Large Language Models Sensitivity to The Order of Options in Multiple-Choice Questions

Pouya Pezeshkpour
Megagon Labs
pouya@megagon.ai

Estevam Hruschka
Megagon Labs
estevam@megagon.ai

Abstract

Large Language Models (LLMs) have demonstrated remarkable capabilities in various NLP tasks. However, previous works have shown these models are sensitive towards prompt wording, and few-shot demonstrations and their order, posing challenges to fair assessment of these models. As these models become more powerful, it becomes imperative to understand and address these limitations. In this paper, we focus on LLMs robustness on the task of multiple-choice questions—commonly adopted task to study reasoning and fact-retrieving capability of LLMs. Investigating the sensitivity of LLMs towards the order of options in multiple-choice questions, we demonstrate a considerable performance gap of approximately 13% to 85% in LLMs on different benchmarks, when answer options are reordered, even when using demonstrations in a few-shot setting. Through a detailed analysis, we conjecture that this sensitivity arises when LLMs are uncertain about the prediction between the top-2/3 choices, and specific options placements may favor certain prediction between those top choices depending on the question caused by positional bias. We also identify patterns in top-2 choices that amplify or mitigate the model's bias toward option placement. We found that for amplifying bias, the optimal strategy involves positioning the top two choices as the first and last options. Conversely, to mitigate bias, we recommend placing these choices among the adjacent options. To validate our conjecture, we conduct various experiments and adopt two approaches to calibrate LLMs' predictions, leading to up to 8 percentage points improvement across different models and benchmarks.

1 Introduction

Large Language Models (LLMs) have demonstrated impressive performance on various tasks, surpassing that of supervised models and, in some cases, even outperforming humans (Chowdhery et al., 2022; Touvron et al., 2023a; OpenAI, 2023). However, despite their impressive capabilities, previous research has highlighted certain limitations. For instance, LLMs have shown significant sensitivity to small changes in the prompt (Zhao et al., 2021; Wang et al., 2023a; Zhu et al., 2023). Therefore, a more comprehensive and conclusive analysis of different aspects that can affect/limit LLMs' performance is crucial for a fair assessment and their successful real-world adoption.

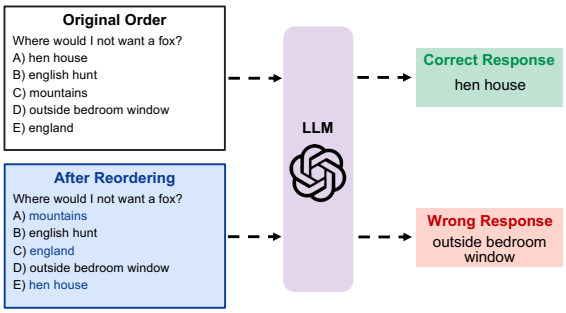

Figure 1: **GPT-4 sensitivity to reordering options:** upon changing the order of choices, GPT-4 changes its prediction from "hen house" to "outside of bedroom window" (the example is from CSQA dataset).

One significant limitation lies in the robustness of LLMs concerning the arrangement of various components in a prompt, as it directly impacts the assessment of their capability in understanding and reasoning for specific tasks. Prior research has demonstrated that LLMs exhibit sensitivity to the arrangement of few-shot demonstrations (Zhao et al., 2021) and the order of appearance for responses generated by candidate models when LLMs are used as referees to evaluate quality (Wang et al., 2023b). Given these findings, it becomes pertinent to inquire whether LLMs are also sensitive to the order of elements of the prompts in different tasks. For example, how much does the order of options in multiple-choice question (MCQ) answering tasks impact the LLMs performance.

Findings of the Association for Computational Linguistics: NAACL 2024, pages 2006–2017

In this paper, we investigate the sensitivity of LLMs to the order of options in MCQs; using it as a proxy to understand LLMs sensitivity to the order of prompt elements in in-context learning paradigm. We demonstrate an example of GPT-4's sensitivity to options order in Figure 1, using a sample from the CSQA benchmark (Talmor et al., 2018). Notably, by merely rearranging the placement of options among choices A, C, and E, GPT-4 incorrectly predicts the answer to be "outside bedroom window". Within this context, we aim to address the following research questions: (1) To what extent do LLMs exhibit sensitivity to the order of options in multiple-choice questions? (2) What factors contribute to LLMs' sensitivity to the order of options? (3) How can we improve LLMs' robustness to the order of options sensitivity?

To answer the first question, we conducted experiments using GPT-4 (OpenAI, 2023), InstructGPT (text-davinci-003) (Ouyang et al., 2022), and Llama-2-13b (chat version) (Touvron et al., 2023b) on five different multiple-choice question benchmarks. Surprisingly, we discovered a substantial sensitivity gap of up to 85% in the zero-shot setting. Additionally, in the few-shot setting, we observed that introducing demonstrations to the prompt only led to marginal improvements in LLMs' robustness if their performance increased. Moving on to the second question, we put forth a conjecture that the sensitivity of LLMs stems from their positional bias, wherein they tend to favor certain placements when uncertain about the answer among the top choices. To validate our conjecture, we analyzed instances where the models' predictions changed upon reordering the options. Furthermore, we showed that the complexity of the number of choices, while retaining the top possible answers, had only a gradual impact on the performance.

