Improving Score Reliability of Multiple Choice Benchmarks with Consistency Evaluation and Altered Answer Choices

Anonymous ACL submission

Abstract

In this work we present the Consistency-Rebalanced Accuracy (CoRA) metric, improving the reliability of Large Language Model (LLM) scores computed on multiple choice (MC) benchmarks. Our metric explores the response consistency of the LLMs, taking advantage of synthetically-generated questions with altered answer choices. With two intermediate scores, i.e. Bare-Minimum-Consistency Accuracy (BMCA) and Consistency Index (CI), CoRA is computed by adjusting the multiple-choice question answering (MCQA) scores to better reflect the level of consistency of the LLM. We present evaluations in different benchmarks using diverse LLMs, and not only demonstrate that LLMs can present low response consistency even when they present high MCQA scores, but also that CoRA can successfully scale down the scores of inconsistent models.

1 Introduction

Despite the current popularity of Large Language Models (LLMs), and the undeniable capabilities that they have demonstrated to solve very complex real-world problems, it is also the real truth that there is yet a lot to be done in terms of understanding and measuring precisely their capabilities and risks to deploy reliable and liable applications.

The most used approach to evaluate LLMs is to measure its performance on question-answering (QA) benchmark datasets (or simply benchmarks), i.e. datasets containing questions (aka prompts) with their respective expected outputs, where the outputs generated by the LLM are compared to the expected outputs from the benchmark, resulting in univariate scores that are used to rank and evaluate the LLMs. A common way to structure QA benchmarks is to rely on multiple choice (MC) questions, which is not only a widely adopted method to evaluate human for several knowledge-testing objectives, but also has the advantage of being a very simple way to compute right and wrong answers.

Although the research community has been highly active in investigating the limitations of current benchmarking practices, and several issues have already been identified for MC evaluations, such as choice biases, variability to rewordings, inconsistent confidence, among others (Zheng et al., 2024; Reif and Schwartz, 2024; Ye et al., 2024), we believe that there is still a gap in better quantifying the capabilities of an LLM. Given that the most used method to evaluate LLM on MC benchmarks is to compute the ratio of matches between the responses of the LLM against the correct alternatives, an approach that we refer to MCQA, we think that this approach is lacking in providing a realistic and informative evaluation whether the LLM is actually knowledgeable about the test questions, or if that the scores are a by-product of issues such as training data contamination or random guesses given the stochastic nature of inference algorithms.

In this work we argue that computing response consistency is key to have metrics that are able to present a more reliable score for the evaluation of LLMs on MC benchmarks. As already demonstrated, LLMs can suffer from inconsistencies when subjected to variations in the input, especially when the set of presented alternatives is slightly modified with reorderings or changes in the set of distractors (Pezeshkpour and Hruschka, 2024; Wang et al., 2025). Notice that distractors consists of the alternatives in a MC question that are not correct, so usually a MC question is composed with a question and a set of choices, where there is one correct choice[1] and one or more distractors. Thus, it is quite easy to synthetically generated altered questions by playing with the set of distractors, while keeping the correct choice.

[1] Although it is possible to have more than correct choice in a MC question, we delimit the scope of this work for cases with only one correct alternative.

Based on generating altered sets of questions with modified distractors (or simple reorderings) in the choices, a set to which we refer as the divergent questions, we propose the Consistency-Rebalanced Accuracy (CoRA) metric to better reflect the level of consistency of the LLM on the MC benchmark. The metric is based on two intermediate metrics, i.e. Bare-Minimum-Consistency Accuracy (BMCA) and Consistency Index (CI), where the first is used to compute the accuracy according to a specified minimum level of consistency, and the second computes the gap of accuracy between the score on the original benchmark, i.e. the MCQA method, and the BMCA(1.0), the accuracy for 100% consistency. The CoRA score is then computed by scaling the related MCQA score with the value computed with CI.

