Breaking Bias: An Approach to Quantify and Reduce Bias of LLMs In Multiple Choice Answering

Anvitha Kannapu akannapu@umass.edu
Roshini Pulishetty rpulishetty@umass.edu
Sphurti Agarwal sphurtiagarw@umass.edu
Bhanusree Ponnam bponnam@umass.edu
Ruchira Sharma rssharma@umass.edu

1 Problem statement

We explore the behavior of Large Language Models when answering questions with multiple choice. While LLMs outperform humans on creative tasks, summarization and memorization, and perform reasonably well on reasoning and question answering, their performance is not on par in constrained generation. When posed with multiple choices and constrained to choose an answer from them, LLMs tend to select choices against a particular option ID often, rather than choosing the option ID with the correct answer. This we term as the **"token bias"** in this document. Similarly, their tendency to choose options at a particular position frequently over other options is termed as **"positional bias"**. We examine whether a few popular LLMs exhibit these 2 biases when encountered with a multiple choice, and analyze and quantify the degree of their bias. Further, we attempt to mitigate this selection bias using various approaches, thereby understanding the source and the nature of the bias.

2 What you proposed vs. what you accomplished

We proposed to analyse the bias, followed by reducing this bias using prompt tuning and CoT across different LLMs.

- *While our problem statement remains consistent with what we proposed, we diverted from the approaches we mentioned. We proposed to develop a AutoCoT adapter to tackle this problem and use this adapter aginst a few LLMs to verify if it reduces the bias. However, we currently employed a supervised fine-tuning with LoRA.*
- *Due to compute issues, we couldn't load models larger than 7B ones, hence couldn't get run experiments on them.*
- *We proposed to only analyze the bias of a few LLMs. But in our experiments, we could segregate it into positional and token bias, quantify them and draw parallels systematically. We evaluated this bias using state-of-the-art metrics mentioned in the recent papers. Further, we could do this analysis for both in-domain and out-of-domain data for the above mentioned approaches.*
- *We proposed to use 2 datasets, MMLU and SummEdits, but we used MMLU since it has 57 subjects, which we could divide into training sets of 9963 samples, validation sets of around 1100 samples, and in-domain and out-of-domain test sets of 7900 and 2800 samples.*

3 Introduction

Question answering remains to be the fundamental aspect of language understanding and learning, with Multiple Choice Answering being one of the widely adopted formats. As cited by some of the recent studies (16), (9), existing LLMs exhibit biases in their selection of answers, often favoring specific option(s). This bias not only affects their performance but also undermines the reliability of their answers, limiting their practical applicability. Hence, it is non-trivial to quantify the bias of LLMs towards each option and each position.

The objective of this project is to address this critical issue in the Question Answering domain, by initially gauging the bias and then, employing

different strategies. Analyzing the LLMs performance with these strategies assists us in understanding the nature and origin of the bias. This study draws attention towards the bias and robustness of the LLMs.

4 Related work

Evaluation of Bias in LLMs: With the growing prominence of LLMs in addressing NLP tasks, significant attention has been devoted to examining the robustness of these models. Studies by (14), (3), and (9) show LLMs are sensitive to input manipulations like prompt wording, demonstration order, and adversarial attacks, highlighting potential vulnerabilities in individual model outputs. Alignment research, exemplified by (6), (17), (13), and (12), explores how to deliberately misalign LLMs through techniques like adversarial prompts. These studies reveal limitations in current alignment methods, suggesting that specific prompts can manipulate LLM behavior.

(5) investigated the LLMs capabilities to understand the task at hand in the question answering scenario, and to respond with the corresponding option for the answer they predicted. Performing these explorations on small-sized models up to 9B parameters, they concluded that surprisingly only less than half of the models understood the task and very few could match their answer to the choices given. Further, hardly 4 models exhibited negligible bias.

In their work, (11) explore the robustness of textual answers when compared to the first token probability approaches in evaluating large language models. They found that as the mismatch rate between textual answers and first token answers increases, the robustness of textual answers increases. So, when the questions in the dataset were modified and the models were tested, the test based answer evaluation was proven to be more robust than just the single token prediction!

In this project, we study the bias of LLMs, during MCQ answering, when elicited with different options due to their tokens/order, unlike societal bias. (7) investigated LLMs' robustness in MCQs, finding significant performance drops (13-75%) when answer options are reordered. This suggests a positional bias, particularly when LLMs are uncertain between top choices. By analyzing top-choice patterns, the authors recommend strategies to amplify or mitigate this bias, leading to improved LLM performance.

Contrary to the common view of LLMs favoring options presented at specific *ordering positions* (like first or last) ((10) (7)), (15) pinpoint one more salient intrinsic cause of selection bias as the model's *token bias* when predicting answers from the option IDs given the standard MCQ prompt, where the model a priori assigns more probabilistic mass to specific ID tokens (e.g., A/B/C/D).

Attempts at Debiasing: (15) introduced a debiasing technique (PriDe) by first estimating the prior bias and then, debiasing the system during inference time by applying this prior. Though it is proven to be a light-weight effective technique to retrieve answers with reduced partiality, this study fixes the symptoms of the problem, rather than removing the intrinsic cause of this nature in the LLMs. Unlike this study, our work fine-tunes the existing LLMs, allowing it to learn appropriate cues to mitigate this intrinsic bias. Additionally, while this work shows that the imbalance of recalls is correlated with delta in accuracy, we demonstrate that imbalance of recalls as a metric is correlated with distribution of likelihood of incorrect answers.

5 Dataset

We used the MMLU (Massive Multitask Language Understanding) benchmark (2) for training, validation and testing. It encompasses 57 subjects spanning STEM, the humanities, the social sciences, and beyond, and spans difficulty levels from elementary to advanced professional, evaluating both worldly knowledge and problem-solving prowess. All the questions homogeneously contained 4 options. With a broad domain of subjects, we can effectively utilize different splits for training, validation and testing, while evaluating performance on both in-domain and out-of-domain test sets.

5.1 Data Pre-processing

The MMLU dataset (2) was readily available to us through HuggingFace with the ground truth answers. However, we did a substantial amount of

pre-processing to obtain data in the format we required. Each table represented questions from a different subject. Firstly, we left a few datasets (8 datasets) only for testing performance on out-of-domain samples. They are "abstract algebra", "college medicine", "conceptual physics", "high school mathematics", "philosophy", "prehistory" and "professional law", where some of the subjects like "abstract algebra" closely with the fine tuned tables while "philosophy" and "prehistory" don't match with any of the fine-tuning subjects. Then, we split the remaining 49 datasets into training data, validation data and test data in 70%, 10%, and 20% respectively. These splits of the train-val-test contain random samples from each table, hence they have unbalanced questions with different options as the correct answer.

Each of the in-domain and out-of-domain test sets are first balanced with the number of correct answers for each option. That is, for a total of N samples chosen from each table, there are exactly $N/4$ questions with each option (A/B/C/D) as the correct answer. We permuted the test sets by placing the correct answer in each option and created a testset, say TS_1. Likewise, we permuted by only varying the positions, without altering the options attached and stored it as a different testset, say TS_2. This assists us in evaluating token and positional bias separately.

5.2 Data annotation

Our project did not require any annotation of datasets since the dataset we used contains the questions on world knowledge in different domains. Since there is no ambiguity and each question has exactly one correct answer among the choices, the answers are provided within the dataset and no human annotations were required.

6 Evaluation Metrics

Our evaluation protocol primarily considers the logit probabilities of the generated response and doesn't rely heavily on counting the labels. We chose the below 4 metrics to depict the existence of bias with respect to either token or position, quantify them with 95% confidence and then, correlate with the other metrics and compare across the range of domains and models.

1. Top-2 and top-3 bins: This metric fundamentally measures the percentage of the questions the correct answer falls in the top-2 or top-3 buckets if the model has predicted a wrong answer. As mentioned in (7), the tendency of the correct answer being the top second option indicates the model's sensitivity to these orderings of choices.

As we constraint the models to predict only a single token and re-normalize the probabilities across the 4 option tokens only, option with the highest probability is taken as the prediction. Thus, the more probability mass lies in top-2 and top-3 bins, the model is likely getting confused in choosing the correct option over the biased option, assigning the highest likelihood to the biased option. To depict the existence of bias, we use top-2 and top-3 bins.

2. Recall Imbalance: When the correct answers to all the questions are moved to an option token (A/B/C/D), the standard deviation of likelihoods of choosing that option is termed as the "recall imbalance" towards that option. It is taken as a key metric in (15).

Intuitively, when a model is posed a set of questions with the same token as the answer, the variation in its confidence to select that token is recall imbalance. The greater the standard deviation towards an option, the more imbalanced it is and the higher is its bias towards that option. Whereas if these likelihoods are balanced when all answers are moved to a specific option and exhibit less deviation, the LLM shows more confidence in choosing the answer. Thus, recall imbalance can potentially quantify the bias towards each option using TS_1 test set.

$$\text{Recall imbalance}_A = RStd([p_1^A, p_2^A, ..p_N^A])$$

Where $p_1^A, p_2^A, ..p_N^A$ are the logit probabilities of the prediction being 'A' for the N questions. Similarly,

$$\text{Recall imbalance}_B = RStd([p_1^B, p_2^B, ..p_N^B])$$
$$\text{Recall imbalance}_C = RStd([p_1^C, p_2^C, ..p_N^C])$$
$$\text{Recall imbalance}_D = RStd([p_1^D, p_2^D, ..p_N^D])$$

Similarly, we extend this quantitative metric for each position using the TS_2 testset.

Figure 1: Permutation of questions for testing by varying option

Question: Aquinas claims that the ultimate perfection of operation is	Question: Aquinas claims that the ultimate perfection of operation is	Question: Aquinas claims that the ultimate perfection of operation is	Question: Aquinas claims that the ultimate perfection of operation is
Options:	Options:	Options:	Options:
A. delight	B. peace	C. pleasure	D. Godliness.
B. peace	A. delight	B. peace	B. peace
C. pleasure	C. pleasure	A. delight	C. pleasure
D. Godliness.	D. Godliness.	D. Godliness.	A. delight

Figure 2: Permutation of questions for testing by varying position

Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:
Options:	Options:	Options:	Options:
A. testosterone.	B. testosterone.	C. testosterone.	D. testosterone.
B. cortisol.	A. cortisol.	B. cortisol.	B. cortisol.
C. progesterone.	C. progesterone.	A. progesterone.	C. progesterone.
D. aldosterone.	D. aldosterone.	D. aldosterone.	A. aldosterone.

Our claim is that the recall imbalance serves as the robust measure of the bias, by quantifying the imbalance across different domains and estimating the confidence intervals.

$$\text{Bias} \sim \text{Recall Imbalance}$$

3. Incorrect Likelihoods: The error likelihood is a counting-based metric, which indicates the proportion of incorrectly answered questions where the model predicts an option against a specific token as the correct answer.

$$\text{IncorrectLikelihood}_A = \frac{\mathcal{C}('A')}{T} \times 100$$
$$\text{IncorrectLikelihood}_B = \frac{\mathcal{C}('B')}{T} \times 100$$
$$\text{IncorrectLikelihood}_C = \frac{\mathcal{C}('C')}{T} \times 100$$
$$\text{IncorrectLikelihood}_D = \frac{\mathcal{C}('D')}{T} \times 100$$

where $\mathcal{C}('A')$ = Count of model incorrectly predicting option 'A' as answer and T = Total incorrectly predicted questions.

For instance, when a model is prompted with a question supported by the options, if the model selects a wrong answer, our interest lies in which option it has selected and the corresponding token. This facilitates us to assess its tendency to choose a particular token as the answer when it is wrong as well as its behavior for false positive cases. Intuitively, if a model has intrinsic bias towards a specific token, it tends to select this option rather than using its pre-trained knowledge to find the correct answer. Though it is a count-based metric, we employ it in our evaluation framework for drawing parallels with the recall imbalance scores.

Similar to the above metric, we aggregate the model's performance across different domains and tasks to determine the point estimate and the 95% confidence interval of error likelihood. Likewise, we gauge the error likelihood by varying position as well.

4. PPA scores: While accuracy refers to the proportion of the correct answers generated by the model, we determine PPA scores that measure the proportion of the time the correlated answer is generated. As introduced by (8), to measure the PPA score of the model, we present each question with different orderings of options and record the answer with the highest probability. Then, we take the option chosen maximum number of times across different orderings as the most correlated answer.

This is an important metric since bias is not about the correctness, rather the model's tendency to change the answer based on the option or position it is assigned with. Thus, we chose PPA score in our evaluation protocol, instead of accuracy.