Additionally, we discerned patterns in the occurrence of top-2 possible choices that influence the model's probability of selecting a particular option or somewhat mitigate LLMs' positional bias. For amplifying bias, we found that the optimal strategy involves positioning the top two choices as the first and last options. Conversely, to mitigate bias, we recommend placing these choices among the adjacent options. To validate our findings, we conducted qualitative evaluations. Addressing the last question, we demonstrated that employing two different calibrating approaches led to a notable improvement in LLMs' performance, up to 8 percentage points. Through these investigations, we contribute to a deeper understanding of how the order of options affects LLMs' decision-making in MCQs and offer practical solutions to increase their robustness and accuracy in such scenarios.

2 Background and Experimental Details

This paper focuses on the task of multiple-choice question answering. In MCQs, the objective is to identify the correct answer to a given question from a set of possible options (see Figure 1). To address this task using LLMs, we present a prompt in the following format: "Choose the answer to the question only from A, B, C, D, and E choices. Question: {question}. Choices: {options}. Answer:" to the models. This in-context framing of multiple-choice questions is consistent with prior research (OpenAI, 2023; Savelka et al., 2023). Additionally, an illustrative example of our prompting approach and more experimental details are presented in Appendix.

Models: We considered three widely-used large language models, Llama-2-13b (chat version) (Touvron et al., 2023b), InstructGPT (text-davinci-003) (Ouyang et al., 2022) and GPT-4 (OpenAI, 2023). This selection aimed to represent a diverse range of LLMs, encompassing varying sizes and both open-source and commercial models. We primarily focus on these models due to their notable superior performance in the context of multiple-choice question answering tasks that require reasoning.

Data: To investigate the sensitivity of LLMs to the order of options and the reasons behind this phenomenon, we conducted experiments on five distinct MCQ benchmarks. These benchmarks are as follows: CSQA (Talmor et al., 2018): A commonsense multiple-choice question answering dataset, where each question is accompanied by 5 options. Abstract Algebra, High School Chemistry, and Professional Law from the MMLU benchmark (Hendrycks et al., 2020): These benchmarks consist of multiple-choice questions with 4 options provided for each question. And, Logical deduction from the Big-Bench dataset (Srivastava et al., 2022): This benchmark offers multiple-choice questions with 3 options for each question. Our selection of these benchmarks was guided by three specific criteria: (1) Domain diversity: We aimed to investigate the sensitivity to options order across different domains. (2) Varying option numbers: In order to explore the impact of the num-

Tasks	GPT-4			InstructGPT			Llama-2-13b		
	Vanila	Min	Max	Vanila	Min	Max	Vanila	Min	Max
CSQA	84.3	-12.6	+10.3	72.3	-24.0	+19.1	62.2	-28.9	+25.5
Logical Deduction	92.3	-8.1	+5.0	64.0	-39.4	+34.7	53.0	-30.7	+34.7
Abstract Algebra	57.0	-30.0	+23.0	33.0	-31.0	+39.0	32.0	-32.0	+53.0
High School Chemistry	71.9	-23.6	+18.2	44.8	-28.5	+38.0	40.6	-32.7	+45.6
Professional Law	66.1	-12.7	+12.1	48.6	-24.9	+25.7	43.8	-32.8	+32.9

Table 1: **Zero-shot order sensitivity**; all three LLMs display a notable level of sensitivity to the order of options across various benchmarks.

ber of provided options, we selected benchmarks with different option counts, namely 3, 4, and 5 options per question. And (3) performance levels: By incorporating benchmarks with varying levels of LLMs' demonstrated performance, we sought to better understand how model proficiency influences sensitivity to the options order. Although exploring a broader range of multiple-choice question tasks could enhance our comprehension of LLMs' sensitivity to options' order, due to constraints related to OpenAI API costs, we are compelled to narrow our focus to these five benchmarks.

3 Sensitivity to Order

In this section, we first investigate the sensitivity of LLMs to the order of options in the zero-shot setting. Then, we set out to determine whether introducing demonstrations to the prompt in the few-shot setting can enhance the models' robustness. To quantify sensitivity, we calculate the sensitivity gap, which is the difference between the maximum and minimum LLMs' performance when using an oracle ordering. In other words, we examine how specific reordering of options affects the models' predictions when the ground truth is known.

3.1 Zero-shot Sensitivity

The result of LLMs sensitivity to the order of options is presented in Table 1. Several noteworthy observations emerge from these results: (1) GPT-4 demonstrates significantly lower sensitivity gap compared to other LLMs. This suggests that GPT-4 is less affected by the rearrangement of options in the prompt, making it more robust in handling such variations. (2) Even in tasks where GPT-4 achieves high accuracy levels exceeding 90%, we still observe a considerable sensitivity gap of 13.1%. This indicates that even high-performing models are susceptible to changes in options order, which can impact their fair assessment. (3) Although the sensitivity gap shows some correlation with the models' performance, tasks where LLMs perform poorly do not necessarily exhibit higher sensitivity gaps. This suggests that factors beyond overall accuracy may also influence LLMs' sensitivity to options order. (4) The domain and the number of options in the MCQ tasks seem to affect the model's performance. However, we do not observe a clear correlation between these factors and the sensitivity gap. Given the poor performance of Llama-2-13b in comparison to InstructGPT and GPT-4 on the benchmarks, in the remainder of paper, we only focus on InstructGPT and GPT-4.

