ANPMI: Assessing the True Comprehension Capabilities of LLMs for Multiple Choice Questions

Anonymous ACL submission

Abstract

Multiple-choice benchmarks, consisting of various prompts and choices, are among the most widely used methods to assess a language model's natural language understanding capability. Given a specific prompt, we typically compute $P(Choice|Prompt)$ to evaluate how likely a language model is to generate the correct choice compared to incorrect ones. However, we observe that performance measured using this approach reflects not only the model's comprehension of the prompt but also its inherent biases for certain choices regardless of the prompt. This issue makes it challenging to accurately measure a model's natural language understanding, as models may select the answer without fully understanding the prompt. To address this limitation, we propose a novel metric called **ANPMI**, which normalizes Pointwise Mutual Information (PMI) by $-\log P(Choice)$. ANPMI provides a more accurate assessment of the model's natural language understanding by ensuring that it is theoretically impossible to answer a question without properly understanding the prompt.

1 Introduction

Suppose that a man/woman answers a multiple-choice question, and the answer is correct. Could he truly solve the problem if he only looked at the options and guessed? It would not accurately reflect his ability or understanding that was intended to be assessed by the question.

A similar issue arises when we evaluate a language model. Currently, the natural language understanding capability of the model is often assessed using multiple choice questions (Achiam et al., 2023; Team et al., 2023; Jiang et al., 2024; Dubey et al., 2024). The performance of the model is measured by how frequently it selects the correct answer, based on the probability $P(Choice|Prompt)$ - the likelihood that the model will generate a given choice in response to the prompt. However, this method overlooks whether the decision is made based on a genuine understanding of the prompt, focusing solely on the model's final choice. It is similar to solving the problem by only looking at the choices without seeing the question.

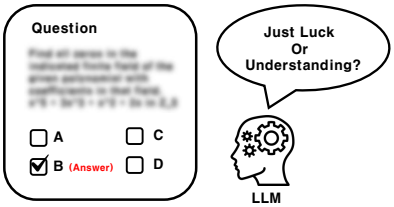


Figure 1: When a model selects an answer solely based on the choices without understanding the question, accurately assessing its comprehension of the problem becomes difficult.

The options in multiple-choice questions consist of diverse sentences, and the language model is not trained to generate these sentences with equal probabilities with a given prompt. It is a natural phenomenon for the language model, but it may lead to performance measurements that do not accurately reflect the model's understanding of the prompt. For example, the model might select the correct choice c because $P(c)$ is much higher than others, leading to overestimating the model's actual performance. Conversely, it might choose an incorrect option if the correct choice has a lower probability, leading to an underestimation of its performance.

To assess the model's actual ability to understand the given multiple-choice question and answer it correctly, it is important to equalize the generation probabilities of each answer choice. However, modifying the language model to deal with this problem is not only complicated but also sabotages the process of assessing the model's performance. Adjusting the answer choices in benchmarks is not a practical solution either, as finding suitable alternatives is challenging and could limit the diver-

sity of the tasks, restricting the evaluation of the model's ability.

Instead of relying on $P(Choice|Prompt)$, alternative methods are often used to determine the selection by the model. For example, in benchmarks like Hellaswag (Zellers et al., 2019), the model's performance is usually measured by normalizing $P(Choice|Prompt)$ based on the length of the choice (Gao et al., 2024; Zhang et al., 2024), addressing the probability imbalance caused by the varying lengths of the choices. Another approach involves calculating mutual information to measure the dependence between the choice and the prompt (Gao et al., 2024). However, these methods do not completely solve the issue stated above regarding the probability imbalance between choices.

This paper analyzes the impact of the imbalance in $P(Choice)$ on language model performance and confirms the importance of addressing the issue. We propose a method to measure model performance by normalizing the Pointwise Mutual Information (PMI) between the prompt and choice using $-\log P(Choice)$ to assess the model's actual understanding of the prompt. Our approach is theoretically unaffected by the imbalance in $P(Choice)$. Using various pre-trained models and benchmarks, we show that the proposed method more accurately evaluates the understanding of prompts by the model than existing approaches.

2 Related Work

In Deep Learning, objective functions, such as Mean Squared Error (MSE) and Cross-Entropy, are commonly optimized to train models effectively. However, these functions may not truly represent the quality of outcomes, such as the perceptual quality of generated images or a model's true language understanding capabilities. To address this issue, researchers have focused on developing diverse benchmarks (Rajpurkar et al., 2016; Sarlin et al., 2020) and evaluation metrics (Zhang et al., 2018; Ding et al., 2020; Ren et al., 2023) that better align with human judgment and application-specific needs.

In natural language processing (NLP), the most common approach to evaluate generative language models involves measuring the likelihood of generating correct answers based on specific prompts. However, this method is sensitive to the choice of prompts, which can lead to substantial outcome variations and heavily affect measured performance. As a result, many studies have investigated techniques to identify prompts that most accurately reflect a model's language understanding capabilities (Webson and Pavlick, 2021; Wei et al., 2022; Leidinger et al., 2023).

However, our observation indicates that prompt selection and answer choice design significantly influence evaluating the language model's capabilities. This paper examines how the aspects of answer choices impact the assessment of language models and proposes effective methods to address the challenge.

3 Impact of the Prior Probability

Multiple-choice questions are standard for evaluating a language model's natural language understanding. The model solves each question based on the probability $P(Choice|Prompt)$ — the likelihood of generating a particular choice $Choice$ given the prompt $Prompt$. The predicted answer is the choice with the highest probability, and the number of correctly predicted answers determines accuracy. This section explores how $P(Choice)$, the prior probability, affects model performance when calculating $P(Choice|Prompt)$. We also investigate how varying the answer choices affects the model's accuracy.