When presented with questions by permuting

option tokens,

$$PPA_A = \frac{\mathcal{P}('A')}{\mathcal{T}('A')} \times 100$$
$$PPA_B = \frac{\mathcal{P}('B')}{\mathcal{T}('B')} \times 100$$
$$PPA_C = \frac{\mathcal{P}('C')}{\mathcal{T}('C')} \times 100$$
$$PPA_D = \frac{\mathcal{P}('D')}{\mathcal{T}('D')} \times 100$$

where $\mathcal{P}('A')$ = Count of correctly choosing option 'A' for all questions with answer 'A' and $\mathcal{T}('A')$ = Total questions with 'A' as the correct answer.

Note that this is the precision counterpart of recall imbalance. While recall imbalance serves as the primary metric to quantify bias, PPA score helps us compare the behavior before and after applying an approach.

7 Baselines

The main baselines in our experiments are the pre-trained models without any further training or modifications, for we benchmark our approaches against the existing bias in the models. Since our efforts are focused towards reducing this bias, we quantify the existing bias. We did extensive analysis and experiments on 4 models primarily - Gemma-2B instruct-tuned, Gemma-7B instruct-tuned, Llama 1 7B, and Llama 2 7B, 2 models from 2 families of models - Gemma and Llama. Further, the Gemma models we chose are instruct-tuned and Llama models are non-instruct-tuned. We chose these models because they are small-sized. Our hypothesis is that smaller models tend to exhibit more bias compared to larger models. Also, due to compute issues, we couldn't load models bigger than 7B. Apart from these models, we also performed the initial analysis on other models and measured their bias for cross-domain and cross-family comparisons.

As mentioned in section 3.1, leaving out 8 tables exclusively for testing, the rest of the tables are divided into train-val-test splits with 70%/10%/20% respectively. Hence, there is no overlap between our train, validation and test sets.

Inference:

For the initial inference from the models, there were no hyper-parameters to be tuned. We directly fed the pre-processed permuted testsets TS_1 and TS_2 to the models and recorded their predictions, along with the likelihoods of option tokens - A, B, C, D.

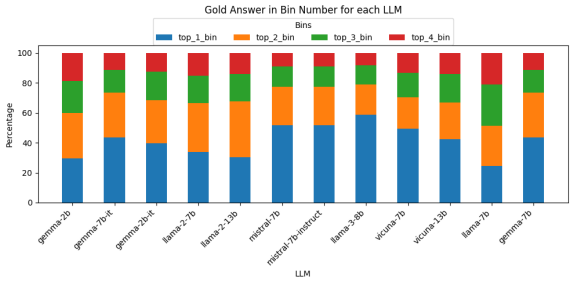
Key Observations:

Our analysis using testset TS_1 demonstrates the behavior of LLMs by varying only the option/token associated with the options. Likewise, analyzing LLM's performance on TS_2 test set shows the key role position plays in the model's prediction.

7.0.1 By Varying Option

From our initial analysis by permuting only the option tokens while keeping their positions unchanged, we could observe from Figure [3] that all the below LLMs are uncertain about the prediction between the top-2 choices. That is, there is a higher probability mass for the top-2 bin compared to top-3 or top-4 bin. For instance, Gemma 2B IT predicts the correct answer with 39.49% accuracy and the correct answer falls in the top second choice for 29.02%. Similarly, Gemma 7b IT has 43.46% mean accuracy and 56.54% of the times it incorrectly predicts, the correct answer is just its next choice. Our conjecture for this high likelihood of correct answer being the top second choice is their sensitivity towards orderings [(7)], thus we hypothesize that their inherent token bias has caused it. This conjecture is supported by larger top-2 bin across cross-domains, as shown in Figure [3].

Figure 3: Top bins across different Large Language Models. The top-1 bin indicates the proportion of the times the correct answer is chosen with highest likelihood, top-2 bin for correct answer being the second highest likelihood.



With this hypothesis, we attempted to measure this option bias. Recall imbalance measures the token bias as the standard deviation of recalls for each option token. If the imbalance is positive and higher, the model is strongly biased towards the option token. If the imbalance is lower and positive or negative, we deduce that the model is unbiased whereas negative and higher imbalance reflects the bias against the option token. In Figure [5], we demonstrate a correlation between the balance of recalls between different option IDs and incorrect likelihood. That is, we draw parallels between the probability-based metric, recall imbalance and count-based metric, incorrect likelihoods. Since incorrect likelihoods represent the tendency of the model to predict a particular option token when its prediction is incorrect and is correlated with recall imbalance, we take the recall imbalance as the primary measure of bias. Aggregated over 49 domains, each model's bias is gauged along with their 99% confidence intervals.

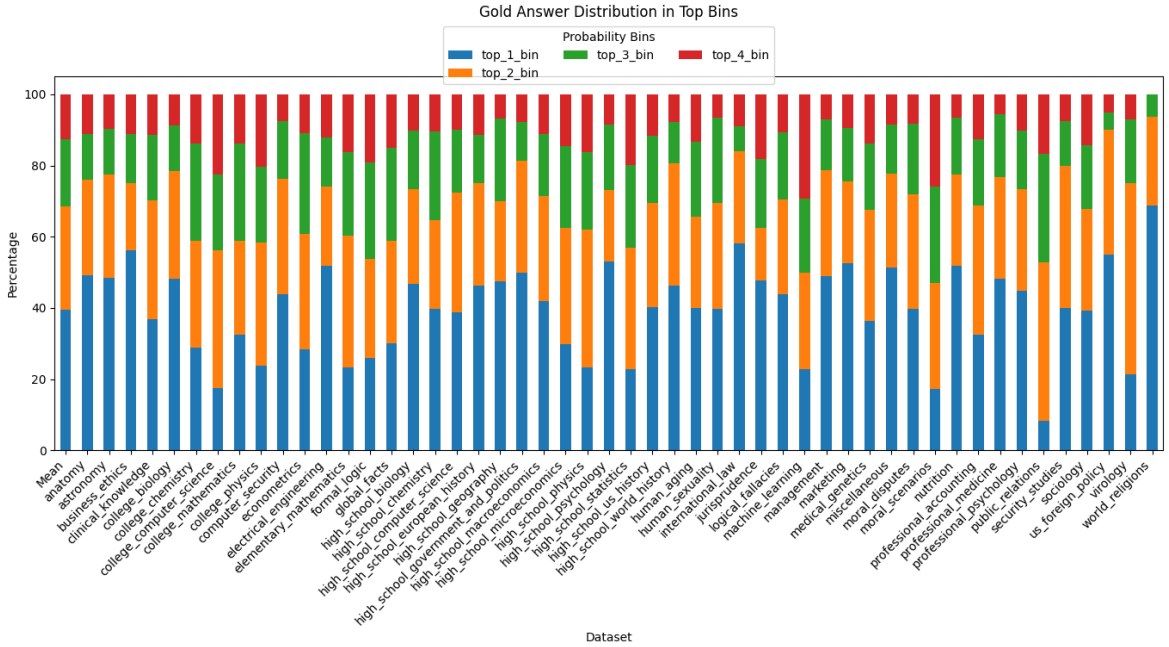
If a model is unbiased, we expect the error likelihoods to be uniform across different options, that is closer to 25% for each option and imbalance scores to be negligible. From Figure [5], strikingly Gemma 2B is likely to choose option 'A' or option 'C' for 70.96% when it is wrong, that is it has higher incorrect likelihood for option 'A' and option 'C', while for option 'D' it is minimal. Despite this testset containing each question permuted and equally distributed among all options, the model shows a wide gap in choosing the answer. In parallel, recall imbalance is negative for option 'D' (-0.60%), indicating the model's bias against this option.

Whereas Gemma 2B Instruct-tuned tends to opt 'A' for 46.02% of the time when incorrect, while the rest of the options 'B', 'C' and 'D' 17.31%, 16.99% and 19.67% respectively. This vividly indicates its tendency to generate token 'A', which could be misleading it for choosing the wrong option. Correspondingly, the recall imbalance is higher for option 'A' for Gemma 2B Instruct-tuned. Similarly, Gemma 7B tends to choose option 'A' for a whopping 87.97% of the time, which is very high compared to the rest, hence has a strong bias of +1.77. While the model is almost unbiased towards C and D with just +0.26 and +0.25, it shows a stronger bias of -2.29 against option B. Similar to Gemma 7B, Gemma 7B Instruct-tuned has strong bias towards option 'A' (+1.77), unbiased towards C (+0.265) and D (+0.25), and strong bias against 'D' (-2.29), supported by their corresponding error likelihoods.

Llama models show a higher tendency to generate 'A' when wrong, hence mostly choosing the option linked to 'A'. Llama 7B has strong bias towards A (+2.08), followed by 'C' (+0.497) and negative bias for 'B' (-1.34) and 'D' (-1.23). Hence, it responds with 'A' for an average of 52.12% of the time, which accounts for slightly more than half, reducing the chances of responding with other options. Llama 2 7B has strong bias towards A (+1.44) while slightly negatively biased towards the rest C (-0.16), B (-0.42) and D (-0.86). So, it chooses A when incorrect for an average 80% of the time, which is a whopping 589% greater chance compared to the second highest option 'C'. The Llama 2 13B too shows stronger bias towards A (+3.38), predicting it for 95% of the time it is incorrect. This drives it to be negatively biased towards the rest, with strong bias against B (-1.538) and D (-1.645). Hence, these 3 Llama models have bias towards 'A' to a different degree. (All these results are submitted along with code in the "mmlu 02" folder.)

Figure [6]: Taking a closer look at Gemma 2B Instruct-tuned for cross domains, the model clearly gives the highest bias towards option 'A' across most of these domains. That is, it demonstrated a bias of (+4.77) towards A in professional law, (+3.7) in philosophy, +8.9 in high school mathematics, and (+7.9) in abstract algebra. Hence, has stronger tendency to choose option 'A' of 56% in professional law, 47.17% in philosophy, 55.55% in high school mathematics and 33.76% in abstract algebra, aggregating to 46.02% of the time model choosing option 'A' when gone wrong. Please note that there are a few fluctuations in some domains due to the limited number of questions (around 400) in each domain. This indicates that the model shows similar behavior across different domains, hence the tendency to

Figure 4: Top bins of Gemma 2B IT predictions across domains, indicating larger top-2 bin for model prediction consistently across different domains.



incorrectly predicting an option token over others is decoupled from the domain of the question. (For the rest of the models, we added our metrics plots in the appendix.)

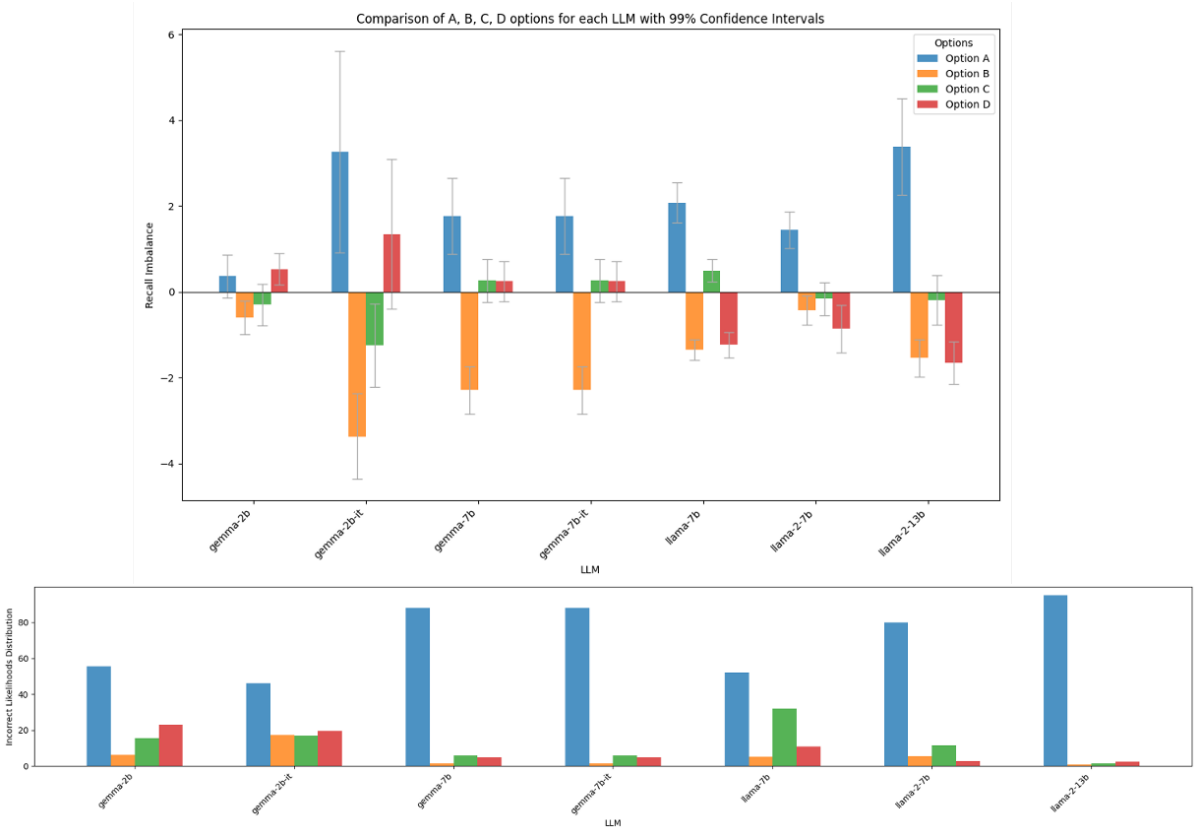
Figure [7] PPA scores is an indicative measure of the proportion of the times the predicted answer correlates with the frequently chosen answer, across different sortings of the options. That is, for each option, it shows the answer chosen aligns with the answer it chose across different orderings of options. If the score is higher for an option, it shows the model is not driven by bias when choosing this option with certainty. However, if the score is the least, it clearly indicates the bias towards a different option led it to lesser scores. Thus, lesser PPA scores convey the bias against the option token, while higher scores are of interest in this study.