We evaluate CoRA in different popular benchmarks, with both open source and a commercial LLM, and observe that CoRA tends to reflect more realistic distribution of scores according to the consistency level of the LLMs. That is, with BMCA evaluated with different levels of minimum consistency, we observe that some top-performing LLMs present a drastic decrease of accuracy, and reach very poor performances with BMCA(1.0), indicating that the scores with MCQA and the other baselines are not reliably reflecting the consistency of the LLM. Consequently, scaling down that score with the consistency index CI results in a more faithful measurement of the capabilities of the LLM: the CoRA metric.

We believe that this paper not only contributes to improve the evaluation of LLMs in MC benchmarks, but also in emphasizing that the use of response consistency evaluation is a viable and necessary approach to provide more faithful benchmark evaluation scores. We show an LLM such as GPT4o can present a drop of at least 0.10 points in accuracy, comparing MCQA with BMCA(1.0), showing that even this top-performing LLM can be 'unsure' about the correct answer for about 14% of the correct response. More concerning is that models that perform close to GPT4o in MCQA score, such as MedLlama3, can present very low consistency levels, making the CoRA score to be less than half of the original MCQA score. In our opinion, it is mandatory to include consistency evaluation before releasing any score computed on an MC benchmark.

In order to make our research accessible by the community, we are publicly releasing the source code for computing CoRA scores: http://anonymous4now.github.com.

2 Related Work

Understanding well the capabilities of LLMs is key for deploying safe, liable, end-user applications, and several efforts have been made towards improving the evaluation of such models (Lin and Chen, 2023; Wang et al., 2024b; Lei et al., 2024). Although we can see some works focusing on the evaluation of open-end questions (Myrzakhan et al., 2024), multiple-choice (MC) evaluation is a common practice for mainstream models (Singhal et al., 2023; Jiang et al., 2023; Nori et al., 2023; Dubey and et al, 2024). Multiple-choice questions can be more objectively evaluated as opposed to open questions, the evaluation for which can be difficult and subjective even for human evaluators. Furthermore, MC evaluation is a widely-used practice to evaluate proficiency of humans in several areas, for instance medical and law domains (Lesage et al., 2013; Curtis et al., 2013; Grazziotin-Soares et al., 2021). It is thus natural to rely on a similar evaluation process to measure the proficiency of LLMs.

It is well known, though, that there is room to make MC evaluation more reliable and believable. Some efforts have been made in trying to understand the limitations of MC evaluation focused on confidence levels, either considering the logits of the neural networks or self-confidence scores provides by the LLM itself (Ye et al., 2024). In (Wiegreffe et al., 2024), an analysis on how the weights of transformers react to predict a correct answer is presented. The correlation between model confidence (probability outputs) and model self-confidence (a confidence level expressed by the model) have explored in (Kumar et al., 2024), which show that the LLMs usually present low to moderate correlation. Others have focused on identifying possible biases that can be exhibited by the LLM, such as selection, token, and label biases (Zheng et al., 2024; Reif and Schwartz, 2024).

Another group of researchers focused on understanding the sensibility of the LLM according to changes in the input. In (Mirzadeh et al., 2024), the authors show that LLMs are negatively impacted by changes in the input question. In this case, the model performs significantly worse when only the numbers are changed in the input for math-related questions. Assuming that LLMs can be affect by changes in the input, the work presented in (Acker-

man et al., 2024) proposes a metric for computing the robustness of LLMs to input changes, considering perturbations in the input and reporting the impact in the accuracy. Intriguing results were reported in (Balepur et al., 2024), where the authors query LLMs only with choices without the question was investigated, and show that even without the questions the LLMs can correctly answer a considerable number of questions. The authors looked for memorization but could not fully explain the phenomenon.