We divide $P(Choice|Prompt)$ into two components: $P(Choice)$ determined independently of the prompt $Prompt$ and $\frac{P(Choice|Prompt)}{P(Choice)}$ influenced by the prompt. This allows us to express $P(Choice|Prompt)$ as a product of the two components:

$$P(Choice|Prompt) = P(Choice) \cdot \frac{P(Choice|Prompt)}{P(Choice)}. \quad (1)$$

$P(Choice)$ represents the probability of generating a choice $Choice$ without any prompt, which we refer to as *prior probability*. On the other hand, $\frac{P(Choice|Prompt)}{P(Choice)}$ indicates how much the prompt $Prompt$ affects the probability of generating the choice $Choice$. It is equivalent to the exponential of the *Pointwise Mutual Information (PMI)*, $\log \frac{P(Choice|Prompt)}{P(Choice)}$ (Fano and Hawkins, 1961). We analyze the two components for each choice across various benchmarks to understand how the choices influence the model's final decision.

3.1 Effects of Prior Probability and PMI

We first investigate which of the two components influences the model's final decision more. We focus on two choices, C_1 with the highest value of

Table 1: The percentage of cases in which the log prior probability difference exceeds the PMI difference for each dataset. A high percentage value indicates that the model's decision is primarily driven by the prior probability difference, indicating limited influence from the prompt.

	Hellaswag	**PiQA**	**ARC-e**	**ARC-c**	**LogiQA**	**RACE**	**SciQ**	**MMLU**
OPT-125M	72.14%	78.62%	53.03%	58.62%	50.69%	55.22%	33.90%	96.39%
OPT-350M	72.13%	79.38%	51.43%	54.61%	50.23%	55.50%	29.40%	81.74%
OPT-1.3B	70.33%	76.17%	48.57%	53.67%	49.16%	56.08%	26.70%	15.23%
OPT-2.7B	69.40%	76.33%	49.41%	55.63%	49.46%	53.68%	25.30%	13.33%
OPT-6.7B	67.96%	74.65%	45.88%	52.22%	51.61%	53.30%	20.70%	29.56%
Mistral-7B	60.35%	64.25%	27.15%	37.37%	46.85%	47.94%	7.80%	13.05%
Gemma-7B	75.60%	80.63%	56.99%	56.57%	48.39%	57.61%	48.30%	14.72%
LLaMA3.1-8B	65.26%	71.82%	37.25%	44.71%	49.00%	48.80%	16.50%	15.95%

$P(Choice|Prompt)$ and C_2 with the second highest, among all choices. We compare them by calculating $\log P(C_1|Prompt) - \log P(C_2|Prompt)$. By taking the logarithm of both sides of equation (1), we express $\log P(Choice|Prompt)$ as the sum of the log prior probability and the PMI:

$$\begin{aligned}&\log P(Choice|Prompt)\\&\quad = \log P(Choice) + \log \tfrac{P(Choice|Prompt)}{P(Choice)}.\end{aligned} \tag{2}$$

Then, we calculate $\log P(C_1|Prompt) - \log P(C_2|Prompt)$ using the differences in log prior probabilities and PMIs between C_1 and C_2.

$$\begin{aligned}&\log P(C_1|Prompt) - \log P(C_2|Prompt)\\&= (\log P(C_1) - \log P(C_2))\\&\quad + \left(\log \tfrac{P(C_1|Prompt)}{P(C_1)} - \log \tfrac{P(C_2|Prompt)}{P(C_2)}\right)\\&= (\log P(C_1) - \log P(C_2))\\&\quad +(\text{PMI}(C_1, Prompt) - \text{PMI}(C_2, Prompt)).\end{aligned} \tag{3}$$

Suppose the final decision is primarily driven by differences in prior probability between the two choices. In that case, we expect the difference of the log prior probabilities to exceed that of the PMI values as follows:

$$\begin{aligned}&(\log P(C_1) - \log P(C_2))\\&\quad > (\text{PMI}(C_1, Prompt) - \text{PMI}(C_2, Prompt)).\end{aligned} \tag{4}$$

Otherwise, we expect the difference in the PMI values to be higher. To analyze whether the model's final decision is more influenced by the prior probability or exponential of PMI, we calculate the percentage of cases where the difference of the log prior probabilities exceeds the difference of the PMI values across various benchmarks. A higher percentage indicates that the model's final choice is primarily influenced by the prior probability, implying that the prompt has a limited impact on the final decision.

The experiment is performed across eight multiple-choice tasks (Welbl et al., 2017; Lai et al., 2017; Hendrycks et al., 2020; Liu et al., 2021) including Hellaswag (Zellers et al., 2019), PiQA (Bisk et al., 2020), and ARC (easy and challenge) (Clark et al., 2018) using four different language models: OPT with five different sizes(125M, 350M, 1.3B, 2.7B, and 6.7B) (Zhang et al., 2022), LLaMA3.1-8B (Dubey et al., 2024), Mistral-7B(version 0.3) (Jiang et al., 2024), and Gemma-7B (Team et al., 2024). We employ the instruction-tuned versions of LLaMA3.1, Mistral, and Gemma. The benchmarks used are briefly described in Appendix A. All results are measured under the zero-shot setting using Language Model Evaluation Harness (Gao et al., 2024). The results are summarized in Table 1.