Gemma 2B has a lower PPA score of just 27.60% for option 'B', saying that the model doesn't choose the answer it usually chooses, when the answer is associated with 'B'. This bias against B perfectly aligns with the bias (-0.60) we measured using recall imbalance scores. Similarly, Gemma 2B instruct-tuned has least PPA score of 62.46% for option 'B' among other options, and negative bias of (-3.37). Gemma 7B and Gemma 7B instruct tuned have negative bias for B with (-2.28) and (-2.28), thus the least PPA scores of 23.46% and 23.46% respectively.

The Llama family of models too showed correlation between the least PPA scored option and the bias against this option, measured through recall imbalance. Llama 7b, Llama 2 7b and Llama 2 13b show lesser PPA scores for B of 36.84%, 42.4% and 11.41%, which is clearly far less than half, the model doesn't choose the correlated answer. Likewise, their bias toward option B is -1.34, -0.424 and -1.538. Thus, we observe from these results that the stronger bias towards an option, makes the model negatively biased against a different option.

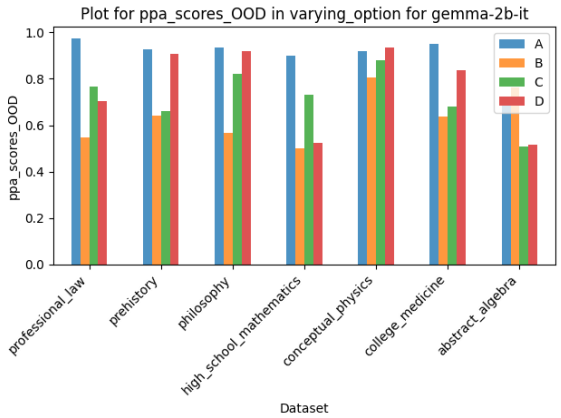
Strikingly, the PPA scores also follow a similar pattern across cross-domain for each LLM, as can be seen in the below plot. Here, when Gemma 2B instruct-tuned is posed with questions of different subjects, its prediction of 'A' aligns with the correlated answer with at least 80% certainty, followed by option 'D', while it's prediction of 'B' aligns with the correlated answer with only about 54.88% in professional law, 64.04% in prehistory and 0.5% in high-school mathematics, aggregat-

Figure 5: Recall imbalance vs the incorrect likelihoods distribution for Gemma and Llama family models (Varying Option)



ing up to 62.4% of PPA score for option B.

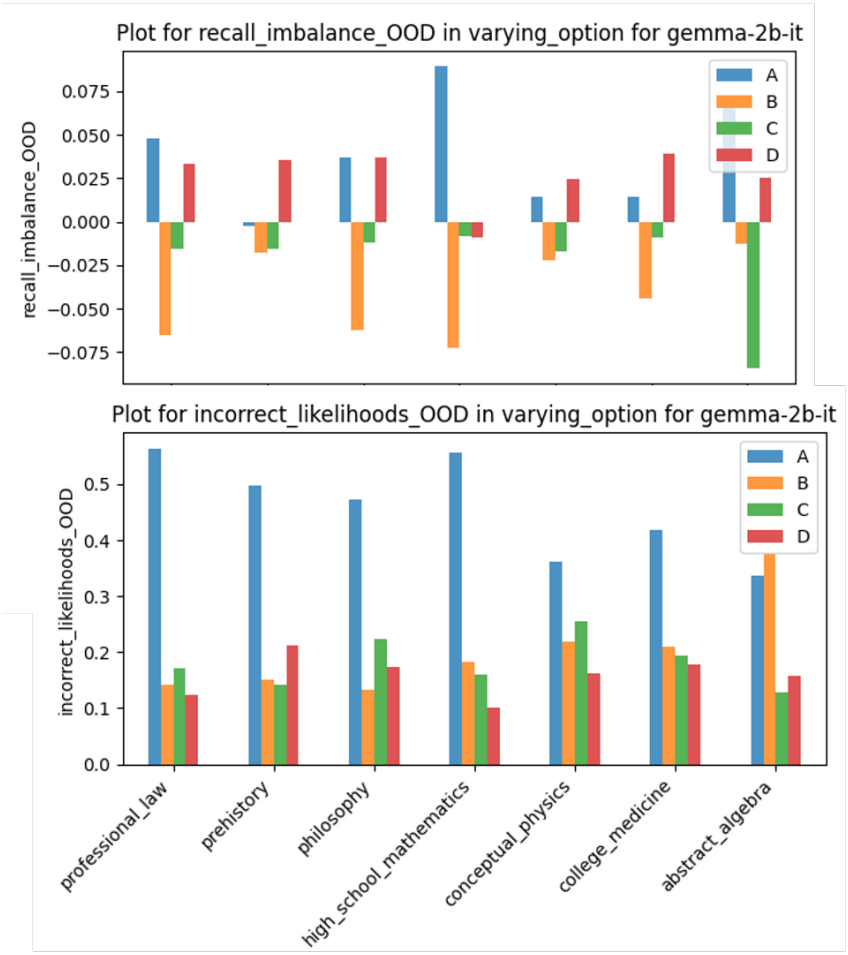
Figure 8



7.0.2 By Varying Position

*Figure [9]*In this scenario, we test the hypothesis that the models also exhibit positional bias, which is different from the token bias, discussed above. Using testset TS_2 for inference, we could observe from the below graph that each of the models have the larger proportion of the time correct answer falling as the top second choice. That is, the models could be uncertain in predicting the answer among top-2 choices, when posed questions with different order of options while the token associated with the option remains steady. Gemma 2B has 30.29% accuracy, while the answer is top second choice for 29.55%, which is a significantly larger portion of the probabilistic mass. Likewise, Gemma 2B instruct-tuned has 29.67% for the answer falling in top-second choice, which is comparable to the 39.25% portion for accurately choosing the answer. Llama 7B has a higher chunk of the proportion for answer being the top-second choice (25.4%) and top-third choice (27.4%), this is even higher than answer being the first choice (24.13%), drifting our attention towards this model. Llama 2 7B and Llama 2 13B show a larger percentage of choosing the correct answer as the top second choice with 34.13% and 38.38%. This is even higher than the propor-

Figure 6: Recall imbalance the incorrect likelihoods distribution for Gemma-2b-it



tion of answers being the first choice, conveying Llama models significantly show more bias. Thus, with a majority portion of the correct answer being the top second choice, sometimes even surpassing the top first choice, we have suspicion that positional bias could be a major factor in placing the answer as the second choice while drifting towards the biased option.

Figure 10: Gemma 2b IT Top bins

Subject	Top-1 bin	Top-2 bin	Top-3 bin	Top-4 bin
professional_law	31.25	31.25	22.5	15
philosophy	33.75	32	22.5	11.75
high_school_mathematics	23.5	35.25	27	14.25
conceptual_physics	38.5	29	21.75	10.75
college_medicine	39.5	32.25	16	12.25
Aggregate (across 57 subjects)	39.50	29.02	18.94	12.54

A glimpse at Gemma 2B instruct-tuned behavior towards these positional variations conveys that the answer falls in the second choice for a major portion just after top-1 bin in professional law (31.25%), philosophy (32%) and college medicine (32.25%). Sometimes, even surpassing top 1 bin like in high school mathematics (35.25%). We suspect there are small variations of top bins among different domains due to few samples from each domain (around 400 samples). However, when we aggregate over all the 57 subjects, we could see that the model has a higher proportion in the top-2 bin of 29.02%, while the correct answer has the highest likelihood for 39.50%.

Figure [11] The recall imbalance and incorrect likelihoods across various LLMs depict that there is a correlation between them. For instance, Gemma 7B has a stark bias towards options placed in the first position (+1.96) and a little bias against options placed in second position (-0.51), third position (-0.52) and forth position (-0.93). Thus,

Figure 7: PPA scores for Gemma and Llama models (Varying Option)

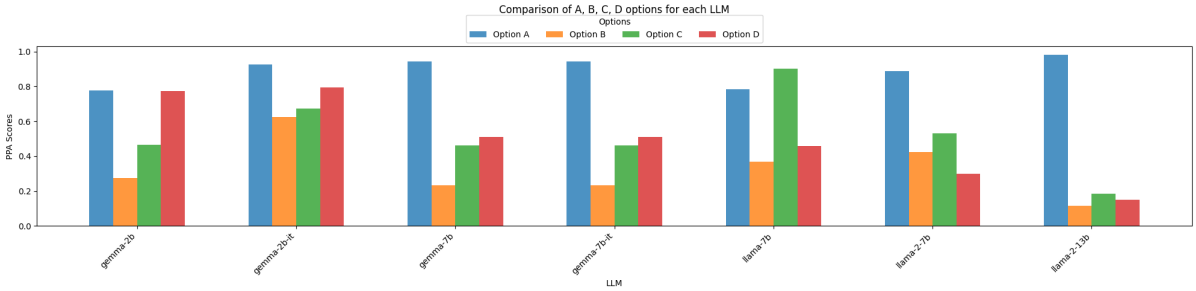
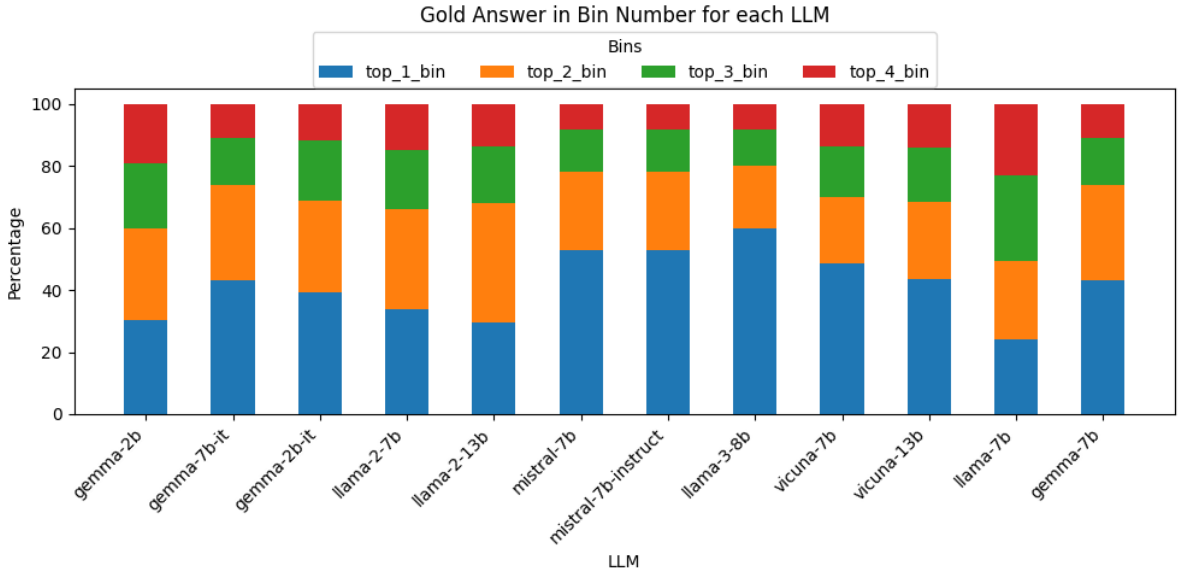


