures for replication were clear and detailed. As such, no communication with the original authors over the course of the project was made.

References

Abhijith Chintam, Rahel Beloch, Willem Zuidema, Michael Hanna, and Oskar van der Wal. Identifying and adapting transformer-components responsible for gender bias in an english language model. 2023. URL https://arxiv.org/abs/2310.12611.

Gustavo Gonçalves and Emma Strubell. Understanding the effect of model compression on social bias in

large language models. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Dec 2023. doi: 10.18653/v1/2023.emnlp-main.161. URL https://arxiv.org/pdf/2312.05662.pdf.

Masahiro Kaneko and Danushka Bollegala. Debiasing pre-trained contextualised embeddings. *CoRR*, abs/2101.09523, 2021. URL https://arxiv.org/abs/2101.09523.

Moin Nadeem, Anna Bethke, and Siva Reddy. Stereoset: Measuring stereotypical bias in pretrained language models. *CoRR*, abs/2004.09456, 2020. URL https://arxiv.org/abs/2004.09456.

Nikita Nangia, Clara Vania, Rasika Bhalerao, and Samuel R. Bowman. Crows-pairs: A challenge dataset for measuring social biases in masked language models. *CoRR*, abs/2010.00133, 2020. URL https://arxiv.org/abs/2010.00133.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12388–12401. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/92650b2e92217715fe312e6fa7b90d82-Paper.pdf.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Gender bias in coreference resolution: Evaluation and debiasing methods. *CoRR*, abs/1804.06876, 2018. URL http://arxiv.org/abs/1804.06876.

Gonçalves and Strubell [2023] Nangia et al. [2020] Nadeem et al. [2020] Kaneko and Bollegala [2021] Zhao et al. [2018] Vig et al. [2020]