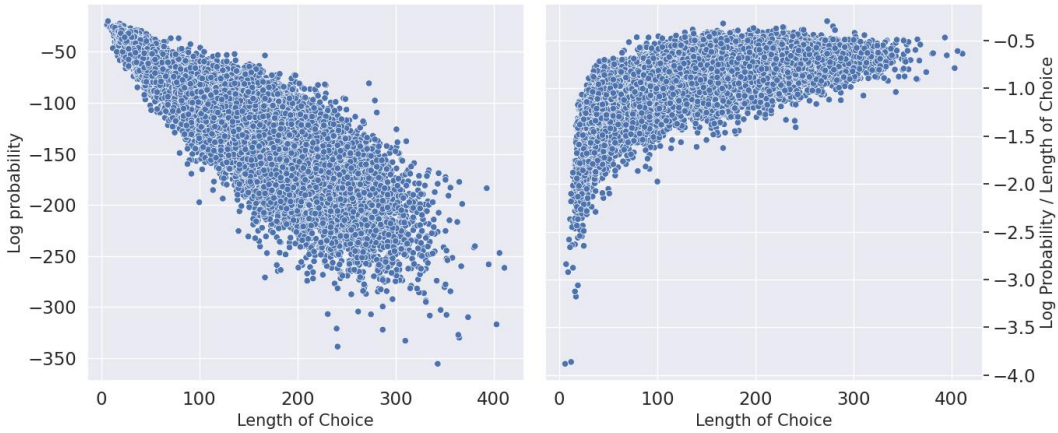
When a model lacks sufficient language understanding capability, the probabilities of choices are generated independently of the prompt, i.e., $P(Choice|Prompt) = P(Choice)$, leading to higher percentages in Table 1. For the OPT model, as the model size increases, we observe that the percentage decreases for most benchmarks in general, leading to improved language understanding. However, for the models Mistral-7B, Gemma-7B, and LLaMA3.1-8B, where instruction tuning has been applied to significantly enhance the downstream task performance (Ouyang et al., 2022; Rafailov et al., 2024), up to 80% of choices are still determined by the prior probability difference. This reinforces the assertion that, in many cases, the prior probability plays a significant role in determining the model's overall performance.

3.2 Effects of Altering Choices

To further investigate the impact of $P(Choice)$ on the model performance, we modify the choices for each problem and examine how these changes affect the model performance. To maintain the model performance as much as possible while altering the choices, we replace the choice with the lowest $P(Choice|Prompt)$ with the sentence "Hi." The

Table 2: Model performance before and after altering the choices. **Orig** refers to the performance before altering the choices, **Modified** refers to the performance after replacing the choice with the smallest $P(Choice|Prompt)$ value by "Hi."

Model	**Hellaswag**			**Arc-e**			**SciQ**		
	Orig	Modified	Orig - Modified	Orig	Modified	Orig - Modified	Orig	Modified	Orig - Modified
Mistral-7B	64.73%	3.97%	-60.76%	84.30%	71.51%	-12.79%	96.30%	95.80%	-0.50%
Gemma-7B	55.97%	3.65%	-52.32%	75.72%	61.70%	-14.02%	95.40%	93.80%	-1.60%
LLaMA3.1-8B	59.05%	3.66%	-55.39%	81.78%	65.70%	-16.08%	96.60%	96.20%	-0.40%



(a) Raw log probabilities (b) Length-normalized log probabilities

Figure 2: Comparison of log probabilities for Hellaswag choices options based on their length. We use instruction-tuned LLaMA3.1-8B.

sentence "Hi" appears frequently in various text data, resulting in a high prior probability $P(\text{"Hi"})$. However, since choices like "Hi" are unrelated to the prompt, the model's performance should remain stable if it truly relies on prompt understanding rather than $P(Choice)$ alone. If such a choice affects the model performance, this would indicate that $P(Choice)$ plays a significant role in the model's decision-making. We expect that altering a choice with a high prior probability, such as "Hi," will lead to cases where the model incorrectly selects this option over the correct one. To verify it, we perform an experiment using three instruction-tuned language models: Mistral-7B(version 0.3), Gemma-7B, and LLaMA3.1-8B, with three downstream tasks: Hellaswag, Arc-easy, and SciQ. The results are summarized in Table 2.

Table 2 shows performance decreases across all benchmarks after altering the choices. SciQ's performance drop is minimal, ranging from -0.4% to -1.6%. The log prior probability difference has less impact on performance than the PMI difference. However, for Hellaswag, where 60.35% to 75.60% of choices are determined by the prior probability difference, its performance decreases significantly, ranging from -52.32% to -60.76%. The results demonstrate that $P(Choice)$ substantially affects model performance depending on the benchmark.

4 Existing Metrics

Due to the limitations of evaluating model performance based solely on $P(Choice|Prompt)$, some benchmarks employ additional metrics. This section explores several alternative metrics commonly used in such evaluations. We explain how these metrics address the limitations of $P(Choice|Prompt)$ and discuss their constraints.

4.1 Length-Normalized Accuracy

Language models generally assign higher probabilities to shorter sentences than longer ones. It means that when there are significant differences in the lengths of the choice options, the model's answer (choice) can be biased, favoring shorter options. This results in an imbalance in $P(Choice)$ based on the length of a choice option $Choice$. To address it, length-normalized accuracy is used, which normalizes $\log P(Choice|Prompt)$ based on the text length of $Choice$. For example, the Language Model Evaluation Harness uses length normalization by dividing each choice option's log-likelihood by its

length in bytes (Gao et al., 2024). It is particularly effective for datasets, such as Hellaswag, where there are significant differences in choice lengths.