Figure 9

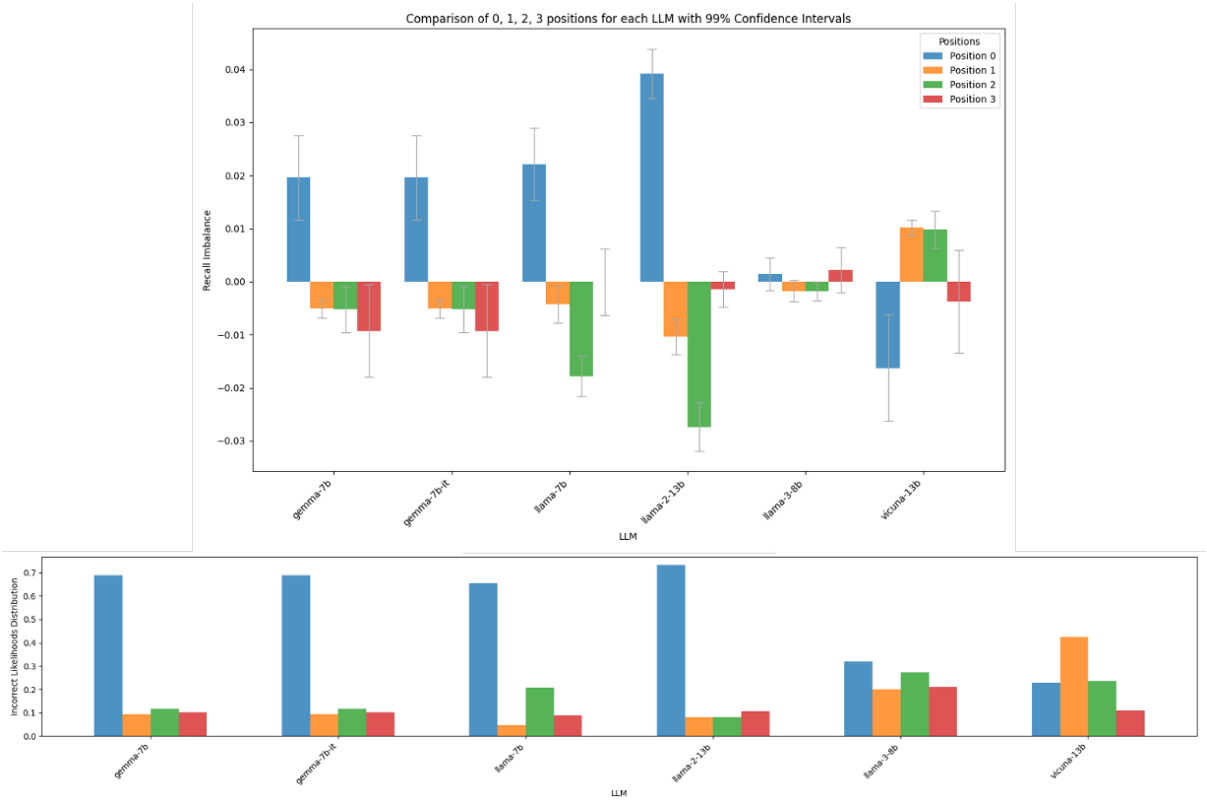


it has the highest likelihood for choosing options in the first place (68.74%) when it is incorrect followed by the other 3 options. Gemma 7B instruct-tuned has similarly high bias towards options placed in the first place (1.96) while biased against the rest of the positions with (-0.51), (-0.52) and (-0.93) respectively in order. This order of biases perfectly matches with that of the incorrect likelihoods. 68.73% of the times model chooses incorrect answer, the answer it predicts is the option placed in the first position, while gradually decreasing to 8.70% for second position, 8.30% for third position and 8.30% for the fourth position.

Llama 7B has the strongest bias for option in the very first place (+2.21), hence it chooses the options in the first position for 63.6% of the time, when it predicts incorrectly. Options in second and third positions are adversely biased, hence placing correct answers in these positions could potentially impact the performance. Likewise, Llama 2 13B has the highest bias for the first position (3.65), hence mapping to options in this position whenever incorrect with 73% probability. While it hardly shows any bias for options in the last position, it shows high bias against second (-1.04) and third positions (-2.74). The graph above depicts that Llama 3 8B shows quite less bias among the positions (1/2/3/4) the answers are placed in of (0.14/-0.178/-0.18/0.22) which is quite less in comparison to other LLMs. Correspondingly, their incorrect likelihoods are balanced across the positions (31.77%/19.9%/27.3%/21.02%).

Vicuna 13B has bias towards options in the second place (+1.017) followed by options in the

Figure 11: Recall imbalance vs the error likelihoods for Gemma and Llama family models (Varying Position)

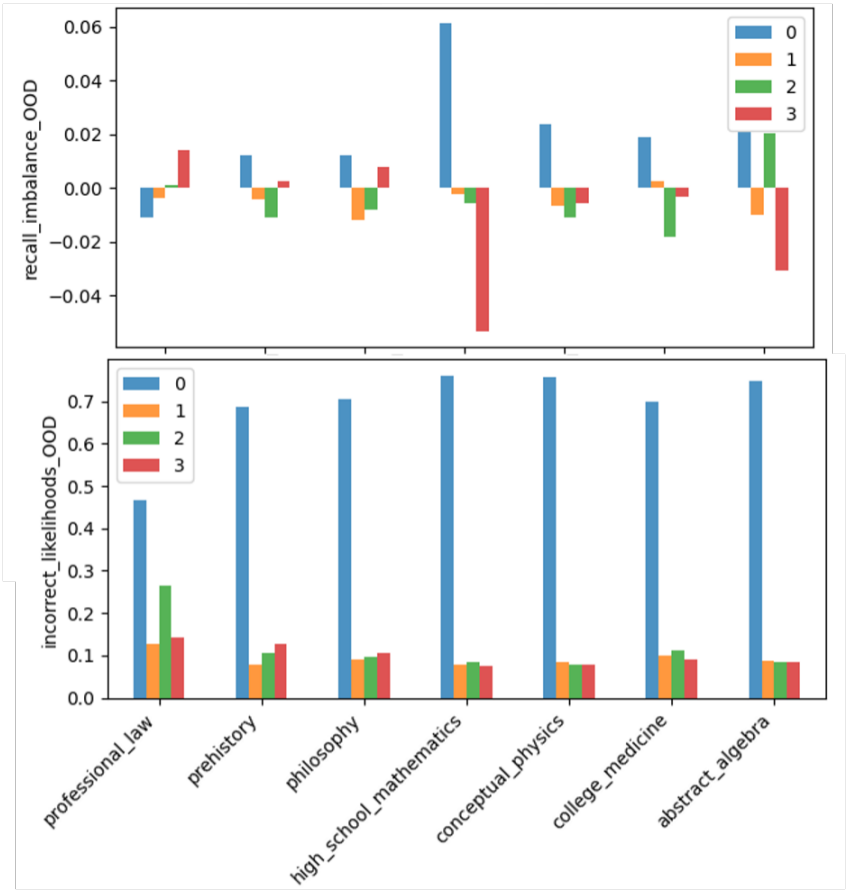


third place (+0.98). It has bias against options in the first place (-1.60) and last position (-0.37).

Figure [12]: Gemma 7B Instruct-tuned shows strong bias towards options placed in first place (+1.96) while it is equally biased against the rest of the positions (-0.51), (-0.52) and (-0.93) respectively. This aggregate bias obtained by taking mean over cross domains aligns with the recall imbalance for each domain. That is, it gives recall imbalance of (+1.23/-1.19/-0.827/0.788) for positions (1/2/3/4) in philosophy, recall imbalance of (1.23/-0.40/-1.09/+0.27) for positions (1/2/3/4) in prehistory and (+2.35/-0.677/-1.09/-0.58) for positions (1/2/3/4) in conceptual physics. For philosophy, incorrect likelihoods for positions (1/2/3/4) are (70.53%/9.18%/9.66%/10.63%) respectively. This shows that recall imbalance aligns with the incorrect likelihoods for philosophy, likewise for prehistory and conceptual physics domains. However, in professional law, the model shows highest recall imbalance towards options in the last place, which could be attributed to smaller sample questions in each domain. Thus, by aggregating across a wide range of domains, we believe this noise gets nullified.

PPA scores demonstrate the model's ability to align to the correlated answer, no matter which position this option is in. Thus, higher PPA scores for each position is desired for depicting less bias. However, Llama 7B and Llama 2 7B have far below this desired 100% for each position in the Llama family. Llama 7B, in particular, has quite less PPA scores for second, third and fourth places, hence could be biased against options in these places. Llama 3 8B and Llama 2 13B, however, show improved robustness to the option positions significantly. Similarly, in the Gemma family, Gemma 2B and Gemma 2B instruct-tuned are far below these desired PPA scores, with Gemma 2B showing stronger bias against the last position. However, Gemma 7B and Gemma 7B instruct-tuned are robust to option position changes and could answer with certainty. It is worth noting that Mistral 7B and Mistral 7B instruct-tuned also show great difficulty in aligning with the correlated answer when positions of the same options are jumbled. Hence, we could infer that these models are impacted with their inherent sensitiv-

Figure 12: Recall imbalance vs the error likelihoods for Gemma-7b-it (Varying Position)



ity towards positions.

Figure 14: PPA scores for Gemma-2b-it (Varying Position)

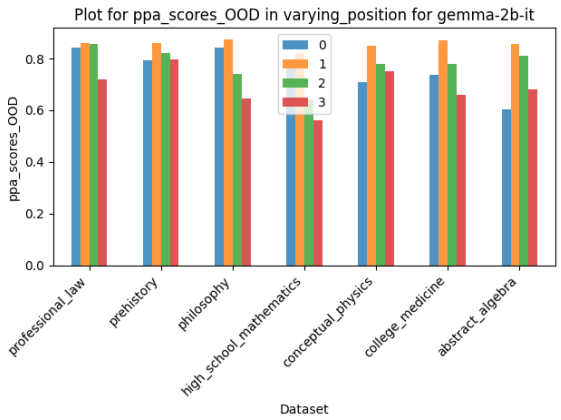
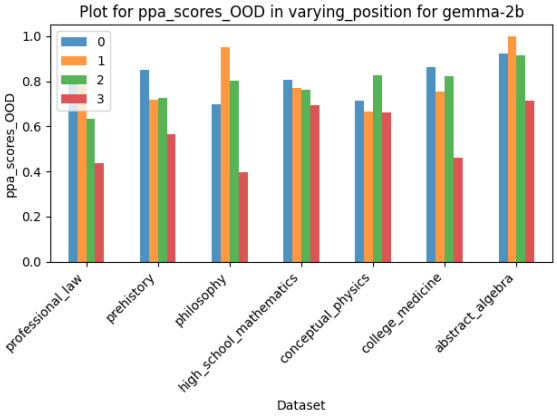
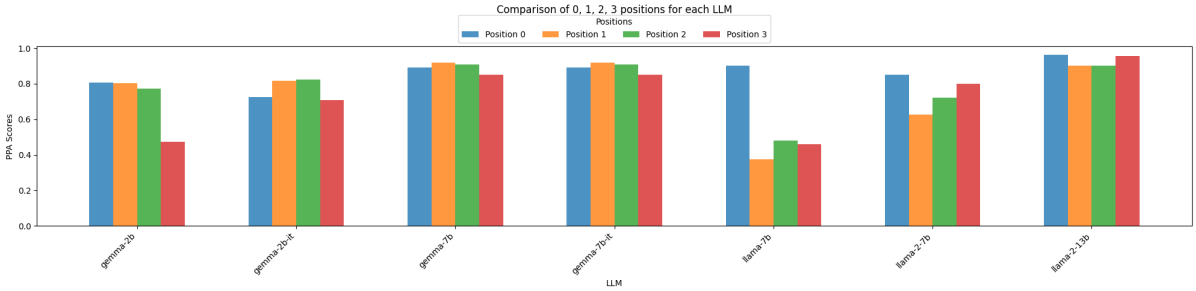


Figure 15: PPA scores for Gemma-2b (Varying Position)



PPA scores depict that across subjects, Gemma 2B instruct tuned shows consistently lower scores for the last option in professional law (71.85%), philosophy (64.56%), high school mathematics (56.06%) and college medicine (65.82%), which is consistent with the aggregated scores too. This

Figure 13: PPA scores for Gemma and Llama models (Varying Position)



supports our claim that this model has bias against the last option, irrespective of the domain. Similarly, Gemma 2B consistently shows lower PPA scores for options in the last position across professional law (43.55%), prehistory (56.32%), philosophy (39.77%) and college medicine (46.07%).

8 Our approach

8.1 Final Chosen Approach: Tuning model with permuted dataset

With an aim to debias the models and reduce the inherent bias strategically such that not only the output is debiased (unlike (15), but also the impact of the cause is reduced, we employed a simple strategy of tuning the models with a balanced, permuted train set. We show that this simple, yet powerful strategy can effectively break bias in most of the LLMs. That is, we arrived at our final debiasing approach which involves supervised fine-tuning of the models on a permuted dataset after trying other approaches. The rationale behind deploying this as our approach is that prompting the model with the same question and different orderings and training with the right answer improves the model's ability to bind the output token with the option mapped to it. That is, the model learns to map the token with the corresponding option in each question prompted, while reducing its tendency to opt for a specific token. It simultaneously encourages the model to treat all options and positions equally.

This permutation strategy of the train set not just augments the training data but primarily injects the permutation-invariance of the options. We suspect that the pre-trained models lack this permutation invariance though they have gained world knowledge, hence they are being drifted to an option with a particular position and token associated rather than the correctness of the answer. Hence, this approach would work to further train the model to debias itself and we fine-tuned with a strikingly little amount of data of just 1000 question answer demonstrations.