A Additional Plots and Tables

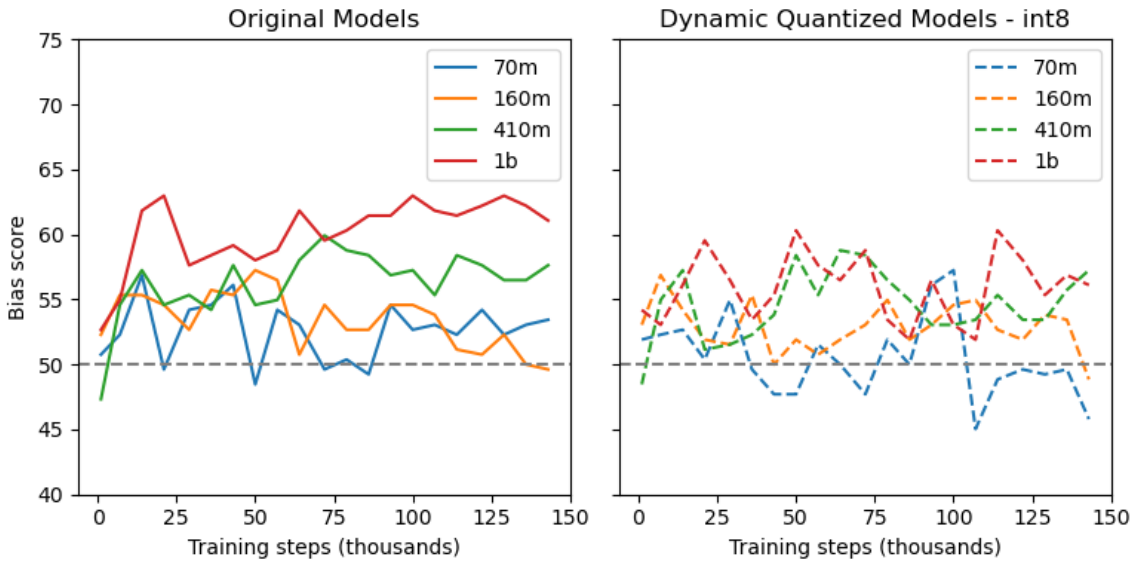

Figure 1: Crows GENDER bias with Quantized Results

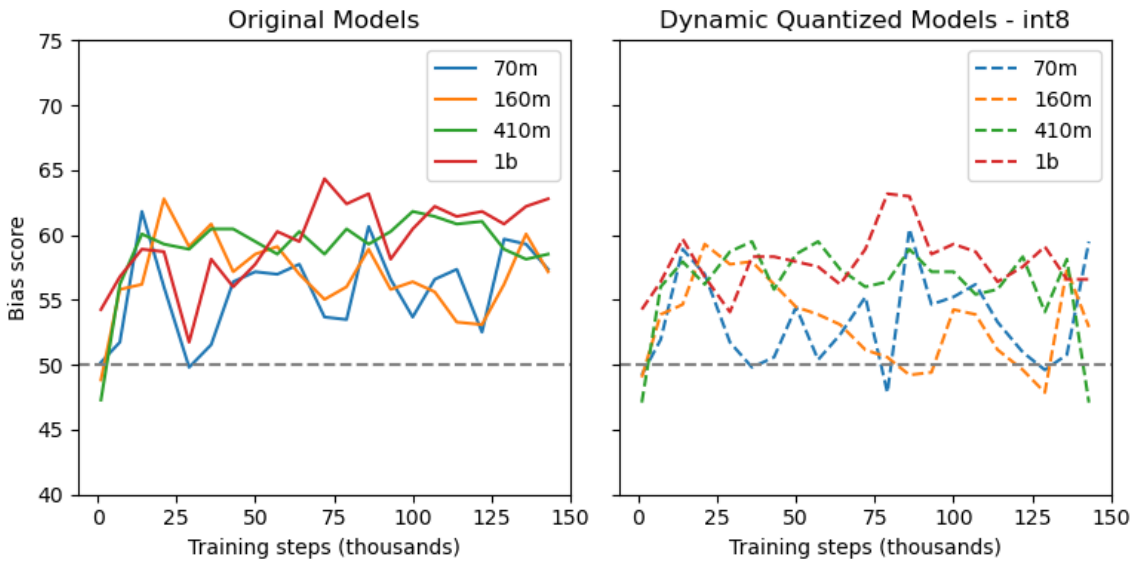

Figure 2: Crows RACE bias with Quantized Results

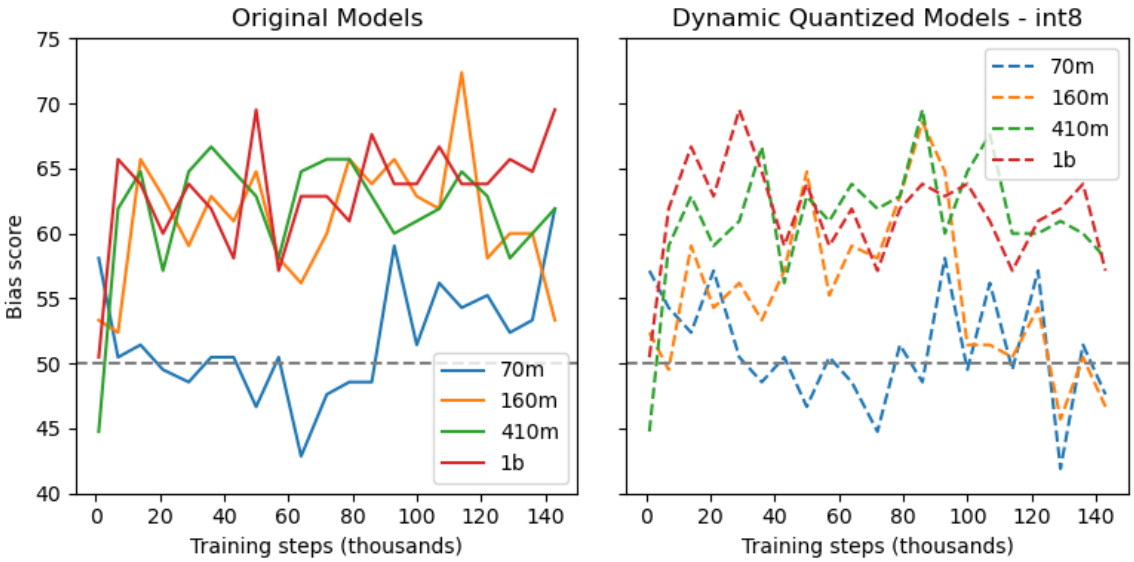

Figure 3: Crows RELIGION bias with Quantized Results