One particular line of research focuses in investigating the *consistency* of LLMs in providing a response when the question is kept intact but with variations in other factors, such as the set of choices and parameters of the inference algorithm. An investigation on the sensitivity of choice order is presented (Li et al., 2024), along with an exploration on the consistency of LLM according to different values for the temperature parameter, but they show that models such as GPT-3.5 tend to present high-consistency when prompted with different temperature values. In (Wei et al., 2024), the authors compare the results of MC evaluation and open-ended answer, and find low consistency between these two evaluations. However, both works presented in (Pezeshkpour and Hruschka, 2024) and (Wang et al., 2024a) explore changes in the set of choices, either with simple reorderings or by modifying the set of distractors, and provide convincing evidence that LLMs that perform well is some MC benchmark are prone to lack of consistency when facing questions with modified distractors. As a consequence, two new metrics exploiting the consistency of LLM, namely MCQA+ and MV (see Section 3), have been proposed.

This work is heavily inspired by the results on the consistency of LLM when the sets of choices are modified. Our main contribution is on improving the robustness of metrics to more faithfully take into account the consistency level of a given LLM on a MC benchmark and express that into a score.

3 Baselines

In this section we will revisit how the accuracy score is computed for MC benchmarks. We will first describe how this is done in the most traditional method, i.e. to perform single-run evaluations and compute the ratio of hits. Next, we will also describe existing methods that explore divergent sets of answers to enhance the computation of such scores.

Let $MCQ = \{mcq_1, \ldots, mcq_N\}$ be the original set of N questions, choice, and answers of a MC benchmark. Consider also that there is a function denoted $llm(mcq_i)$ that returns 1 if a given LLM presents the correct response, i.e. the response provided by the LLM is equal to the correct alternative in mcq_i, or 0 otherwise. The most used baseline consists of computing the accuracy directly on MCQ, which we refer to as MCQA and define as:

$$\text{MCQA} = \frac{1}{N}\sum_{i=1}^{N} llm(mcq_i) \quad (1)$$

But as we presented in the previous section, LLMs can be inconsistent even we very simple test, and it is important to take such aspect into account during the evaluation process. As a consequence, consider also that there is a set denoted $MCQ* = \{\hat{MCQ}*_1, \ldots, \hat{MCQ}*_N\}$, comprising N divergence sets which are derivations of the samples in the MCQ set. The divergence sets can be created using M different methods, so that $\hat{MCQ}*_i = \{mcq*_i^1, \ldots, mcq*_i^M\}$ and $mcq_i \in \hat{MCQ}*_i$, i.e. the original question can also be included in the divergence set. We can find in the literature some methods that explore the of creating divergence sets and using them to materialize into metrics (Pezeshkpour and Hruschka, 2024; Wang et al., 2025).

One metric is MCQA+, based on the idea of computing the MCQA scores using disjoint divergence sets and aggregating the results using the mean of all M divergence sets, i.e. the mean for the entire set of questions. This metric can be defined as:

$$\text{MCQA+} = \frac{1}{N*M}\sum_{i=1}^{N}\sum_{j=1}^{M} llm(mcq*_i^j) \quad (2)$$

Notice that MCQA+ considers the divergence sets for generating alternative evaluations, but the aggregation of the scores under-explores the computation of consistency. In fact, we can say that MCQA+ does rely on an implicit use consistency, but given that incorrect response also contribute to the score, it is not trivial to associate the metic to the consistency of correct responses only.

Another metric that includes consistency in the computation of accuracy scores is the Majority Voting (MV) metric, proposed in (Pezeshkpour and Hruschka, 2024). This metric relies on the set of the

divergent sets $MCQ*$ and computes the correctness of an evaluation sample based on achieving the correct response in the majority of the derivations, i.e. if the LLM provides the correct response for more than half of the samples in $\hat{MCQ}*_i$, or in other words, more than half response consistency. That is, consider the definition response consistency for a given sample i as:

$$\text{RC(i)} = \frac{1}{M}\sum_{j=1}^{M} llm(mcq*_i^j) \tag{3}$$

Now, consider the function $1(expression)$, which returns 1 if $expression$ is true or 0 otherwise, the MV metric can then be defined as:

$$\text{MV} = \frac{1}{N}\sum_{i=1}^{N} 1(\text{RC(i)} > 0.5) \tag{4}$$

Even though MV relies on consistency to compute hits, i.e. the majority of the divergent questions need to get correct responses for a question to be computed as a hit, the metric relies on very permissive level of consistency (0.5), for which samples with low response consistency values have the same weights of those with higher values. In some sense, that hinders the impact of consistency in the metric, apart from being a more statistically-robust score compared with MCQA.