While the length-normalized accuracy addresses the problem of length imbalance and its impact on the model performance, $P(Choice)$ is not always inversely proportional to the length in bytes. Figure 2 shows the distribution of log probabilities and the length-normalized log probabilities for an instruction-tuned LLaMa3.1-8B on the choices used in Hellaswag. In Figure 2(a), we observe that the relationship between the choice length and its log probability is not linear. Consequently, the normalized log-likelihood is not constant with the text length, as shown in Figure 2(b). As a result, normalizing by length can sometimes introduce new biases, particularly when $P(Choice)$ values are already similar across options of varying lengths.

4.2 Pointwise Mutual Information (PMI)

Using mutual information (Shannon, 1948) in language modeling has a different motivation. Its goal is to measure how much the presence of a prompt increases the likelihood of a particular choice $Choice$ compared to its prior probability $P(Choice)$. Specifically, the model selects a choice option based on the Pointwise Mutual Information (PMI) value (Fano and Hawkins, 1961), $\log \frac{P(Choice|Prompt)}{P(Choice)}$. This approach counteracts the tendency of high-probability choices to dominate the selection. When $P(Choice)$ is high, indicating that the model is likely to select $Choice$ regardless of $Prompt$, PMI normalizes $P(Choice|Prompt)$ using the prior probability of $Choice$, allowing selection of less common but contextually relevant responses more often. Thus, PMI focuses on enhancing contextual relevance over raw likelihood. While less common than metrics, such as accuracy and length-normalized accuracy, PMI has been used selectively in some studies (Askell et al., 2021; Biderman et al., 2024).

The PMI value is always zero when no prompt is given, regardless of the choice. It implies that in the absence of a prompt, each choice option has an equal probability of being chosen by the model. However, when a prompt is provided, the maximum possible PMI value is $-\log P(Choice)$, as PMI reaches its peak when $P(Choice|Prompt) = 1$. As a result, each choice has a different maximum possible value based on its prior probability. When $P(Choice)$ is high, the maximum PMI value decreases, resulting in an unintended issue: choices with high $P(Choice)$ values are penalized by PMI, even if they are not inherently incorrect nor intentionally boosted. It becomes problematic when a correct $Choice$ has both a meaningfully high $P(Choice|Prompt)$, indicating relevance to the prompt, and a naturally high $P(Choice)$. This case prevents the model from selecting the correct answer simply because the answer's prior probability happens to be high.

4.3 Normalized PMI (NPMI)

PMI yields different maximum values depending on the choice. Due to this property, PMI is unsuitable for comparing different choices. To address this limitation, Normalized PMI (NPMI) (Bouma, 2009) was introduced by normalizing PMI with $-\log P(Choice, Prompt)$. NPMI normalizes PMI so that it falls within $[-1, 1]$ under the assumption that $P(Choice, Prompt) = P(Prompt, Choice)$ to allow a fair comparison.

If $P(Choice, Prompt) = P(Prompt, Choice)$, PMI satisfies the following relationship:

$$\begin{aligned} &\text{PMI}(Choice, Prompt) \\ &= \log \frac{P(Choice, Prompt)}{P(Choice)P(Prompt)} \\ &= \log \frac{P(Choice|Prompt)}{P(Choice)} \\ &= \log \frac{P(Prompt|Choice)}{P(Prompt)}. \end{aligned} \tag{5}$$

In this case, we find,

$$\log \frac{P(Choice|Prompt)}{P(Choice)} \leq -\log P(Choice).$$

and

$$\log \frac{P(Prompt|Choice)}{P(Prompt)} \leq -\log P(Prompt).$$

This means,

$$\begin{aligned} &\max(\text{PMI}(Choice, Prompt)) \\ &= \min(-\log P(Choice), -\log P(Prompt)). \end{aligned}$$

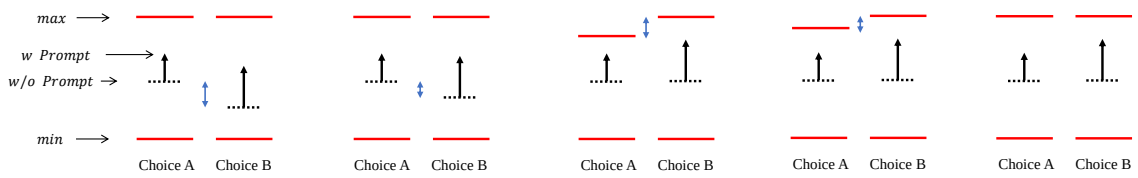
Normalization by $-\log P(Choice, Prompt)$ ensures the maximum value 1 because,

$$-\log P(Choice) < -\log P(Choice, Prompt)$$

and

$$-\log P(Prompt) < -\log P(Choice, Prompt).$$

However, in the case of language models, $P(Choice, Prompt)$ represents the probability of generating the sentence $Prompt + Choice$, which results in $P(Choice, Prompt) \neq P(Prompt, Choice)$. Thus, PMI_{LM} in language models satisfies the following relationship where x is $Choice$, and y is $Prompt$:



(a) *P(Choice|Prompt)* (b) Length Normalized *P(Choice|Prompt)* (c) PMI (d) NPMI (e) ANPMI

Figure 3: Comparison of the existing metrics and ANPMI for two different choice options, A and B. Dotted lines indicate values calculated without a prompt, while the heads of block arrows represent values after a prompt is provided. The red lines denote the theoretical minimum and maximum values, and the blue arrows highlight the difference caused by the two different choices. The value of $P(Choice|Prompt)$ differs depending on choices when no prompt is given. Furthermore, PMI has a different theoretical maximum value depending on the choice. Normalizing by the length and NPMI mitigates this difference but does not eliminate it due to their incorrect assumptions. ANMPI, on the other hand, always has the same value for all cases independent of the prompt.

$$\text{PMI}_{\text{LM}}(x, y) = \log \frac{P(x|y)}{P(x)} \neq \log \frac{P(y|x)}{P(y)}. \quad (6)$$

Thus, NPMI is not an appropriate normalization method for PMI in language models. As a result, NPMI is treated as nonstandard in language model evaluation and is not commonly used.