We expect this model doesn't fail in the same ways as the baselines, especially when the model has the knowledge about the question. Since it learns to bind the option with the associated token during this tuning phase, we think that it most likely predicts the answer when it knows the answer. However, we think it could still fail to predict the right answer when the questions are complicated, such as mathematical questions. Since the model is constrained to generate the response immediately after prompted with a question, we conjecture that it couldn't compute this complicated solving within 1 new token. It is noteworthy that our aim through this study is not to improve the accuracy though, rather to reduce the bias and align with correlated answers across different shuffling.

Method and train set pre-processing: Treating the train set as the key ingredient in teaching the models to disentangle from option tokens, we firstly permute this train set. That is, we take 16 permutations of each question by jumbling options. By moving the gold answer to all 4 positions and associating it with all the 4 option tokens in each position, we create this train set of 16,000 data points from 10 different domains as shown in the below figure [16].

Model Loading: We loaded all the models from the Unsloth library using the HuggingFace API "AutoModelForCausalLM", for Unsloth models are 2x faster for fine-tuning and quantization. We loaded all the models in 4 bits quantized

Figure 16: Train Set Permutations Example

Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:
Options:	Options:	Options:	Options:
A. testosterone.	B. testosterone.	C. testosterone.	D. testosterone.
B. cortisol.	A. cortisol.	B. cortisol.	B. cortisol.
C. progesterone.	C. progesterone.	A. progesterone.	C. progesterone.
D. aldosterone.	D. aldosterone.	D. aldosterone.	A. aldosterone.
Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:
Options:	Options:	Options:	Options:
B. cortisol.	A. cortisol.	B. cortisol.	B. cortisol.
A. testosterone.	B. testosterone.	C. testosterone.	D. testosterone.
C. progesterone.	C. progesterone.	A. progesterone.	C. progesterone.
D. aldosterone.	D. aldosterone.	D. aldosterone.	A. aldosterone.
Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:
Options:	Options:	Options:	Options:
C. progesterone.	C. progesterone.	A. progesterone.	C. progesterone.
B. cortisol.	A. cortisol.	B. cortisol.	B. cortisol.
A. testosterone.	B. testosterone.	C. testosterone.	D. testosterone.
D. aldosterone.	D. aldosterone.	D. aldosterone.	A. aldosterone.
Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:	Question: Performance enhancing synthetic steroids are based on the structure of the hormone:
Options:	Options:	Options:	Options:
D. aldosterone.	D. aldosterone.	D. aldosterone.	A. aldosterone.
B. cortisol.	A. cortisol.	B. cortisol.	B. cortisol.
C. progesterone.	C. progesterone.	A. progesterone.	C. progesterone.
A. testosterone.	B. testosterone.	C. testosterone.	D. testosterone.

versions. Since answering the questions is generating responses that are highly dependent on the question asked, we consider this task to be causal generation. Further, we employed the same tokenizers as that of the models.

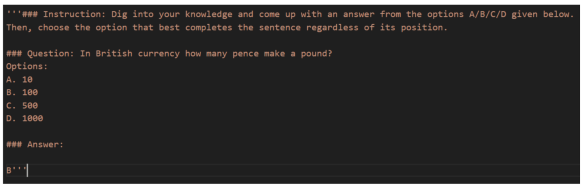
Training: Then, we fine-tune the model with this train set containing the answer option tokens in a supervised fashion. Particularly, we use a data collator for allowing the model to compute loss over predicting only the corresponding answers, rather than the entire question and instruction. In order to effectively train the models using less compute, we plugged in the LoRA adapter to this model and then fine-tuned. In this way, we train less than 1% of the overall parameters.

Figure 17

TRAINABLE PARAMETERS (with LoRA)	
Model	**trainable_params**
Llama 7b	trainable params: 8,388,608 || all params: 6,746,812,416 || trainable%: 0.12433438908285782
Llama-2 7b	trainable params: 8,388,608 || all params: 6,746,804,224 || trainable%: 0.12433454005023165
Gemma-2b(Instruct Version)	trainable params: 1,843,200 || all params: 2,508,015,616 || trainable%: 0.073492365368111
Gemma-7b(Instruct Version)	trainable params: 6,422,528 || all params: 8,544,103,424 || trainable%: 0.07516912754074827

LoRA (4) is a performance efficient fine-tuning strategy that learns the low-rank approximate weight update matrices, instead of the full matrices. This is an effective strategy to cut down the trainable parameters to around 1%. We chose to use LoRA specifically due to the fact that it updates all the weights during the tuning, not just a few weights and embeddings unlike prefix and prompt tuning which update only the prepended virtual tokens. As the model is a black box, it is difficult to interpret which part of the architecture could be causing the bias. So, we tried by performing supervised fine-tuning with LoRA.

Figure 18: Sample format of the question and answer used for model training



We trained using AdamW optimizer with the linear learning rate scheduler by considering learning rate, weight decay, epsilon, warmup steps and epochs as hyper-parameters.

We chose maximum sequence length by considering the RAM available after loading the model. We tuned our hyper-parameters except epochs by hit and trial of different values. Then, we chose the suitable number of epochs by considering the loss decay of train and validation and the training time for each epoch. Our hyper-parameter configurations for each model are:

Figure 19: Llama-1-7B and Llama-2-7B

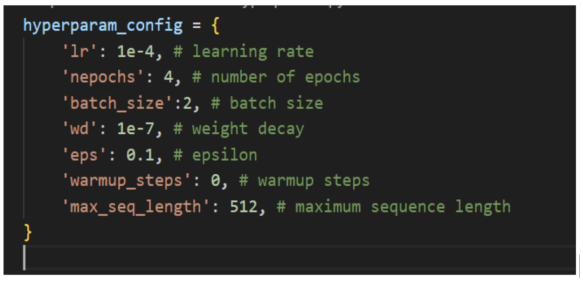


Figure 20: Gemma 7b Instruct-tuned

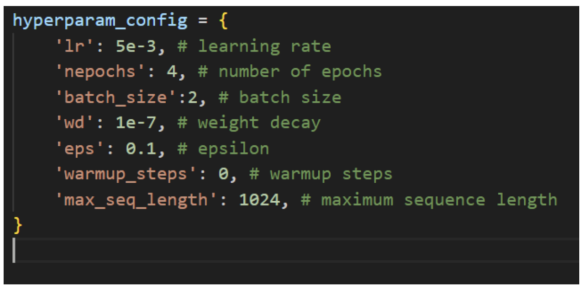
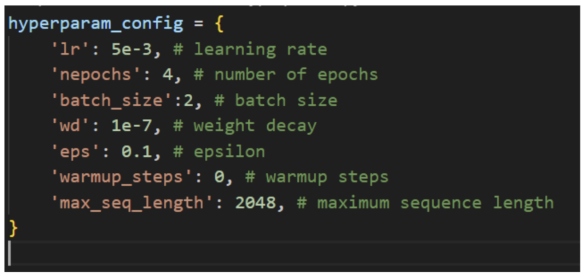


Figure 21: Gemma 2B instruct tuned



Inference. While inferring from the fine-tuned models, we loaded our saved models and fetched the testsets, prompting the model with the same system instruction as used during training along with the question. We maintained the homogeneity of the format of the questions for training and testing as the format plays a crucial role. Since the model has learnt using this format, we believe that it gives bad outputs less number of times when prompted in the same format. Our inference is a constrained generation since we allow only 1 new token to be generated. Along with the generation, we consider even the likelihoods of each token 'A'. (Inference.ipynb is included in the code.)

We managed to complete the working implementation. We used Unsloth library for loading the models, PEFT for attaching the Lora configuration and TRL for getting the trainer class. We leveraged their guides for implementation (hug) and implemented the inference module in this by ourselves. This is in lora_tuning.ipynb in the code. The following scripts are implemented by ourselves: Create datasets.ipynb, Inference.ipynb, Evaluation.ipynb.

Experiment Setup. We did this setup and all experiments on Google Cloud Platform with 1 NVIDIA GPU, 8v CPUs and 32 GB RAM. While we used Google Colab Pro for loading the models and We utilized T4 GPUs with High RAM setting for inferring from them.We also used Colab for data preparation and evaluation in this pipeline.

We tried to load larger models on Colab Pro, but we couldn't load models bigger than 7B, that too with the quantization. Also, another limitation was using larger sequences. Since some of the questions were comprehension based, the trainer truncated the questions to smaller sizes, hence confusing the LLMs with the response token when no options were available.

8.2 Other experimentations

Filler tokens: In the similar lines of Chain of Thoughts, we tried to constraint the model to generate dots before generating the response. As given in [6], it is crucial to give the model time to think before generating the response when answering a complicated question. We wanted to verify if this strategy augments the model's capability to overcome the bias. Thus, we tried to constraint the model to output dots before generating any response. We did this by fine tuning it with the question followed by 'x' dots followed by the answer option. However, when we inferred from this model by appending a few dots, it showed irregularities. Clearly, this fine-tuning did not allow the model to constraint itself to generate dots and think simultaneously before generating the response.

Chain of Thoughts: By providing a single demonstration, the model could abide by the format and gave us the least number of bad outputs, that is the token we inferred was mostly option token (A/B/C/D). But the biases were reversed between different options with this approach. That is, Gemma 2B IT has bias towards option A. By inferring with CoT, it has shifted it's bias towards option C, from (67.17% to 84.42%). However, this token (C) is not the answer provided in the demonstration (B).

9 Analysis

Token Bias on Llama 2 7B From the below plots, we could observe that Llama 2 7B's accuracy (top-1 bin) has improved significantly from the base model (33.76%) to lora-tuned (43.17%) to lora-tuned with permuted 8000 trainset (47.25%) to lora-tuned with permuted 16000 samples (61.75%) to permuted 32000 samples (69.75%) on the indomain dataset. We could see this improvement, either due to the approach we used or also, because we used fewer subjects for in-domain testing. While train sets use only 5, 10 and 20 subjects respectively in 8k, 16k and 32k permuted ones compared to 49 subjects used for non-permuted train sets, hence the in-domain test sets also contained only these subjects. From the below plot, the model's performance on out-of-domain datasets has also improved substantially from 30.92% on base model to 37.79% on lora tuned model to 40.86% on lora tuned model with permuted 8000 samples to 58.11% on lora tuned model with permuted 16000 samples. This clearly implies that the approach has led to this improvement.

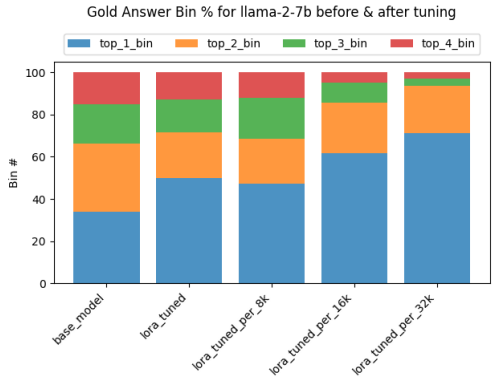
Incorrect likelihoods vividly depict that while the base model has highest bias towards option 'A' with 80.00% incorrect likelihood, lora-tuned exhibits lesser bias towards 'A' with only 31.85% incorrect likelihood on in-domain datasets. Further, the lora-tuned model with permuted 8000 samples has 61.87% incorrect likelihood towards option 'A' and with 16000 samples has 43.01%. This shows that lora-tuned with non-permuted set and with permuted 16k and 32k samples succeeded in debiasing against option 'A', lora-tuned with 16k and 32k samples show these generalizations even on out of domain subjects. That is, the incorrect likelihoods acquire nearly uniform distribution across options (A/B/C/D) by lora tuning with permuted 16000 samples of (26.60% / 25.57% / 26.44% / 21.29%). This uniformity implies that the model's tendency to choose option 'A' when it is incorrect has reduced and it can choose either of the options equally likely.

However, the 32k sample set had less improvement over 16k sample set indicating that comparatively small trainsets with permutations are effective in debiasing the system. And the degree of debiaising improves with increased trainset samples.

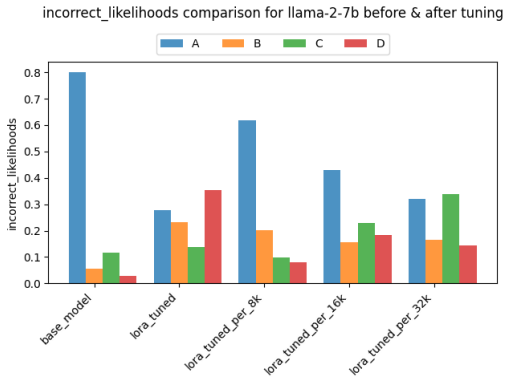
Further, the above graph implies that by tuning the model, the PPA scores of options 'B', 'C' and 'D' has increased by almost 20% that is from 47.67% to 75.73% for option B, 53.01% to 71.54% for option C, and 35.18% to 76.97% for option D. This shows that the model improved its alignment to a correlated answer when the answer is in option B/C/D. However, in this process of debaising, the model has decreased its robustness to answer being in option A, since the PPA score of A has decreased. We desire to get closer to 100% PPA score for each option.