4 Using Consistency for More Reliable Accuracy Computation

In this section we propose our method to rebalance LLM accuracy on MC benchmarks, to which we refer as the Consistency-Rebalanced Accuracy (CoRA). The metric is built upon the idea of computing the Consistency Index (CI) score using the score computed with the Bare-Minimum-Consistency Accuracy (BMCA) method using 100% consistency as target, and then adjusting the scores computed with the MCQA method to scale down LLMs that are inconsistent, resulting in CoRA scores. Details are provided next.

4.1 Bare-Minimum-Consistency Accuracy

The first metric we propose is the Bare-Minimum Consistency Accuracy (BMCA), which can be considered as an extension of the MV metric but using the adjustable parameter c to determine the minimum response consistency level that is expected for a sample to be considered correct. That is, for each sample i, we will only consider the samples as correct if the RC score is greater than c

In greater details, given the consistency level c as a parameter, the BMCA metric can be defined as:

$$\text{BMCA(c)} = \frac{1}{N}\sum_{i=1}^{N} 1(\text{RC}(i) \geq c) \tag{5}$$

4.2 Consistency Index

Given that BMCA can compute scores for different levels of consistency, when $c = 1.0$, the metric will compute the accuracy score only for cases where the model provide 100% response consistency in the M trials.

We associate here the idea of this index with the elimination of random guessing as a viable option for the models being evaluated. As detailed in the appendix A, for $M = 10$, a model has to be guessing at a success rate greater than 0.9999 to be able to be 100% consistent on M trials. When no random guess is allowed at all, the LLM is arguably knowledgeable about the responses provided for the benchmark questions.

Therefore, we use BMCA(1.0) as a proxy to define the proportion of samples for which the LLM being evaluated demonstrates real knowledge when answering the questions, and use the score to compute a quality metric for the original MCQA score. We refer to this metric as the Consistency Index (CI) score.

Formally, the CI score is computed using the difference between MCQA and BMCA(1.0), providing the gap of the MCQA score to the accuracy considering only cases with 100% of response consistency, and subtracting from 1.0 so a higher value denotes higher consistency, such as in:

$$\text{CI} = 1.0 - (\text{MCQA} - \text{BMCA(1.0)}) \tag{6}$$

4.3 Consistency-Rebalanced Accuracy

The end result of our approach is the Consistency-Rebalanced Accuracy (CoRA), consisting of scaling down the scores computed with MCQA using the CI score described in the previous section. The idea is to take advantage of the CI score and adjust MCQA scores to make them reflect more authentically the quality of the LLM in terms of response consistency.

The implementation is straightforward, as we denote in the equation below:

$$\text{CoRA} = \text{MCQA} * \text{CI} \tag{7}$$

With this approach, the MCQA scores are at best kept, if the model presents 100% response consistency for the correct responses (which is very unlikely, as we will show later), but they can be scaled down as the LLM scores presents larger gaps from BMCA(1.0) to MCQA.

5 Details for Implementation

As a method to implement the divergent set $MCQ*$, we focus in creating derivations of the set of choices C by creating variations only in the way multiple options are presented to the model. Although we can also vary the set of questions Q with the variations in the input, such as with rephrasings, we wanted to avoid introducing any potential error in this process. By including variations in the set C, we can generate alternative sets of choices where the expected answer is always kept but the set of distractors is modified with three operators: reorderings, where the order of the choices presented to the LLM is modified, such as by shuffling the choices; deletions, where one or more alternatives are removed from the original set of choices; and inclusions, where alternatives that do not affect the expected answers are included, such as adding a *none of the above* (NOTA) alternative.