5 The Proposed Metric, ANPMI

We observed that due to the imbalance in $P(Choice)$, accurately assessing a model's language comprehension ability in multiple-choice tasks is challenging. While carefully constructing answer choices could address this issue, designing choices that prevent $P(Choice)$ imbalance across all language models is impractical. Thus, we propose a normalized PMI metric, *Asymmetric NPMI (ANPMI)* to evaluate the model performance in multiple-choice tasks. It is defined as follows:

$$\text{ANPMI}(Choice, Prompt) = \frac{\text{PMI}(Choice, Prompt)}{-\log P(Choice)}. \quad (7)$$

It mitigates the influence of the $P(Choice)$ imbalance, offering a more reliable indicator of a model's understanding of the prompt.

Ideally, the following requirements should be met by an assessment metric to measure a model's true language comprehension capability:

- In the absence of a prompt, the model should assign equal probabilities to each choice option of a question, indicating that the prompt is essential for answering it.
- The maximum and minimum values for the metric for selecting choices should remain consistent across the choices. A fair comparison between choices becomes difficult if a certain choice yields a disproportionately high or low value.

Theoretically, PMI meets the first requirement for accurately assessing a model's language comprehension ability. However, it does not satisfy the second requirement (discussed in Section 4.2). PMI may yield different maximum values depending on the choice.

NPMI normalizes PMI under the assumption that $P(Choice, Prompt) = P(Prompt, Choice)$. However, this assumption does not hold in the language model, making it an unsuitable normalization method.

To overcome this, we propose ANPMI (Asymmetric NPMI) that normalizes PMI by $-\log P(Choice)$ for the evaluation metric. It yields a value from 1 to $-\infty$ regardless of the choice, thus satisfying the second requirement for fair and accurate evaluation. Figure 3 illustrates how these characteristics distinguish ANPMI from existing metrics.

Unlike NPMI, which normalizes PMI by $-\log P(Choice, Prompt)$, ANPMI normalizes PMI by $-\log P(Choice)$ to consider the inherent asymmetry in $P(Choice, Prompt)$ when computed with a language model.

From Equation 6, we find,

$$\text{PMI}_{LM}(Choice, Prompt) = \log \frac{P(Choice|Prompt)}{P(Choice)} \leq -\log P(Choice). \quad (8)$$

Thus, the maximum of PMI in a language model is $-\log P(Choice)$, not $-\log P(Choice, Prompt)$, which is why ANPMI normalizes PMI using $-\log P(Choice)$ to account for the asymmetry.

Table 3: Model performance when no prompt is provided.

Metric	Model	Hellaswag	PiQA	ARC-e	ARC-c	LogiQA	RACE	SciQ	MMLU
Random	label 0	25.04%	49.51%	25.08%	22.70%	20.08%	25.93%	0.00%	22.95%
	label 1	24.75%	50.49%	24.62%	26.54%	24.42%	24.78%	0.00%	24.65%
	label 2	25.73%	-	26.64%	26.45%	27.50%	25.93%	0.00%	25.51%
	label 3	24.48%	-	23.61%	24.32%	27.80%	23.35%	100%	26.89%
Acc	Mistral-7B	46.22%	71.65%	35.06%	22.70%	19.35%	23.92%	27.50%	22.95%
	Gemma-7B	40.79%	67.79%	33.00%	23.72%	19.66%	25.55%	24.60%	22.95%
	LLaMA3.1-8B	43.24%	71.60%	35.23%	24.06%	19.35%	24.21%	27.50%	22.95%
Acc_{Norm}	Mistral-7B	59.06%	72.09%	32.45%	30.12%	24.42%	29.79%	31.90%	22.95%
	Gemma-7B	29.37%	57.24%	27.15%	28.24%	30.26%	29.09%	26.10%	22.95%
	LLaMA3.1-8B	54.74%	71.82%	33.84%	28.75%	24.88%	29.00%	32.30%	22.95%
Acc_{PMI}	Mistral-7B	25.04%	49.51%	25.08%	22.70%	20.28%	25.93%	0.00%	22.95%
	Gemma-7B	25.04%	49.51%	25.08%	22.70%	20.28%	25.93%	0.00%	22.95%
	LLaMA3.1-8B	25.04%	49.51%	25.08%	22.70%	20.28%	25.93%	0.00%	22.95%
$\text{Acc}_{\text{ANPMI}}$	Mistral-7B	25.04%	49.51%	25.08%	22.70%	20.28%	25.93%	0.00%	22.95%
	Gemma-7B	25.04%	49.51%	25.08%	22.70%	20.28%	25.93%	0.00%	22.95%
	LLaMA3.1-8B	25.04%	49.51%	25.08%	22.70%	20.28%	25.93%	0.00%	22.95%

Table 4: Zero-shot model performance measured with various metrics. Bold numbers represent the best performance for each model and each benchmark.