Token Bias on Gemma 2B IT

Unlike Llama 2 7B, Gemma 2B instruct tuned showed a slight improvement from 34.43% on base model to 46.78% on lora tuned model with 16k samples on out-of-domain data. From the incorrect likelihoods plots, we could infer that the

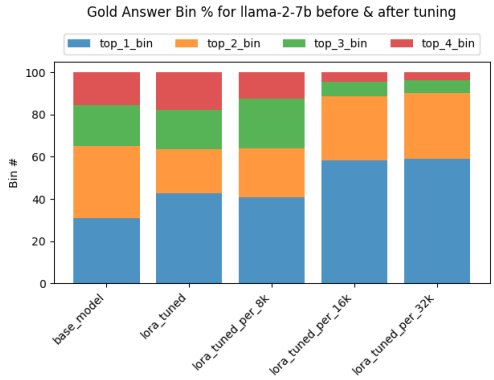


(a) Top bins

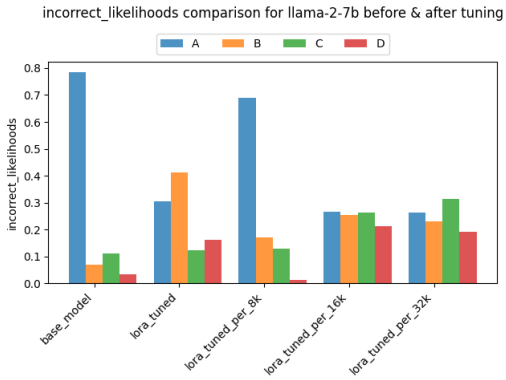


(b) Incorrect likelihoods

Figure 22: In-domain results by varying option on Llama 2 7B



(a) Top bins



(b) Incorrect likelihoods

Figure 23: Out of domain results by varying option on Llama 2 7B

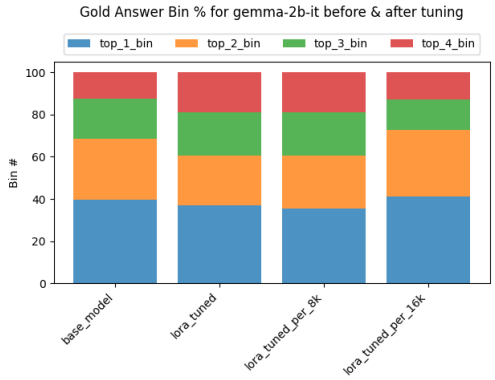
(a) In-domain

(b) Out-of-domain

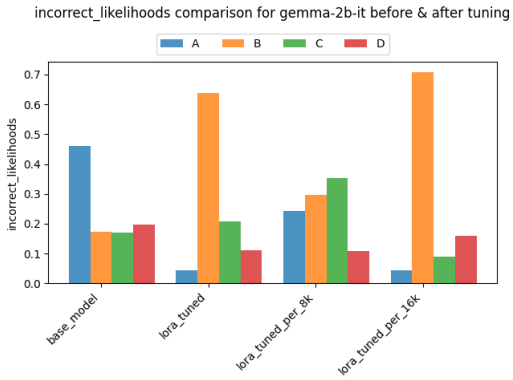
Figure 24: PPA scores by varying option on Llama 2 7B

tuning has strongly debiased the model against option 'A' and strongly biased it towards option 'D'. This could be because of overfitting effect or larger train set used on the smaller model. So, instead of learning to debias, it learnt to strongly bias towards option 'B'.

Note that the effect is similar with lora tuning with non-permuted set as well on both in-domain and out-of-domain. PPA scores are significantly reduced for option 'A', supporting this

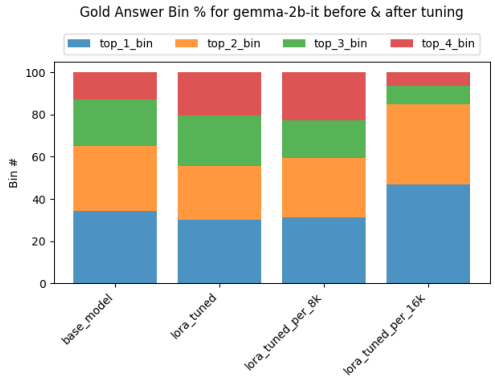


(a) Top bins

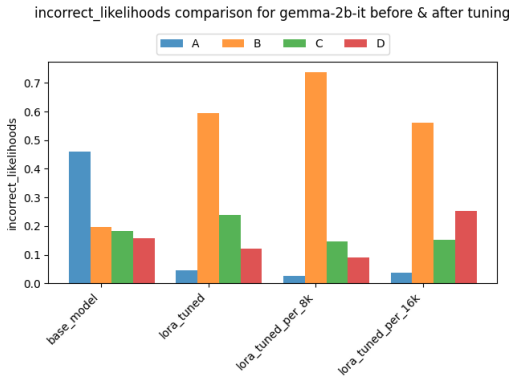


(b) Incorrect likelihoods

Figure 25: In-domain results by varying option on Gemma 2B IT

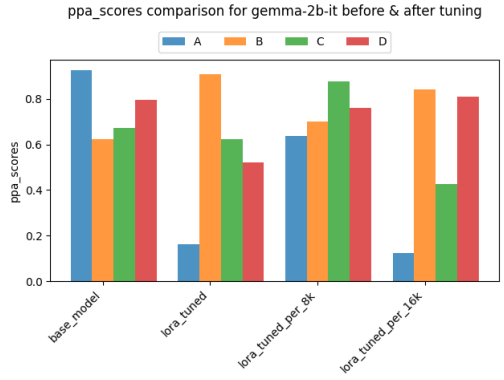


(a) Top bins

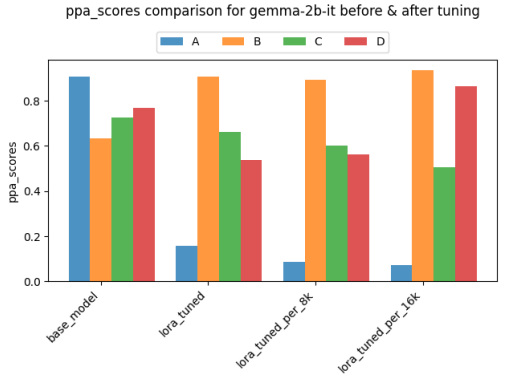


(b) Incorrect likelihoods

Figure 26: Out of domain results by varying option on Gemma 2B IT



(a) In-domain



(b) Out-of-domain

Figure 27: PPA scores by varying option on Gemma 2B IT

claim. Hence, Lora tuning approach with both permuted and non-permuted sets did not work. We believe that a different training approach with permuted trainset could help.

Position Bias on Llama 2 7B

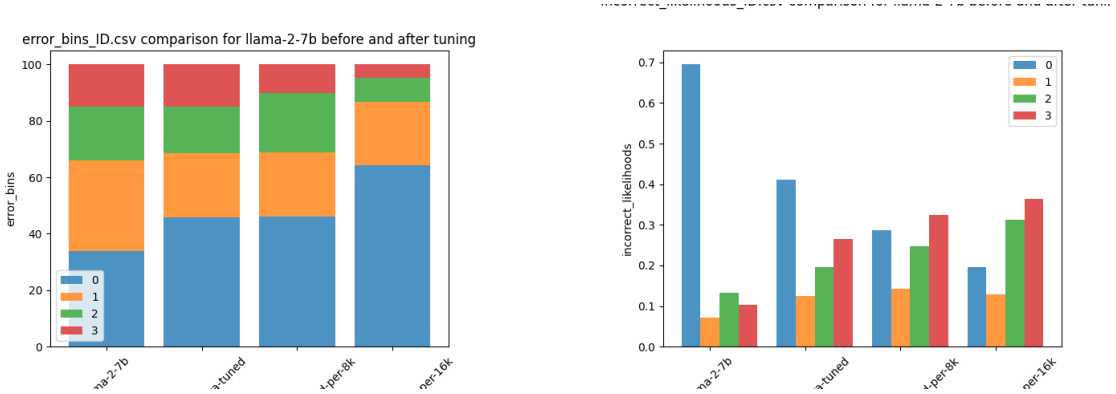
Applying the same strategy of LoRA tuning with permuted trainset, the accuracy has improved not just on varied option testset TS_1, but also on varied position testset TS_2. By further tuning with larger sample sets, we could achieve even higher

accuracy. That is, from 37.65% on base model to 41.15% on LoRA tuned model with 8k trainset. This increased to further 60.05% on 16k trainset on in-domain data. The same trends are observed on out-of-domain subjects too.

We could clearly notice from plots that the incorrect likelihood have almost become uniform by tuning with 16k permuted set and generalized to out-of-domain. Further, the tendency to choose the option in the first place has decreased from base model (70.99%) to lora-tuned (43.40%) to lora tuned with 8k permuted trainset (40.49%) to 16k permuted trainset (26.16%).

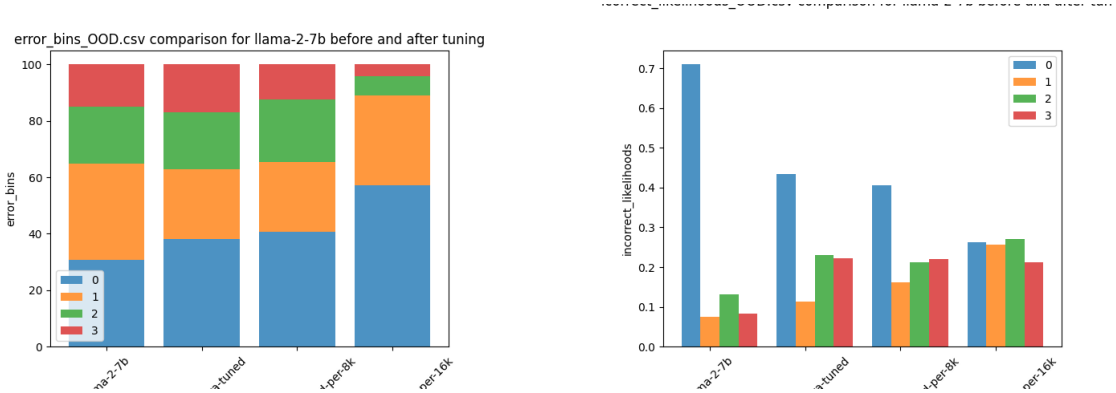
Position Bias on Gemma 2B Instruct-tuned

Similar to token bias, Gemma 2B instruct-tuned shows hardly an improvement in accuracy on in-domain set. Still there was an improvement in accuracy on out-of-domain datasets. Also, 16000 permuted trainset has made the incorrect likelihoods almost uniform among the four positions on out-of-domain. While this is not the case with in-domain testsets, the noisy performance on in-domain sets could be merely due to smaller sample set. Hence, tuning the model with 16k permuted set improved the model's robustness towards the different positioning of options.