In details, consider that A is the number of alternatives in the original question, we consider the following approaches to generate altered sets of answer choices: **Shuffled**, where the set of choices is shuffled (we can shuffle multiple times, but in this work we shuffle only once); **With NOTA**, where each distractor is replaced by the NOTA alternative, resulting in $A - 1$ new sets of alternatives; **With NOTA shuffled**, employing a mix of *With NOTA* and *Shuffled*, where the NOTA alternative replaces a distractor and them the set of choices is shuffled (this approach also results in $A - 1$ variations); **Decoupled**, which takes the original set of choices and decouples it into $A-1$ binary subset of choices, pairing each of non-correct choices with the correct one in each subset; **Decoupled shuffled**, similar to *Decoupled* but with an additional shuffling step; **Decoupled with NOTA**, also mixing *Decoupled* with *With NOTA*, where the set of alternatives is decoupled into $A - 1$ binary subsets and a NOTA distractor is add to each subset, creating ternary subsets; **Decoupled with NOTA shuffled**, which is similar to *Decoupled with NOTA* with additional shufflings on the ternary subsets.

An illustration of the previously described techniques is presented in Figure 1. With these variations in the set of choices, we can then format a prompt for each variation and prompted the model for the correct alternative. We consider the following base prompt, where $QUESTION$ is replaced by the text of the question, followed by the corresponding options which are formatted and filled in $CHOICES$:

```
Answer the following multiple choice question.
The first line of your response should be of the
following format: 'LETTER' (without quotes), where
LETTER is one of ABCD (depending on the number
of alternatives), followed by a step-by-step
explanation.

Question: $QUESTION$
Choices: $CHOICES$
Answer:
```

To evaluate the output of the LLMs, we parse the first token of the response and remove any additional character such as punctuation and spaces.

Our method generates a total of $2 + 6 * (A - 1)$ variations of the set of choices for each question. For instance, with with five alternatives, i.e. $A = 5$, 26 variations are generated. Notice that one could easily generate more variations by including more exhaustive shufflings and generate subsets with more than two and less than M alternatives, but that implies in extra costs to evaluate the models since we need to query the model for each variations of the original question, and there is a linear increase in the cost of running LLM inferences.

6 Empirical Evaluation

In this section we describe the evaluations conducted to validate our proposed CoRA metric. We focus on comparing the results against the three baselines described in Section 3: MCQA, MCQA+, and MCQA+MV.

We divide this evaluation into two main parts. In this first we focus on a domain-specific dataset, i.e. the MedQA benchmark (Jin et al., 2020), considering four different LLMs either finetuned on medical data or known as good performer in this type of data: GPT4o version "gpt-4o-2024-11-20", MedLlama3 7B (MedL) (Medical, 2024), BioMedical Llama3 8B (BioML) (Medical, 2024), and BioMistral 7B (BMist) (Labrak et al., 2024). In the second part we evaluate three general knowledge benchmarks, i.e. MMLU (Hendrycks et al., 2021), Arc-C (Clark et al., 2018), and TruthFulQA (Lin et al., 2022), considering four general-purpose LLMs: Mistral v0.1 7B (Mist) (Jiang et al., 2023), Llama 3.1 8B (Llam) (Aaron Grattafiori et al, 2024), Granite 3.0 8B (Gran) (Granite Team, 2024), and DeepSeek chat 7B (DSeek) (DeepSeek, 2024). No-