Metric	Model	Hellaswag	PiQA	ARC-e	ARC-c	LogiQA	RACE	SciQ	MMLU
Acc	Mistral-7B	64.73%	81.56%	84.30%	57.51%	32.72%	46.70%	96.30%	59.72%
	Gemma-7B	55.97%	76.61%	**75.72%**	47.53%	24.88%	41.34%	95.40%	50.27%
	LLaMA3.1-8B	59.05%	80.09%	**81.78%**	51.28%	31.64%	44.31%	96.60%	67.70%
Acc_{Norm}	Mistral-7B	**82.91%**	**82.64%**	82.87%	58.79%	33.79%	47.27%	94.50%	59.72%
	Gemma-7B	**73.10%**	**77.91%**	72.69%	**48.81%**	29.19%	**43.92%**	91.80%	50.27%
	LLaMA3.1-8B	**79.25%**	**81.01%**	79.55%	54.95%	31.95%	46.70%	96.10%	67.70%
Acc_{PMI}	Mistral-7B	69.44%	73.56%	80.51%	62.54%	32.10%	47.46%	96.00%	**60.00%**
	Gemma-7B	54.15%	66.76%	61.20%	46.67%	**30.41%**	40.29%	84.50%	50.40%
	LLaMA3.1-8B	62.33%	68.61%	68.14%	55.38%	33.64%	44.69%	92.20%	66.32%
$\text{Acc}_{\text{ANPMI}}$	Mistral-7B	77.67%	77.58%	**85.90%**	**63.99%**	**34.10%**	**51.20%**	**96.90%**	59.91%
	Gemma-7B	57.77%	76.55%	75.34%	47.78%	25.81%	42.11%	**95.50%**	**50.41%**
	LLaMA3.1-8B	73.73%	77.69%	80.98%	**57.85%**	**34.25%**	**48.13%**	**97.40%**	**67.79%**

6 Experiments

In this section, we evaluate the performance of the models using ANPMI, while comparing it with the existing metrics. Specifically, we conduct experiments using instruction-tuned language models, such as Mistral-7B(version 0.3) (Jiang et al., 2024), Gemma-7B (Team et al., 2024), and LLaMA3.1-8B (Dubey et al., 2024), along with seven widely used multiple-choice benchmarks (Zellers et al., 2019; Bisk et al., 2020; Clark et al., 2018; Hendrycks et al., 2020; Welbl et al., 2017; Liu et al., 2021; Lai et al., 2017). We aim to highlight the differences between ANPMI and other popular existing metrics, demonstrating both their benefits and limitations through empirical analysis. The model performance is denoted as Acc, Acc_{Norm}, Acc_{PMI}, and Acc_{ANPMI} when measured using $P(Choice|Prompt)$, length-normalized $P(Choice|Prompt)$, PMI, and ANPMI. **Random** represents the baseline performance, reflecting the probability of selecting the correct label between labels 0, 1, 2, and 3 by chance, based solely on the label distribution. We exclude NPMI because it is not standard, and it is impossible to compute $P(Choice, Prompt)$ if $Prompt + Choice$ is larger than the maximum sequence length.

6.1 Performance When No Prompt Provided

To verify that ANPMI can evaluate performance independently of the differences in $P(Choice)$, we measure the language model performance on the various benchmarks without providing prompts. The results of these evaluations are summarized in Table 3.

For MMLU, we observe identical performance across all models, regardless of the metric used. This is because the same four choices — A, B, C, and D — are given throughout examples. However, for other datasets, such as Hellaswag and ARC, which have a set of different answer choices for each example, model performance varies when evaluated using Acc or Acc_{Norm}. For each benchmark, we observe a difference of up to 30% in performance between models when evaluated using these metrics. Section 3.1 demonstrates that variations in $P(Choice)$ significantly influence a model's

Table 5: The proportion of choices selected in the MMLU task based on PMI and ANPMI metrics for the LLaMA3.1-8B model.

	Choices			
	A	B	C	D
$\log(P(Choice))$	-9.14	-10.08	-10.27	-9.95
PMI	12.36%	31.65%	33.31%	22.69%
ANPMI	18.01%	30.10%	29.66%	23.23%

final decisions. Thus, these performance differences observed without prompts, which highlight the impact of prior probabilities, may complicate accurately ranking models. Moreover, the measured performance for Hellaswag, PiQA, and ARC-easy is significantly higher than that of random guessing. This indicates that when using Acc or Acc_{Norm}, models may achieve high scores on these benchmarks without understanding the prompts, complicating the evaluation of their language comprehension capability.

In contrast, PMI and ANPMI have identical performance across all models when prompts are absent. These metrics always assign a zero value when prompts are not provided, resulting in consistent performance measurements by always choosing the same choice. Consequently, PMI and ANPMI effectively eliminate the influence of $P(Choice)$ on performance, making them reliable metrics for accurately assessing a model's understanding of prompts to answer questions.

6.2 Comparison of the Metrics

The results of evaluating the model performance using various metrics are summarized in Table 4. The experiments are conducted using the Language Model Evaluation Harness (Gao et al., 2024) under a zero-shot setting.

Benchmarks where the final decision of the model depends heavily on $P(Choice)$ show a larger performance gap when measured using metrics other than $P(Choice|Prompt)$. For instance, when evaluating HellaSwag using LLaMA3.1-8B, about 65% of decisions are influenced by the differences in $P(Choice)$ as seen in Table 1, resulting in a 14.68% performance gap between Acc and $\text{Acc}_{\text{ANMPI}}$. Conversely, in MMLU, where only 13% to 16% of decisions of each model depend on the $P(Choice)$ difference according to Table 1, the maximum performance discrepancy is merely up to 0.19% comparing Acc and $\text{Acc}_{\text{ANPMI}}$.