3.2 Can Demonstrations in Few-shot Setting Resolve the Sensitivity?

Having demonstrated the high level of sensitivity when zero-shot prompting LLMs, a crucial question that arises is whether adding demonstrations in the few-shot setting to the prompt can enhance the models' robustness. To address this, we select demonstrations in the few-shot setting by sampling the most similar instances. We achieve this by computing the Euclidean distance over vector representations of questions obtained from Sentence-RoBERTa (Reimers and Gurevych, 2019). The result of order sensitivity in the few-shot setting are visualized in Figure 2 (more detailed results are provided in Appendix). Each bar in the figure is accompanied by error bars, representing the range of maximum and minimum model performance achievable by reordering the options, with knowledge of the ground truth. From the results, we make the following observations: Firstly, the sensitivity gap consistently remains substantial even with the inclusion of more demonstrations in the few-shot setting. Furthermore, as performances improve, the sensitivity gap tends to shrink. However, adding more demonstrations does not necessarily lead to a reduction in the sensitivity gap. This highlights that while demonstrations may marginally improve robustness, they do not entirely mitigate the models' sensitivity to options order.

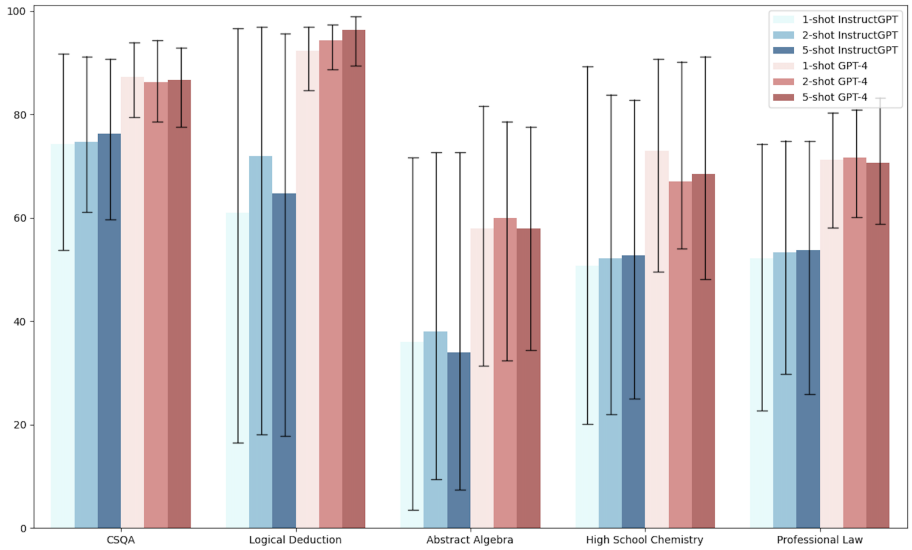

Figure 2: **Order sensitivity in the few-shot setting:** The error bars represent the range of minimum and maximum accuracy achievable in each task through oracle reordering. Our observations are as follows: (1) The sensitivity gap consistently remains substantial in the few-shot setting. (2) As performances improve, the sensitivity gap shrinks. (3) Adding more demonstrations does not necessarily results in a reduction of the gap.

4 Why Do LLMs Show Sensitivity to the Order of Options?

After analyzing instances in which reordering the options resulted in a change in LLMs prediction, we arrive at the following conjecture:

Conjecture 4.1. *The sensitivity of LLMs to the order of options in MCQ arises from the interaction of two colluding forces: (1) Uncertainty of LLMs regarding the correct answer among the top possible choices. And (2) positional bias, leading LLMs to favor specific options based on the order they appear in, depending on the question.*

In this sections, we begin by empirically validating the conjecture. Then, we identify specific patterns in the options that either amplify or mitigate the model's bias towards their placement.

4.1 Uncertainty Meets Positional Bias

To empirically validade our conjecture we devise qualitative experiments aimed at verifying each underlying reason behind the order sensitivity.

Uncertainty: We assess the uncertainty of LLMs concerning instances where reordering affects predictions through a three-step analytical approach. Let us note that GPT-4 and InstructGPT lack direct confidence measurements, necessitating our indirect analyses to validate our hypothesis.

(1) The sensitivity gap, which comprises instances where reordering changes the prediction,

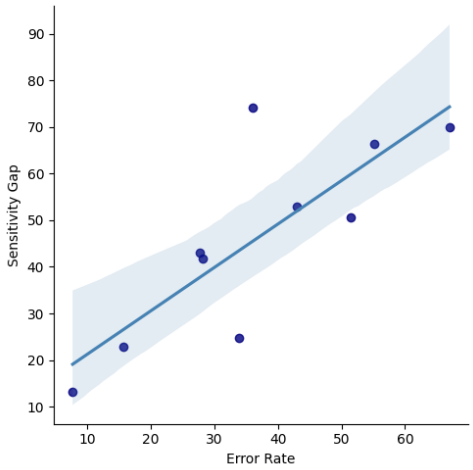

Figure 3: Correlation between the sensitivity gap and error rate for GPT-4 and InstructGPT across various MCQ tasks (each point represents the performance of an LLM on one of the benchmarks).

exhibits a strong correlation with the error rate. The correlation plot between sensitivity gap and LLMs error rate on different benchmarks is depicted in Figure 3. (2) More than 60% of the sensitive samples identified in GPT-4 also exhibit sensitivity in InstructGPT. (3) To further verify models' uncertainty towards sensitive instances, we conduct a self-verification process by posing the following question to the LLMs: "Can more than one of the choices be a highly probable answer to the question? Please respond with 'yes' or 'no'. Question: {question}. Choices: {op-