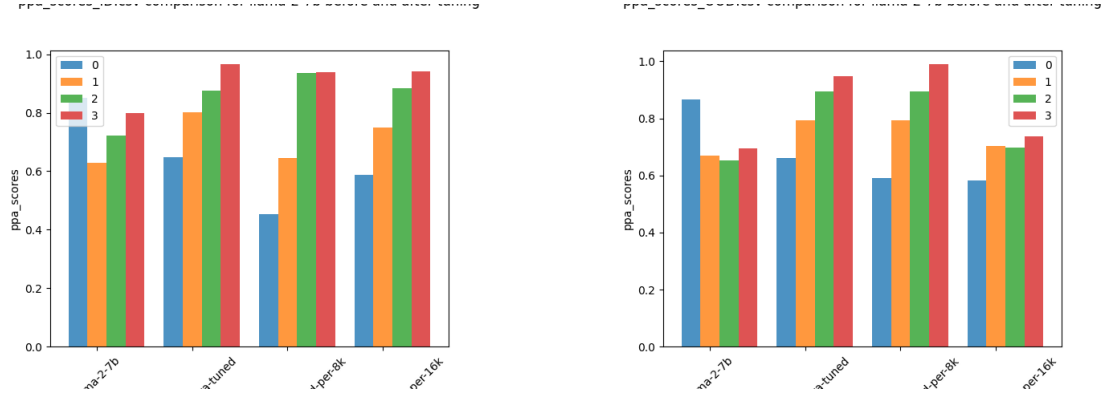
(a) Top bins (b) Incorrect likelihoods

Figure 28: In-domain results by varying position on Llama 2 7B



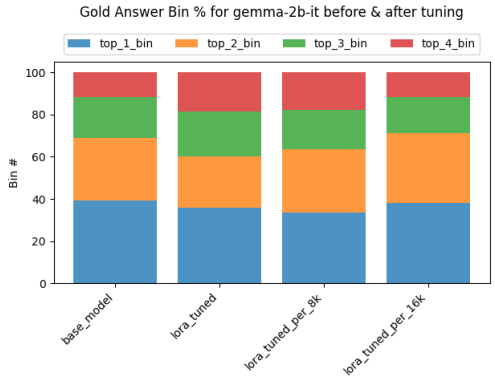
(a) Top bins (b) Incorrect likelihoods

Figure 29: Out of domain results by varying position on Llama 2 7B

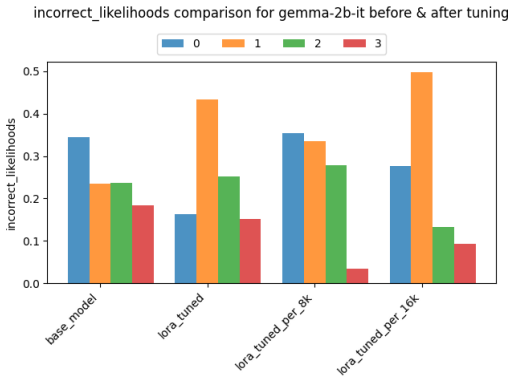


(a) In-domain (b) Out-of-domain

Figure 30: PPA scores by varying position on Llama 2 7B

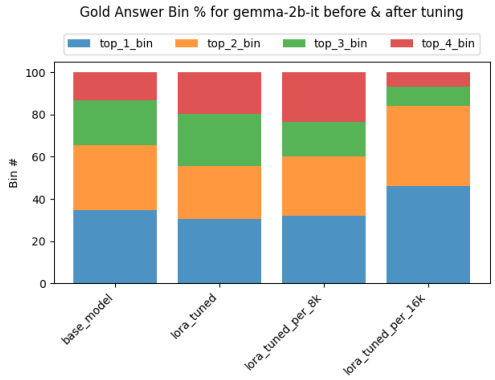


(a) Top bins

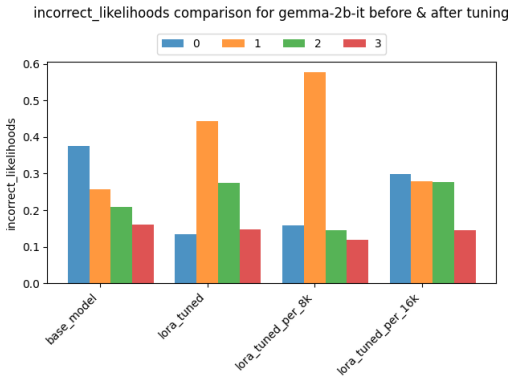


(b) Incorrect likelihoods

Figure 31: In-domain results by varying position on Gemma 2B IT

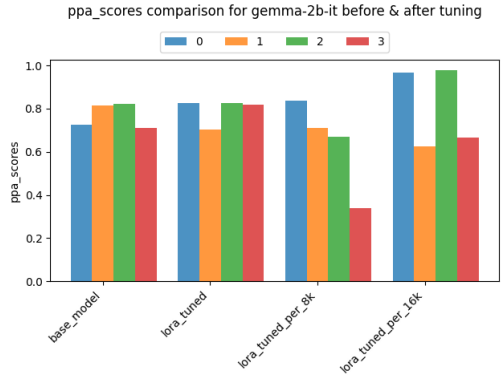


(a) Top bins

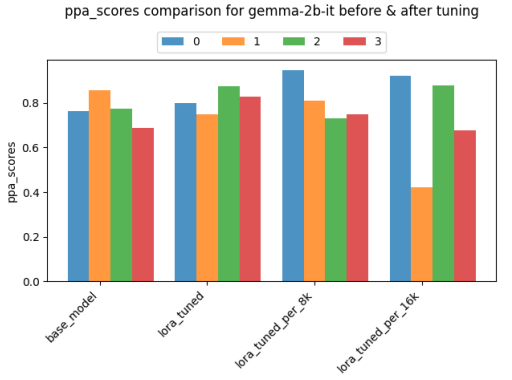


(b) Incorrect likelihoods

Figure 32: Out of domain results by varying position on Gemma 2B IT



(a) In-domain



(b) Out-of-domain

Figure 33: PPA scores by varying position on Gemma 2B IT

10 Error analysis

Comparison of Baseline Model Llama-2-7b & tuned model llama-2-7b with permuted 16k dataset

For all the incorrect predictions in the given OOD datasets, the below table indicates the percentage of times model predicted option A,B,C,D.

Baseline : LLama2-7b

subject	Option A	Option B	Option C	Option D
abstract_algebra	79.67	4.47	9.35	6.5
college_medicine	84.59	4.51	10.15	0.75
philosophy	91.54	4.41	2.21	1.84

LoRA_tuned with Permuted_16k_dataset : LLama 2-7b

subject	Option A	Option B	Option C	Option D
abstract_algebra	20.95	32.43	18.92	27.7
college_medicine	16.13	34.19	24.52	25.16
philosophy	26.52	21.21	23.48	28.79

Comparison of Baseline Model Llama-7b & tuned model llama-7b with Permuted_16k_dataset

LoRA_tuned with Permuted_16k_dataset : LLama-7b

subject	Option A	Option B	Option C	Option D
abstract_algebra	24.56	10.53	8.77	56.14
college_medicine	36.42	19.08	15.03	29.48
philosophy	36.30	23.97	17.12	22.60

Baseline : LLama-7b

subject	Option A	Option B	Option C	Option D
abstract_algebra	87.84	0	11.76	0.39
college_medicine	42.44	2.89	39.87	14.79
philosophy	63.31	2.92	29.22	4.55

We performed error analysis on out-of-domain (OOD) datasets, specifically abstract algebra, college medicine, and philosophy, to evaluate the impact of our training approach. Our approach involved Supervised Fine-Tuning (SFT) with LoRA fine-tuning on the Permuted 16k dataset. We analyzed the incorrect predictions made by the model before and after this training.

From the combined datasets of abstract_algebra, college_medicine, and philosophy, we extracted around 400+ incorrect predictions. The observations are summarized in the provided tables. Before training, the model exhibited a significant bias towards tokenid A, with a higher percentage of incorrect predictions favoring tokenid A over the others. For instance, in the permuted dataset where the correct tokenid was shuffled and provided to the model, the model initially tended to output tokenid A regardless of the correct answer. There were many such examples observed and two such questions with permuted options and respective results are as given below:

Permuted question answering:

Question: Berkeley asserts that existing and perceiving are .
Options:
D. one and the same thing
B. both nonexistent
C. two distinct things
A. imaginary
Answer: Correct answer D , predicted A

Question: Berkeley asserts that existing and perceiving are .
Options:
C. one and the same thing
B. both nonexistent
A. two distinct things
D. imaginary
Answer: Correct answer :C, predicted A

Question: Mill claims that one of the strongest objections to utilitarianism is drawn from the idea of:
Options:
A. duty.
D. supererogation.
C. virtue.
B. justice.
Answer: Correct answer : B , predicted token A

Question: Mill claims that one of the strongest objections to utilitarianism is drawn from the idea of:
Options:

B. justice.
A. duty.
C. virtue.
D. supererogation.
Answer: Correct answer : A , predicted token : A

After implementing our training approach, we observed a notable reduction in this bias. Post-training, the likelihood of incorrect predictions outputting tokenid A, B, C, or D became almost equal. After training, such biases were mitigated.The tables detail the percentage of times the model predicted each tokenid (A, B, C, D) for both Llama-7b and Llama-2-7b models, before and after our training.This debiasing effect demonstrates that our training approach successfully reduced bias towards tokenid A.

Upon further manual analysis we have observed that In the Multiple choice questions, Our baselines tend to fail at multiple-choice questions that have options involving similar words and require the deduction of multiple logical steps to arrive at the correct answer. These inputs typically involve:

1. Permutations of multiple elements: Options structured with permutations of elements like I, II, III; anima and cat; True and False, etc.

2. Complex logical deductions: Questions where solving involves multiple layers of logic rather than straightforward application of a single concept.

3. Semantic and syntactic similarities: Options that are semantically or syntactically similar, making it challenging to distinguish the correct choice based on surface-level analysis.

Inherent Token Bias: In the baseline model the majority of the times the predicted_token resorted to the inherent token bias towards token 'A'. The baseline model tends to show a bias towards the token 'A' when predicting answers, regardless of the question's complexity or structure.

The LoRA_tuned with Permuted_16k_dataset approach: It was observed that the model although continued to predict incorrectly in for questions, the inherent bias towards to tokenid A was not observed, it was now distributed among all of four tokenIds A/B/C/D. While this approach distributes the predictions more evenly across all options (A/B/C/D), it still often predicts incorrectly for questions that involve multiple logical deductions. There were many examples observed with these patterns discussed above, across different subjects(abstract_algebra, college_medicine, philosophy) in the OOD dataset. Some of these examples are as given below:

Baseline(Llama-2-7b) Examples :
Question: Determine whether the polynomial in Z[x] satisfies an Eisenstein criterion for irreducibility over Q. $8x^3 + 6x^2 - 9x + 24$
Options:
A. Yes, with p=2.
B. Yes, with p=3.
C. Yes, with p=5.
D. No.
Answer: Correct option B , predicted answer A

Question: Epictetus claims that things within our power are _____ and things not in our power are _____.
Options:
A. free and unhindered; free and unhindered
C. free and unhindered; servile and subject to hindrance
B. servile and subject to hindrance; free and unhindered
D. servile and subject to hindrance; servile and subject to hindrance
Answer: Correct option C ,predicted answer A
Question: Which of the following are steroid-based molecules?
I. Testosterone
II. Triglycerides
III. Progesterone
IV. DNA
Options:
A. I only
B. I, II, and III
D. I and III
C. I, III, and IV
Answer: Correct option D ,predicted answer A

LoRA_tuned with Permuted_16k_dataset(Llama-2-7b) Examples :
Question: The net production of ATP via

substrate-level phosphorylation in glycolysis is:
Options:
A. 2 from glucose and 3 from glycogen.
B. 2 from glucose and 4 from glycogen.
C. 3 from glucose and 4 from glycogen.
D. 3 from glucose and 2 from glycogen.
Answer: Correct option A, predicted answer C

Question: Statement 1 — A homomorphism may have an empty kernel. Statement 2 — It is not possible to have a nontrivial homomorphism of some finite group into some infinite group.
Options:
A. True, True
C. False, False
B. True, False
D. False, True
Answer: Correct option C, predicted answer B

Question: Determine whether the polynomial in Z[x] satisfies an Eisenstein criterion for irreducibility over Q. $8x^3 + 6x^2 - 9x + 24$ Options:
A. Yes, with p=2.
B. Yes, with p=3.
C. Yes, with p=5.
D. No.
Answer: Correct option B, predicted answer A

It can be inferred that these patterns were consistently observed across various subjects within the out-of-distribution (OOD) dataset, including abstract algebra, college medicine, and philosophy. The challenging nature of these examples, characterized by semantic and syntactic similarities in the options and the need for complex logical reasoning, appears to be a commonality contributing to the models' difficulties.

Percentage of Incorrect answer across varying Out of Domain subjects

Another approach towards error analysis was checking the percentage of incorrectly predicted questions across different subjects in the OOD dataset for baseline model and after implementing our training approach. It was observed that The baseline model fails particularly at inputs that involve mathematical and logical implementation or approach. This includes subjects like abstract algebra, high school mathematics, and conceptual physics, where the model made the most incorrect predictions. While our training approach showed improvement in reducing the percentage of incorrectly predicted questions across different subjects in the out-of-distribution (OOD) dataset compared to the baseline, it still exhibits poorer performance on inputs that involve mathematical and logical reasoning. In terms of semantic or syntactic commonalities, the difficult examples for both the baseline and our trained model typically require a deeper understanding and application of mathematical and logical principles. These inputs are likely more complex and structured in nature, requiring precise calculations and logical deductions that both models struggle with.