The difference between Length-normalized $\log P(Choice|Prompt)$ and ANPMI can be observed on MMLU. Since all choices in MMLU have the same length in bytes (1 byte), Acc_{Norm} is identical to Acc, with no performance change occurring due to length normalization. In contrast, ANPMI theoretically addresses the impact of the imbalance in $P(Choice)$ on model performance measurement. As a result, differences between Acc and $\text{Acc}_{\text{ANPMI}}$ are consistently observed across all models.

The difference in model performance measured by PMI and ANPMI is caused by the fact that PMI does not perform any normalization. Table 5 shows how the lack of normalization affects the model's final choices in MMLU. PMI tends to assign smaller maximum values to choices with higher $\log P(Choice)$, making the model less likely to select options with large $P(Choice)$ values. As demonstrated in Table 5, under PMI, choice A (A has the highest $\log P(Choice)$) is the least frequently chosen, whereas choice C (C has the lowest $\log P(Choice)$) is the most frequently chosen. In contrast, this tendency is less evident when using ANPMI.

The experimental results indicate that when model performance is evaluated using a metric that fails to account for the $P(Choice)$ imbalance, the model's performance does not accurately reflect its natural language understanding capability. As a result, ANPMI, which theoretically addresses the $P(Choice)$ imbalance, is identified as the most appropriate metric for assessing a language model's natural language understanding capability.

7 Conclusion

This paper introduces ANPMI, a novel metric for assessing natural language understanding in language models for multiple-choice tasks. It ensures that the model performance reflects the true comprehension capability of the model rather than unrelated choice preferences. ANPMI is defined by normalizing PMI with $-\log P(Choice)$. All choices yield an identical score without a prompt under ANPMI, requiring the model to understand the prompt to solve the task. Unlike PMI, ANPMI maintains the same maximum and minimum values across all choices, eliminating bias towards any specific choice and focusing solely on the relationship between the prompt and choices. Through evaluations using diverse language models and benchmarks, we demonstrate that ANPMI effectively addresses the issue of inaccurate performance measurement caused by imbalances in $P(Choice)$.

Limitations

While some benchmarks for evaluating language model performance, such as HumanEval (Chen et al., 2021) and IFEval (Zhou et al., 2023), are not in multiple-choice format, this study focuses exclusively on multiple-choice benchmarks. Additionally, although the structure of prompts used in evaluations significantly impacts model performance, our analysis is limited to the effects of choice construction. In the future, we plan to address cases not covered in this study to ensure accurate performance measurement and fair comparisons across models.

Ethics Statement

Our research adheres to rigorous ethical standards while contributing to the advancement of NLP. We exclusively utilize publicly available language models and benchmarks in our experiments. The datasets employed in our study—HellaSwag (MIT), PiQA (AFL), ARC (CC-BY-SA 4.0), LogiQA (CC-BY-NC-SA 4.0), RACE (AFL), SciQ (CC-BY-NC 3.0), and MMLU (MIT)—are all permitted for academic use. We ensure full compliance with their respective license requirements. Furthermore, while our research presents evaluation results across various models, it contains no information that could harm individuals or groups.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, et al. 2021. A general language assistant as a laboratory for alignment. *arXiv preprint arXiv:2112.00861*.

Stella Biderman, Hailey Schoelkopf, Lintang Sutawika, Leo Gao, Jonathan Tow, Baber Abbasi, Alham Fikri Aji, Pawan Sasanka Ammanamanchi, Sidney Black, Jordan Clive, et al. 2024. Lessons from the trenches on reproducible evaluation of language models. *arXiv preprint arXiv:2405.14782*.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Gerlof Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, 30:31–40.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P Simoncelli. 2020. Image quality assessment: Unifying structure and texture similarity. *IEEE transactions on pattern analysis and machine intelligence*, 44(5):2567–2581.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Robert M Fano and David Hawkins. 1961. Transmission of information: A statistical theory of communications. *American Journal of Physics*, 29(11):793–794.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024. A framework for few-shot language model evaluation.

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*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794.

Alina Leidinger, Robert Van Rooij, and Ekaterina Shutova. 2023. The language of prompting: What linguistic properties make a prompt successful? *arXiv preprint arXiv:2311.01967*.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2021. Logiqa: a challenge dataset for machine reading comprehension with logical reasoning. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 3622–3628.

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.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *Preprint*, arXiv:1606.05250.

Liliang Ren, Mankeerat Sidhu, Qi Zeng, Revanth Gangi Reddy, Heng Ji, and ChengXiang Zhai. 2023. C-pmi: Conditional pointwise mutual information for turn-level dialogue evaluation. *arXiv preprint arXiv:2306.15245*.

Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. 2020. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947.

Claude Elwood Shannon. 1948. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Albert Webson and Ellie Pavlick. 2021. Do prompt-based models really understand the meaning of their prompts? *arXiv preprint arXiv:2109.01247*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Johannes Welbl, Nelson F Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 94–106.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. 2024. Tinyllama: An open-source small language model. *arXiv preprint arXiv:2401.02385*.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.

A Benchmarks used in Experiments

In this paper, we perform experiments using seven multiple-choice benchmarks. The experiments are conducted with the Language Model Evaluation Harness (Gao et al., 2024), and we follow the prompt and choice structures outlined by this library. Below, we provide detailed descriptions, evaluation templates, and examples of the benchmarks. Each template and example uses a monospaced font to indicate parts that vary between examples.

A.1 Hellaswag

HellaSwag (Zellers et al., 2019) is a benchmark for evaluating commonsense natural language inference (NLI). The task involves selecting the most appropriate continuation of a given sentence. We use the validation set, which consists of 10,042 examples, for our experiment.