	Tasks	Sorted Options			# Options		
		Hits@1	Hits@2	Hits@3	Top-2	Top-3	All
GPT-4	CSQA	81.3	95.1	98.2	84.2	85.1	84.3
	Logical Deduction	85.3	95.7	97.9	94.8	92.3	92.3
	Abstract Algebra	55.0	72.0	88.0	57.0	52.0	57.0
	High School Chemistry	64.0	74.4	76.8	65.5	68.1	71.9
	Professional Law	51.7	62.9	74.1	65.3	65.1	66.1
InstructGPT	CSQA	63.4	82.3	90.3	70.6	72.1	72.3
	Logical Deduction	65.6	93.0	97.6	66.2	64.0	64.0
	Abstract Algebra	28.0	52.0	73.0	26.0	29.0	33.0
	High School Chemistry	30.0	51.7	66.9	37.9	40.1	44.8
	Professional Law	40.0	63.3	76.7	47.7	50.6	48.6

Table 2: Assessing the accuracy of sorting options with LLMs and analyzing the impact of reducing options complexity on models performance.

tions}. Answer:" (we provide an example prompt in Appendix). Remarkably, LLMs consistently predict "yes" for over 94% of the sensitive cases across various benchmarks, further confirming their uncertainty in these scenarios. It's worth noting that prior research highlights the ability of LLMs to accurately self-approximate and verify their knowledge and confidence (Lin et al., 2022; Kadavath et al., 2022; Weng et al., 2023). We leverage these established findings for the basis of our evaluation. We provide additional evidence regarding the impact of uncertainty on the sensitivity of LLMs by employing logprobs in the Appendix.

Positional Bias: We aim to explore the effect of positional bias in LLMs' order sensitivity by reducing sample difficulty, retaining only the top possible choices while preserving their original order of appearance, and eliminating the rest of the options. The goal is to isolate the influence of positional bias, disentangling it from other potential hidden factors impacting order sensitivity. Specifically, our objective is to examine the correlation between LLMs' predictions and the order of appearance among the top choices. This involves removing the least probable options and observing the resulting changes in LLMs' performance. Minimal changes in performance would indicate a correlation between the order of top choices and LLMs' performance. To identify the top possible choices for each question, we ask LLMs to sort the options in descending order of probability for answering the question (we provide a sample prompt in Appendix). We observe that the Hits@1 metric, which measures the accuracy of the gold truth being the first item in the sorted options, closely aligns with LLMs' overall task accuracy. Moreover, over 95% and 100% of instances that LLMs predict correctly are captured in Hits@2 and Hits@3, respectively. The results of Hits@ metrics for both GPT-4 and InstructGPT are provided in Table 2.

	Tasks	Amplify		Mitigate	
		Pattern	Ord	Pattern	Ord
GPT-4	5-option	2	AE	3	BA
	4-option	2	BD	1	AB
	3-option	2	AC	3	CB
Inst	5-option	4	EA	1	BC
	4-option	4	EA	3	CB
	3-option	4	CA	3	CB

Table 3: **Optimal patterns and their best order instantiation** for amplifying and mitigating positional bias in different LLMs based on available number of options in multiple-choice questions.

With the successful identification of the top possible choices by asking LLMs to sort the options, we proceed to investigate the impact of removing the least probable choices on the models' performance, aiming to establish the presence of positional bias. The results of retaining only the top-2 and top-3 choices after sorting the options using LLMs themselves, while preserving their original order of appearance, are presented in Table 2. We observe that despite achieving high Hits@2 and Hits@3 scores (covering all the samples where models initially predicted them correctly), LLMs' performance remains nearly unchanged or exhibits incremental improvements or declines. This observation provides further evidence of the impact of positional bias in order sensitivity.

4.2 What Patterns Amplify or Mitigate the Positional Bias?

In here we investigate the impact of certain patterns in the options on the intensity of positional bias. We categorize our findings based on number of op-

Tasks	GPT-4		InstructGPT	
	Amplifying-Bias	Mitigating-Bias	Amplifying-Bias	Mitigating-Bias
CSQA	62.9	22.7	71.7	38.3
Logical Deduction	42.0	10.1	61.7	0.9
Abstract Algebra	52.8	15.1	35.7	25.7
High School Chemistry	21.5	22.9	25.7	25.7
Professional Law	31.5	9.7	20.1	25.9

Table 4: **Percentage of initial sensitivity gap covered** using the identified patterns to amplify and mitigate positional bias. A higher percentage in amplifying bias and a lower percentage in mitigating bias indicate better performance in this context.

tions and the target large language model. We limit our investigation to the order of the top-2 choices (extracted from the sorted options list) in the options and their impact on the models' prediction to identify influential patterns. We defer further analysis of patterns involving options beyond the top-2 choices to future research.

Our goal is to identify patterns that amplify the positional bias, increasing the probability of the LLM to choose one answer over another based on their position, or mitigate the positional bias, decreasing dependency of the LLM to choose one answer over another based on their position. Upon investigating the order and placement of top-2 choices in instances where reordering changes the prediction, we discover four different patterns:

Pattern 1: First choice in top-2 appear *earlier* than the second choice in the options, and having *less* gap (less number of other choices) between them helps the goal more, i.e., to amplify or mitigate the positional bias. **Pattern 2:** First choice in top-2 appear *earlier* than the second choice in the options, and having *more* gap between them helps the goal more. **Pattern 3:** First choice in top-2 appear *later* than the second choice in the options, and having *less* gap between them helps the goal more. **Pattern 4:** First choice in top-2 appear *later* than the second choice in the options, and having *more* gap between them helps the goal more.