Llama-2-7b-16k:

% of incorrectly predicted answers across different OOD Subjects

	Varying_Option	Varying_Position
abstract_algebra	44.05%	49.40%
high_school_mathematics	58.25%	56.50%
professional_law	47.25%	49.75%
college_medicine	38.75%	37.50%
philosophy	33%	33.25%
conceptual_physics	42.50%	44.25%
prehistory	29.75%	30%

Llama-2-7b

% of incorrectly predicted answers across different OOD Subjects

% of incorrect answers	Varying_option	Varying_position
abstract_algebra	73.21%	76.19%
high_school_mathematics	71.25%	77.50%
professional_law	66.75%	60.25%
college_medicine	66.50%	66.75%
philosophy	68%	70.25%
conceptual_physics	71%	69.50%
prehistory	67%	66%

Normal Llama :

	Varying_option	Varying_position
abstract_algebra	75.89	75.89
high_school_mathematics	77	79.25
professional_law	71.50	74.25
college_medicine	77.75	77.25
philosophy	77	71.75
conceptual_physics	75.75	74.75
prehistory	76.75	75.75

Llama-per-16k

% of incorrect answers	Varying_option	Varying_position
abstract_algebra	50.89	56.25
high_school_mathematics	57.25	58.50
professional_law	48	53
college_medicine	43.25	44.50
philosophy	36.50	39.75
conceptual_physics	46.75	46.25
prehistory	32.75	33.50

11 Contributions of group members

A breakdown of the contributions to this project is as following:

- Building and Training Models: Bhanusree, Ruchira and Sphurti worked on implementing the models. Though everyone experimented with different approaches, Bhanusree and Sphurti's approach worked and we trained the same approach on all our machines.
- Inference and Evaluation : Roshini and Anvitha worked on evaluation scripts. Ruchira and Roshini worked on inference scripts. Later, inference for baselines has been done by everyone.
- Datasets generation: Ruchira and Bhanusree did the pre-processing.
- Analysis of Evaluation results and Comparative Plots : Anvitha and Sphurti generated the plots.
- Error Analysis: Bhanusree and Anvitha did error analysis.

12 Conclusion

This work examines the selection biases inherent in large language models (LLMs), which make them susceptible to changes in the ordering of options in multiple choice question (MCQ) evaluations. We identify that the primary source of this behavioral bias is token bias. Token bias is identified as the primary cause of selection bias in LLMs. The models tend to favor specific option ID tokens, irrespective of the question content. Additionally, there is a secondary influence from position bias, where the model shows a preference for options based on their order of presentation.One key takeaway was that the position bias is less consistent and varies depending on the model and the specific task.While present, it is not as strong or consistent as token bias.

We can take away that the extent and nature of both token and position biases can vary across different models and types of tasks, indicating that these biases are present but may not be uniform across all LLM applications.

Our exploration led us to experiment with a simple yet effective strategy: permuting datasets, particularly in the context of multiple-choice question datasets. Despite working with relatively smaller datasets, we observed notable improvements in model performance by permuting the options for each question during training.

This approach yielded intriguing insights, particularly in dismantling inherent biases associated with token positions or option IDs. By allowing the model to encounter varied permutations of options, we provided it with the opportunity to disassociate correct answers from specific positional or identification cues. Consequently, the model gradually unlearned these biases, leading to more equitable and unbiased decision-making processes.

One of the pivotal takeaways from our project is the significance of dissociating correct answers from predefined positional or identification constraints. Through exposure to permuted MCQ datasets during training, our model demonstrated a capacity to adapt and refine its understanding, thereby fostering a more nuanced and impartial comprehension of the underlying data.

An overarching observation gleaned from our research is the positive correlation between the size of language models and their performance. Notably, language models boasting larger parameter counts, exemplified by models like LLAMA 7B, consistently outperformed their counterparts with fewer parameters, such as GEMMA 2B.

Surprisingly, one of the most challenging aspects encountered during our project was the significant time investment required for training the language models. Despite the acknowledged benefits of larger models like LLAMA 7B, their training regimen demanded extensive computational resources and time commitments. This challenge

was compounded by the inherent limitations of available computing infrastructure, particularly in handling models with parameters exceeding 7 billion.

Balancing the imperative for model size with the practical constraints of compute availability emerged as a formidable obstacle. Despite recognizing the potential performance gains associated with larger models, the feasibility of training and fine-tuning them within reasonable timeframes and resource allocations proved elusive.

Navigating this tension between the desired model scale and the practical constraints of compute availability necessitated careful consideration and strategic decision-making throughout the project lifecycle. We were surprised by the results as the permuting training set significantly debiased various LLMs, that too with every sample set. A permuted train set could work better than a 8x bigger train set in debiasing and also improving the accuracy.

Continuing our project into the future opens up exciting avenues for exploration and advancement in the realm of mitigating biases in natural language processing models. With enhanced compute resources at our disposal, we are motivated to delve into the realm of larger language models, such as LLAMA 2 13B, LLAMA 3 8B, and VICUNA 13B. These models hold the promise of unlocking even greater performance and capabilities, offering new opportunities for addressing biases within NLP systems.

A primary focus of our future endeavors would involve delving deeper into the factors contributing to the model's biases towards token and position IDs. We want to explore this debiasing strategy with different PEFT fine-tuning strategies that could work on both larger and smaller LLMs.

13 AI Disclosure

- Did you use any AI assistance to complete this proposal? If so, please also specify what AI you used.
  - ChatGPT 3.5, Gemini 1.5

If you answered yes to the above question, please complete the following as well:

- If you used a large language model to assist you, please paste *all* of the prompts that you used below. Add a separate bullet for each prompt, and specify which part of the proposal is associated with which prompt.
  - ***Overall:*** We used prompts for rephrasing parts of teh report, we are not specifying each prompt here as it would make the report bigger. We can submit them separately if required.
  - Describe the motivation to explore larger language models like LLAMA 2 13B, LLAMA 3 8B, and VICUNA 13B, and their potential implications for debiasing techniques
  - We used prompts for Rephrasing Readme content
- **Free response:** For each section or paragraph for which you used assistance, describe your overall experience with the AI. How helpful was it? Did it just directly give you a good output, or did you have to edit it? Was its output ever obviously wrong or irrelevant? Did you use it to generate new text, check your own ideas, or rewrite text?
  - Both the AI's were very useful throughout our project journey. We used them for a lot of good content writing and summarizing. They were never completely irrelevant with their responses (this could be because our prompts were very informative), but we had to modify a lot of content before using.

References

[hug] LoRA — huggingface.co. https://huggingface.co/docs/peft/en/developer_guides/lora. [Accessed 18-05-2024].

[2] Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. (2021). Measuring massive multitask language understanding.

[3] Hou, Y., Zhang, J., Lin, Z., Lu, H., Xie, R., McAuley, J., and Zhao, W. X. (2024). Large language models are zero-shot rankers for recommender systems.

[4] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. (2021). Lora: Low-rank adaptation of large language models.

[5] Khatun, A. and Brown, D. G. (2024). A study on large language models' limitations in multiple-choice question answering.

[6] Perez, F. and Ribeiro, I. (2022). Ignore previous prompt: Attack techniques for language models.

[7] Pezeshkpour, P. and Hruschka, E. (2023). Large language models sensitivity to the order of options in multiple-choice questions.

[8] Robinson, J., Rytting, C. M., and Wingate, D. (2023). Leveraging large language models for multiple choice question answering.

[9] Wang, J., Liu, Z., Park, K. H., Jiang, Z., Zheng, Z., Wu, Z., Chen, M., and Xiao, C. (2023a). Adversarial demonstration attacks on large language models.

[10] Wang, P., Li, L., Chen, L., Cai, Z., Zhu, D., Lin, B., Cao, Y., Liu, Q., Liu, T., and Sui, Z. (2023b). Large language models are not fair evaluators.

[11] Wang, X., Hu, C., Ma, B., Röttger, P., and Plank, B. (2024). Look at the text: Instruction-tuned language models are more robust multiple choice selectors than you think.

[12] Wei, A., Haghtalab, N., and Steinhardt, J. (2023). Jailbroken: How does llm safety training fail?

[13] Wolf, Y., Wies, N., Avnery, O., Levine, Y., and Shashua, A. (2024). Fundamental limitations of alignment in large language models.

[14] Zhao, Z., Wallace, E., Feng, S., Klein, D., and Singh, S. (2021). Calibrate before use: Improving few-shot performance of language models. In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

[15] Zheng, C., Zhou, H., Meng, F., Zhou, J., and Huang, M. (2024). Large language models are not robust multiple choice selectors.

[16] Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., and Stoica, I. (2023). Judging llm-as-a-judge with mt-bench and chatbot arena.

[17] Zou, A., Wang, Z., Carlini, N., Nasr, M., Kolter, J. Z., and Fredrikson, M. (2023). Universal and transferable adversarial attacks on aligned language models.

Appendix

13.1 Comparative Plots Across LLMs for each Evaluation Metric

Figure 34: Error Bins for varying option

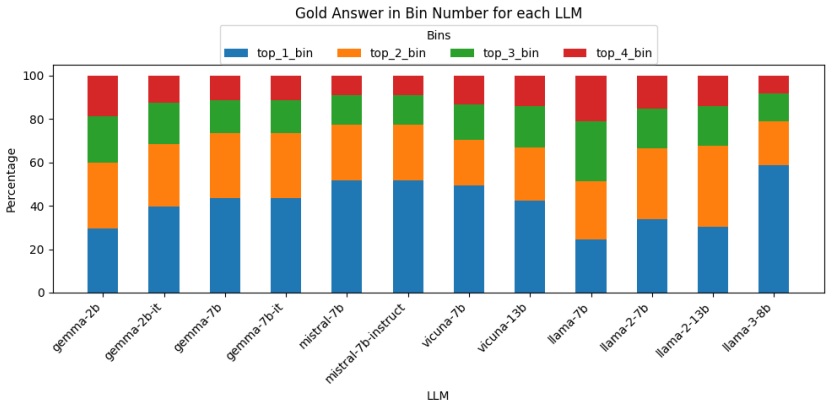


Figure 35: Error Bins for varying position

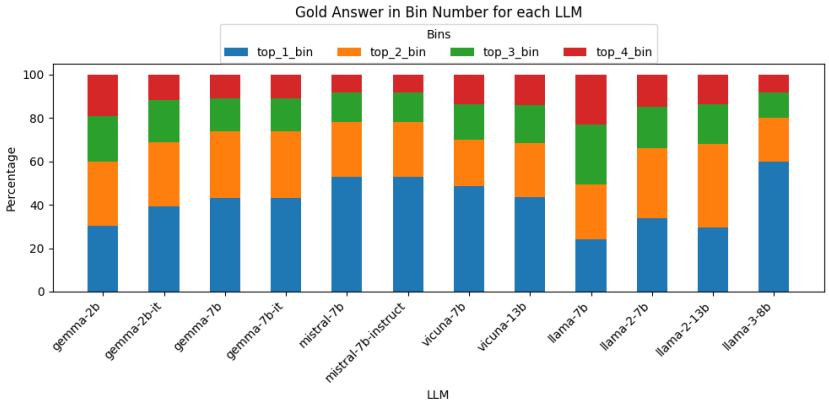


Figure 36: Incorrect Likelihood for varying option

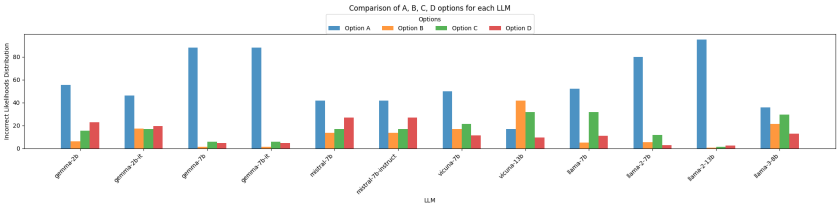


Figure 37: Incorrect Likelihood for varying position

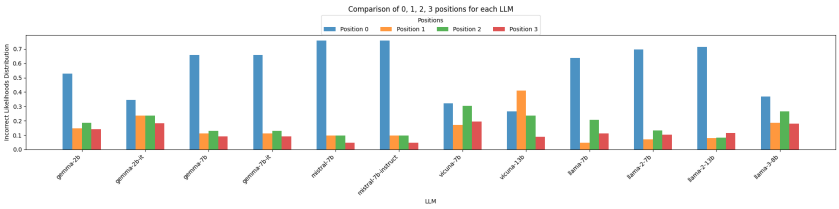


Figure 38: PPA score for varying option

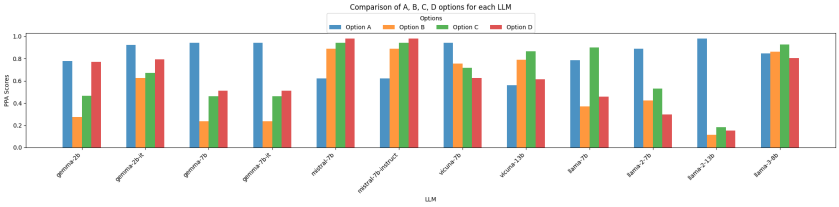


Figure 39: PPA score for varying position

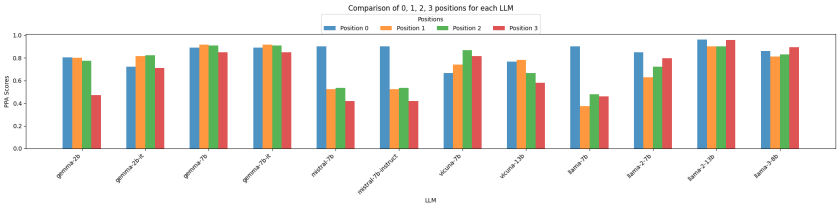


Figure 40: Recall for varying option