[Template]
Prompt: `activity_label: ctx_a ctx_b`
Choices: [`endings1`, `endings2`, `endings3`, `endings4`]

[Example]
Prompt: `Clean and jerk: A lady walks to a barbell. She bends down and grabs the pole. The lady`
Choices: [
`swings and lands in her arms.`,
`pulls the barbell forward.`,
`pulls a rope attached to the barbell.`,
`stands and lifts the weight over her head.`
]

A.2 PiQA

Physical Interaction: Question Answering(PiQA) (Bisk et al., 2020) is a benchmark to evaluate whether a model can answer questions based on physical commonsense knowledge. PiQA focuses on everyday situations with a preference for atypical solutions, and each question has two options. The validation set used for our evaluation consists of 1,838 questions.

[Template]
Prompt: Question: `goal`
Answer:
Choices: [`sol1`, `sol2`]

[Example]
Prompt: Question: `To fight Ivan Drago in Rocky for sega master system.`
Answer:
Choices: [
`Drago isn't in this game because it was released before Rocky IV.`,
`You have to defeat Apollo Creed and Clubber Lang first.`
]

A.3 ARC

The AI2 Reasoning Challenge(ARC) (Clark et al., 2018) comprises science questions and answers targeted at students from grade 3 to grade 9. It is divided into two difficulty levels: *easy* and *challenge*. For model evaluation, we use the test sets for both difficulty levels. The ARC-Easy test set includes 2,376 questions, while the ARC-Challenge test set contains 1,172 questions.

[Template]
Prompt: Question: `question`
Answer:
Choices: [`choices1`, `choices2`, `choices3`, `choices4`]

[Example]
Prompt: Question: `Which piece of safety equipment is used to keep mold spores from entering the respiratory system?`
Answer:
Choices: [`safety goggles`, `breathing mask`, `rubber gloves`, `lead apron`]

A.4 LogiQA

Logical Reasoning Question Answering(LogiQA) (Liu et al., 2021) is a benchmark designed to assess a model's logical reasoning abilities. It consists of expert-written questions that cover multiple types of deductive reasoning. In our experiments, we use a test set of 651 problems.

[Template]
Prompt: Passage: `context`
Question: `question`
Choices:
A: `option1`
B: `option2`
C: `option3`
D: `option4`
Answer:
Choices: [`option1`, `option2`, `option3`, `option4`]

[Example]
Prompt: Passage: `There are five teams participating in the game. The audience had the following comments on the results? (1) The champion is either the Shannan team or the Jiangbei team. (2) The champion is neither Shanbei nor Jiangnan. (3) The champion is Jiangnan Team. (4) The champion is not the Shannan team.`
Question: `The result of the match showed that only one argument was correct, so who won the championship?`
Choices:
A. `Shannan`
B. `Jiangnan`
C. `Shanbei`
D. `Jiangbei`
Answer:
Choices: [`Shannan`, `Jiangnan`, `Shanbei`, `Jiangbei`]

A.5 RACE

ReAding Comprehension dataset from Examinations(RACE) (Lai et al., 2017) is an English reading comprehension dataset derived from China's middle and high school English exam questions. Each question comprises an article followed by several questions and answer choices. For our evaluation, the test set contains 1,045 questions. We include all but the final question from each set in the prompt, ensuring that most of the context is part of the model's input.

[Template]
Prompt: Article: `article`
Question: `problem1`
Answer: `answer1`
...
Question: `problem_last`
Choices: [`option1`, `option2`, `option3`, `option4`]

[Example]
Prompt: Article: `A girl with blue eyes is a blue-eyed girl. ... There are sound-proof rooms in all broadcasting stations.`
Question: `The clothes which you buy from the supermarket are called _ clothes.`
Answer: `ready-made`
`...`
`What do you think is the best title for the article?`
Choices: [
`The Forms of Compound Words.`,
`Compound Words in Everyday Life,`
`How to Use Compound Words.`,
`Water-proof Cloth in the Best.`
]

A.6 SciQ

Scientific Question Answering(SciQ) (Welbl et al., 2017) is a dataset of science exam questions crowdsourced across domains such as Physics, Chemistry, and Biology. Each question includes a question, answer choices, and a paragraph of supporting information to assist reasoning. For our evaluation, we use a test set comprising 1,000 questions.

[Template]
Prompt: `support`
Question: `question`
Answer:
Choices: [`distractor1`, `distractor2`, `distractor3`, `correct_answer`]

[Example]
Prompt: `Tree rings, ice cores, and varves indicate the environmental conditions at the time they were made.`
Question: Ice cores, varves and what else indicate the environmental conditions at the time of their creation?
Answer:
Choices: [`mountain ranges`, `fossils`, `magma`, `tree rings`]

A.7 MMLU

Massive Multitask Language Understanding(MMLU) (Hendrycks et al., 2020) evaluates a model's breadth and depth of knowledge across various domains. The dataset covers 57 topics, including STEM, humanities, and social sciences. Our experiments use the comprehensive test set, which contains 14,042 questions. Each multiple-choice question assesses the model's ability to integrate diverse knowledge.

[Template]
Prompt: `question`
A. `choice1`
B. `choice2`
C. `choice3`
D. `choice4`
Answer:
Choices: [A, B, C, D]

[Example]
Prompt: `The following are multiple choice questions (with answers) about astronomy.`
`What is the second most common element in the solar system?`
A. `Iron`
B. `Hydrogen`
C. `Methane`
D. `Helium`
Answer:
Choices: [A, B, C, D]