The best pattern, along with its best corresponding order instantiation (placement of top-2 choices), for amplifying or mitigating positional bias based on the type of LLMs and the number of options in the multiple-choice question task is presented in Table 3. For instance, to amplify the positional bias between two choices with the objective of increasing the probability of selecting the first choice as the answer for GPT-4, pattern number 2 proves to be the most effective. The ideal instantiation of this pattern is to place the first choice in option A and the second choice in option E. Investigating the positional bias in LLMs with different numbers of options in the MCQ task reveal interesting findings. In both GPT-4 and InstructGPT, the most influential pattern to amplify the bias remains the same while for mitigating bias the best pattern jumps between first and third patterns. Furthermore, there is a notable contrast between InstructGPT and GPT-4 in their reactions to patterns regarding the order of appearance in the top-2. Overall, to mitigate bias, it appears to be more effective for the top-2 choices to either appear in the first two options or in the second and third options. Conversely, for amplifying bias, it is preferable for the top-2 choices to be positioned in the first and last options.

To assess the impact of discovered patterns on LLMs' order sensitivity, we conducted two sets of experiments. Firstly, to confirm the effectiveness of patterns amplifying positional bias, we selected the best instantiation of each pattern and measured the performance improvement achieved by placing only the top-2 choice (where the ground truth is at top-1, and top-2 is obtained by sorting the options) in that instantiation. Meanwhile, we kept the order of appearance for other choices. Also, we measured the decrease in LLMs' performance by using the reverse instantiation of the pattern. Our goal here, is to assess the extent to which the sensitivity gap identified in Section 3.1 could be achieved simply by utilizing the most impactful placement. As a result, a higher percentage of coverage over the original sensitivity gap here means that the identified pattern did a better job at amplifying bias. let us note, that we do not permute the options after rearranging them based on the most effective pattern to calculate the gap. Instead, we determine the gap by subtracting the LLM accuracy for the arrangement with the highest impact from its reverse.

Secondly, to validate the patterns mitigating the bias, we performed a similar experiment as in Section 3.1, but this time, we fixed the top-2 choices in

Tasks	GPT-4		InstructGPT	
	Majority	MEC	Majority	MEC
CSQA	86.1 (+1.8)	81.2 (−3.1)	74.7 (+2.4)	67.3 (−5.0)
Logical Deduction	94.3 (+2.0)	97.4 (+5.1)	72.0 (+8.0)	57.1 (−6.9)
Abstract Algebra	57.0 (0.0)	59.0 (+2.0)	38.0 (+5.0)	31.0 (−2.0)
High School Chemistry	71.9 (0.0)	77.2 (+5.3)	45.8 (+1.0)	39.4 (−5.4)
Professional Law	67.3 (+1.2)	66.3 (+0.2)	54.3 (+5.7)	47.2 (−1.4)

Table 5: Impact of calibration methods on LLMs' performance.

the placements provided in Table 3 and reordered all other options accordingly. The goal here is to demonstrate how much of the sensitivity gap can be minimized by following identified mitigating patterns. As a result, a lower percentage of coverage over the original sensitivity gap here means that the identified pattern did a better job at mitigating bias. Since the Logical Deduction benchmark has only 3 choices there well be only one permutation after rearranging the options based on the most impactful pattern. Thus, we calculate the gap as the absolute difference between the initial performance and the performance after rearranging the options.

Table 4 presents the percentage of initial sensitivity gap covered (initial sensitivity gaps are from Table 1) by the optimal pattern for amplifying and mitigating positional bias, with more detailed results available in Appendix. A higher percentage in amplifying bias and a lower percentage in mitigating bias indicate better performance of the identified pattern. The amplifying patterns demonstrate sensitivity gap coverage ranging from 20% to 72%, while the mitigating bias pattern ranges from 0.9% to 38%. These results validate the effectiveness of the identified pattern for both amplifying and mitigating bias. Additionally, in most cases, the amplifying pattern covers a considerably greater portion of the sensitivity gap comparing to the mitigating pattern. While comparing the gap in Table 1 with the gap resulting from applying mitigation patterns may not be entirely equitable due to the significantly lower number of possible permutations, the considerably lower gap compared to amplifying patterns provides additional evidence for the impact of mitigation patterns. It is important to highlight that the patterns we have identified for amplifying bias can serve as valuable insights for enhancing model performance or launching adversarial attacks against them. Furthermore, the patterns we have established for mitigating bias can play a crucial role in shaping benchmark design and guiding annotating efforts to create less biased evaluation benchmarks for LLMs.

5 Calibrating LLMs for MCQ Tasks

We conduct an in-depth investigation into how large language models react to changes in the order of options, and investigate the reasons behind their sensitivity to such changes. Through our exploration, we have observed that LLMs are highly responsive to the sequence in which options are presented. This has led us to a critical juncture where we need to focus on methods to improve the models' resilience to variations in options order, ensuring more trustworthy evaluations.

One potential solution we have considered is the calibration of LLMs predictions. The outcomes of calibrating LLMs predictions to mitigate order sensitivity by taking majority vote over models prediction in 10 random reorders in a simple bootstrapping approach (Stickland and Murray, 2020; Hou et al., 2023), are provided in Table 5. Our analysis has unveiled a significant observation: employing a majority vote approach for evaluating LLMs results in a substantial performance improvement of up to 8 percentage points. Furthermore, while LLMs' performance on benchmarks featuring four options might be somewhat inferior to those with three or five options, GPT-4 displays a greater resilience following prediction calibration. In contrast, InstructGPT demonstrates minimal performance shift in specific contexts like CSQA and high school chemistry.

We have also incorporated the approach of Multiple Evidence Calibration (MEC) introduced by Wang et al. (2023b). In their work, they propose to counteract LLMs' sensitivity by prompting the model to generate an explanation before providing its prediction. We adopt their provided prompt for solving MCQ tasks. The impact of applying MEC calibration on MCQ tasks are outlined in Table 5.

The results from InstructGPT performance reveal that the introduction of MEC calibration results in a consistent decrease in model performance.

This behavior contradicts the outcomes achieved through majority voting and underscores the unsuitability of MEC calibration for multiple-choice question tasks. In the case of GPT-4, the integration of MEC calibration also yields contrasting outcomes with respect to majority voting, particularly evident in benchmarks such as CSQA, abstract algebra, and high school chemistry. For logical deduction and professional law benchmarks, while both majority voting and MEC calibration result in improving the model performance, the amount of improvement differs considerably, thus casting doubt on the reliability of the MEC approach in GPT-4 as well.

6 Related Work

Large language models (LLMs) show remarkable accomplishments and capabilities on various NLP tasks, including answering multiple-choice questions. In order to ascertain the dependability of LLMs' proficiency, it becomes imperative to delve into the robustness of their performance when subjected to subtle changes in the input.

LLMs and multiple-choice questions In recent years, multiple-choice questions have been introduced as an evaluation method for assessing the reasoning and fact-retrieval capabilities of models (Richardson et al., 2013; Talmor et al., 2018; Clark et al., 2020; Hendrycks et al., 2020). Despite the intricate nature of these tasks, significant strides have been made by large language models achieving human-like performances across various MCQ benchmarks (Liévin et al., 2022; Robinson et al., 2022; OpenAI, 2023; Savelka et al., 2023; Anil et al., 2023). However, the ability of these tasks to effectively gauge the reasoning and factual knowledge of LLMs, along with the reliability of the evaluation settings, presents substantial challenges that warrant deeper investigation.

Sensitivity of LLMs With the growing prominence of LLMs in addressing NLP tasks, significant attention has been devoted to examining the robustness and vulnerabilities of these models. These efforts predominantly focus on two distinct levels: (1) At the instance level, researchers investigate the robustness of LLMs by studying how modifications or adversarial attacks impact individual instances. For example, Zhao et al. (2021) reveal LLMs' sensitivity to prompt choice and demonstrations order in in-context learning (ICL). Hou et al. (2023) show LLMs are sensitive to the order of sequential interaction histories when used as conditions in ranking candidates for recommender systems. Wang et al. (2023a) launch adversarial attacks on LLM predictions through modifications to ICL demonstrations. Wang et al. (2023b) also explore LLMs' susceptibility to the order of response appearances from candidate models when LLMs serve as referees. (2) At the alignment level, attempts are made to deliberately misalign LLMs to manipulate their behavior, often referred to as "jailbreaking." Perez and Ribeiro (2022); Zou et al. (2023) achieve misalignment by adversarially attacking the prompt. In a similar vein, Wolf et al. (2023) propose a theoretical framework that exposes limitations in aligning LLMs, demonstrating there exist prompts that can cause models to exhibit any behavior with finite probability. Furthermore, Wei et al. (2023) propose that jailbreaking arises from conflicting objectives and mismatched generalization, utilizing their hypothesis to develop effective jailbreak strategies. Simultaneous with our work, Zheng et al. (2023) noted a similar sensitivity of LLMs to changes in options position within multiple-choice questions due to an inherent "selection bias." They argue that this bias manifests as a preference for specific option IDs. However, we find that by exclusively focusing on option position rather than the overall order, the observed sensitivity gap consistently remains less than 50% of demonstrated gap in our re-ordering approach (Table 1) across all benchmarks. This underscores that, in addition to selection bias, positional bias plays a crucial role in the sensitivity of LLMs in regard to MCQ tasks.

7 Conclusion

We investigate the inherent sensitivity of language models to the arrangement of options in multiple-choice questions. Upon measuring the intensity of LLMs sensitivity, our aim was twofold: to pinpoint the underlying source of this sensitivity and propose potential solutions to enhance the models' robustness. Our evaluations unequivocally reveal that LLMs not only exhibit pronounced sensitivity to options order, but also that this sensitivity diminishes only slightly when demonstrations are integrated into the few-shot setting if performance increases. In seeking to uncover the root cause of order sensitivity, we conjecture that the issue arises from LLMs' positional bias, particularly manifest-

ing in uncertain instances. We verify our conjecture by conducting diverse experiments that highlight impactful patterns that either magnify or mitigate this positional bias. Finally, to improve the robustness of LLMs' sensitivity against options order, we consider two calibration techniques leading to up to 8 percentage points improvement across different models and benchmarks.

8 Limitations

While our primary focus in this work has been on multiple-choice questions, we have also detected a parallel phenomenon—albeit with varying degrees of sensitivity—in other tasks involving multiple fragments (e.g. the options in MCQ) within inputs. This encompasses tasks like odd word detection, sorting lists of items, and ranking documents. While these observations have been noted, further exploration into these tasks is reserved for future efforts. Moreover, we provide validation for our conjecture on the reason behind LLMs' positional bias through detailed experimentation. Despite convincing outcomes, a deeper comprehension of the issue's origin necessitates a thorough exploration of the training data which is hindered by the size and accessibility of LLMs training data.

Although both calibration methods adopted in this work display promising outcomes, contributing to the improvement of model performance, they are not without their respective limitations. Majority voting is computationally expensive, while MEC diverges significantly from majority voting, casting doubts on its applicability to MCQ tasks. As a result, in order to establish a reliable and accurate evaluation framework for LLMs in the context of multiple-choice questions, it is imperative to develop more efficient calibration strategies. Moreover, refining the evaluation metrics holds the potential to improve LLMs' ability to withstand the challenges posed by options order sensitivity. These avenues present opportunities for in-depth exploration in future works. Finally, we only conduct experiments with GPT-4, InstructGPT, and Llama-2-13b over five different MCQ benchmarks. Further investigation on other LLMs and broader set of benchmarks can shed more light on the reason behind models sensitivity to the order of options and possible solutions to improve their robustness.

References

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Peter Clark, Oren Etzioni, Tushar Khot, Daniel Khashabi, Bhavana Mishra, Kyle Richardson, Ashish Sabharwal, Carissa Schoenick, Oyvind Tafjord, Niket Tandon, et al. 2020. From 'f'to 'a'on the ny regents science exams: An overview of the aristo project. *AI Magazine*, 41(4):39–53.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. 2023. Large language models are zero-shot rankers for recommender systems. *arXiv preprint arXiv:2305.08845*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Valentin Liévin, Christoffer Egeberg Hother, and Ole Winther. 2022. Can large language models reason about medical questions? *arXiv preprint arXiv:2207.08143*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334*.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Fábio Perez and Ian Ribeiro. 2022. Ignore previous prompt: Attack techniques for language models. *arXiv preprint arXiv:2211.09527*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Matthew Richardson, Christopher JC Burges, and Erin Renshaw. 2013. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 193–203.

Joshua Robinson, Christopher Michael Rytting, and David Wingate. 2022. Leveraging large language models for multiple choice question answering. *arXiv preprint arXiv:2210.12353*.

Jaromir Savelka, Arav Agarwal, Christopher Bogart, and Majd Sakr. 2023. Large language models (gpt) struggle to answer multiple-choice questions about code. *arXiv preprint arXiv:2303.08033*.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*.

Asa Cooper Stickland and Iain Murray. 2020. Diverse ensembles improve calibration. *arXiv preprint arXiv:2007.04206*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Jiongxiao Wang, Zichen Liu, Keun Hee Park, Muhao Chen, and Chaowei Xiao. 2023a. Adversarial demonstration attacks on large language models. *arXiv preprint arXiv:2305.14950*.

Peiyi Wang, Lei Li, Liang Chen, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023b. Large language models are not fair evaluators. *arXiv preprint arXiv:2305.17926*.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. 2023. Jailbroken: How does llm safety training fail? *arXiv preprint arXiv:2307.02483*.

Yixuan Weng, Minjun Zhu, Fei Xia, Bin Li, Shizhu He, Kang Liu, and Jun Zhao. 2023. Large language models are better reasoners with self-verification. *CoRR, abs/2212.09561*.

Yotam Wolf, Noam Wies, Yoav Levine, and Amnon Shashua. 2023. Fundamental limitations of alignment in large language models. *arXiv preprint arXiv:2304.11082*.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*, pages 12697–12706. PMLR.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2023. On large language models' selection bias in multi-choice questions. *arXiv preprint arXiv:2309.03882*.

Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Neil Zhenqiang Gong, Yue Zhang, et al. 2023. Promptbench: Towards evaluating the robustness of large language models on adversarial prompts. *arXiv preprint arXiv:2306.04528*.

Andy Zou, Zifan Wang, J Zico Kolter, and Matt Fredrikson. 2023. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*.

A Example Prompts

We provide example prompts for answering multiple-choice questions, self-assessing the LLM's uncertainty in answering the question, and arranging the choices from the most probable to the least probable as the answer to the question, as outlined below:

Prompt A.1: Example prompt for answering MCQ

```
Choose the answer to the question only from A, B, C, D, and E choices.
Question: Where would I not want a fox?
Choices: A) hen house B) english hunt C) mountains D) outside bedroom window E) england
Answer:
```

Prompt A.2: Example prompt for self-assessment of uncertainty

```
Can more than one of the choices be a highly probable answer to the question? Please respond with 'yes' or 'no'.
Question: Where would I not want a fox?
Choices: A) hen house B) english hunt C) mountains D) outside bedroom window E) england
Answer:
```

Prompt A.3: Example prompt for sorting the options

```
Sort the choices from the most probable to the least probable for answering the question without providing extra explanation.
Question: Where would I not want a fox?
Choices: A) hen house B) english hunt C) mountains D) outside bedroom window E) england
Answer:
```

We employed identical prompts for all LLMs, except for Llama-2-13b, where we also wrap the prompt within the necessary tags.

B Experimental Details

To measure the sensitivity gap across all benchmarks and LLMs, we exclusively consider 10 randomly chosen ordering of options. In the instance of the Logical Deduction benchmark, where only 6 ordering of options were available, we calculate the sensitivity gap over all 6 possible orders. Additionally, for the few-shot demonstrations, we randomly select 100 samples and extract the most similar demonstrations from this set using Sentence-RoBERTa (Reimers and Gurevych, 2019).

C Detailed Results

Detailed results of order sensitivity in few-shot setting are provided in Tables 7 and 8 for InstructGPT and GPT-4, respectively. Moreover, we present the impact of the identified patterns aimed at amplifying and mitigating positional bias on order sensitivity in Table 9.

D Assessing the Impact of Uncertainty on the Sensitivity of LLMs Using Logprobs

In this section, we conduct three pivotal experiments using GPT-4's logprobs to investigate the connection between model sensitivity and uncertainty. In our first experiment, we performed a t-test to compare the probabilities of the predicted choices in sensitive samples (where reordering changes the prediction) against non-sensitive samples (where reordering does not affect the prediction). We provide the resulted p-values in Table 6. The results further demonstrate the models uncertainty in sensitive samples by showing a statistically significant higher probability for non-sensitive samples.

In our second experiment, we examined the correlation between the degree of sensitivity in each sample and the probability of the predicted answer.

Tasks	**P-values**	ρ
CSQA	8.6e-10	-0.63
Logical Deduction	1.9e-10	-0.45
Abstract Algebra	6.8e-5	-0.59
High School Chemistry	2.9e-13	-0.7
Professional Law	4.0e-10	-0.62

Table 6: We investigate the impact of the uncertainty on the LLMs' sensitivity by utilising GPT-4's logprobs. We measure the p-value between the probabilities of the predicted choices in sensitive samples against non-sensitive ones. Moreover, we measure Spearman's correlation ρ between reordering entropy and the probability of the original answer.

We measure the entropy of predictions over 10 random reorders, where we calculate the probability of each prediction by dividing the number of times that answer being predicted by 10, and correlating this with the probability of the answer in the original question. The Spearman's correlation coefficients between reordering entropy and the probability of the original answer further validated this relationship across our benchmarks is provided in Table 6. We observe a significant negative correlation. This finding further suggests a profound link between sensitivity and model uncertainty.

In the third experiment, we focus on the average probability of the predicted answer based on the position of the choices using GPT-4 logprobs. Our initial observations revealed an almost uniform distribution of predicted choice positions. However, when we delved deeper and calculated the Standard Deviation for the average probability based on choice position, the results were quite interesting. For benchmarks such as CSQA, Logical Deduction, and Professional Law, we noticed an almost negligible positional bias, with the Standard Deviation hovering around 0.5%. Conversely, for Abstract Algebra and High School Chemistry, a slight preference emerged: GPT-4 marginally favored choice "C" while showing a slight disinclination towards choice "A", with a Standard Deviation of around 3%.

Tasks	1-shot			2-shot			5-shot		
	Vanila	Min	Max	Vanila	Min	Max	Vanila	Min	Max
CSQA	74.2	53.4	92.1	74.7	60.7	91.6	76.3	59.3	91.1
Logical Deduction	61.0	16.0	97.0	72.0	17.7	97.3	64.7	17.3	96.0
Abstract Algebra	36.0	3.0	72.0	38.0	9.0	73.0	34.0	7.0	73.0
High School Chemistry	50.7	19.7	89.7	52.2	21.6	84.2	52.7	24.6	83.2
Professional Law	52.1	22.3	74.7	53.3	29.3	75.3	53.7	25.5	75.2

Table 7: Few-shot order sensitivity in InstructGPT.

Tasks	1-shot			2-shot			5-shot		
	Vanila	Min	Max	Vanila	Min	Max	Vanila	Min	Max
CSQA	87.2	79.1	94.3	86.3	78.2	94.7	86.7	77.2	93.3
Logical Deduction	92.3	84.3	97.3	94.3	88.3	97.7	96.3	89.0	99.3
Abstract Algebra	58.0	31.0	82.0	60.0	32.0	79.0	58.0	34.0	78.0
High School Chemistry	72.9	49.2	91.1	67.1	53.7	90.6	68.5	47.7	91.6
Professional Law	71.2	57.7	80.7	71.7	59.7	81.3	70.6	58.4	83.6

Table 8: Few-shot order sensitivity in GPT-4.

Tasks	GPT-4				InstructGPT			
	Amplifying-Bias		Mitigating-Bias		Amplifying-Bias		Mitigating-Bias	
	Min	Max	Min	Max	Min	Max	Min	Max
CSQA	-8.0	+6.4	-4.8	+0.4	-16.0	+14.9	-7.7	+8.8
Logical Deduction	-3.1	+2.4	+1.3	+1.3	-28.4	+17.3	+0.7	+0.7
Abstract Algebra	-19.0	+9.0	-7.0	+1.0	-17.0	+8.0	-9.0	+9.0
High School Chemistry	-7.0	+2.0	-11.6	-2.0	-11.6	+5.5	-9.3	+7.8
Professional Law	-3.8	+4.0	+3.2	+5.6	-6.4	+3.7	-7.6	+5.5

Table 9: Sensitivity gap after applying the identified patterns to amplify and mitigate positional bias.