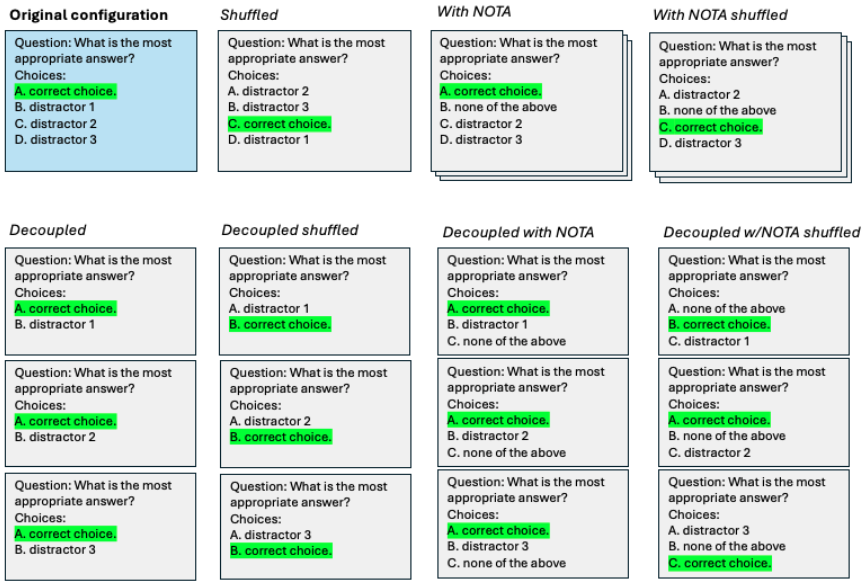

Figure 1: Illustration of the methods to divergent sets of alternatives

tice that we focused on models with slightly similar sizes, i.e. around 7B to 8B parameters, except for GPT4o whose size is not publicly disclosed. For MedQA we consider the 1,273 5-choice questions extracted from USMLE exams, resulting in generating 26 variations for the $MCQ*$ set of divergent questions, and use 0-shot prompts. For MMLU, we used 14,042 questions with either 3 or 4 alternatives, generating 14 to 20 questions in $MCQ*$, and evaluated with 5-shot prompts. For Arc-C, we used 1,172 questions with either 4 or 5 alternatives, generating 20 to 26 questions in $MCQ*$, and evaluated with 25-shot prompts. And for TruthFulQA, we consider 817 questions ranging from 2 or 12 alternatives, generating 14 to 74 questions in $MCQ*$, and evaluated with 0-shot prompts. Notice that the decision for in-context learning method was based on common practices from the literature (Aaron Grattafiori et al, 2024; DeepSeek-AI and Aixin Liu et al, 2024).

For a better understanding of the CoRA scores, for each LLM and dataset we present an extensive evaluation of BMCA considering six different values for c, i.e. 0.5, 0.6, 0.7, 0.8, 0.9, and 1.0, noticing that 0.5 represents a borderline level consistency for deciding if a question is correctly responded or not, and 1.0 corresponds to full consistency, i.e. all responses from the divergent set of questions are correct. That range of values is useful to provide a progressive analysis on the consistency of an LLM, i.e. how the accuracy is affect as we increase the value of c. We report also all values computed for the CI score.

Table 1: Results on MedQA dataset. In parentheses the comparison ranking.

LLMs:	GPT4o	MedL	BioML	BMist
Baselines				
MCQA	0.85 (1)	0.74 (2)	0.73 (3)	0.38 (4)
MCQA+	0.90 (1)	0.69 (3)	0.75 (2)	0.58 (4)
MV	0.91 (1)	0.73 (3)	0.77 (2)	0.58 (4)
Proposed metric				
CoRA	0.74 (1)	0.32 (3)	0.42 (2)	0.28 (4)
Secondary metrics				
BMCA(c)				
$c \geq 0.5$	0.91	0.75	0.80	0.61
$c \geq 0.6$	0.89	0.67	0.73	0.49
$c \geq 0.7$	0.86	0.57	0.66	0.40
$c \geq 0.8$	0.84	0.49	0.60	0.33
$c \geq 0.9$	0.79	0.34	0.46	0.22
$c \geq 1.0$	0.73	0.18	0.31	0.11
CI	0.88	0.44	0.58	0.73

The results on MedQA are presented in Table 1. When comparing the scores of MCQA+ and MV baselines to MCQA, it is noticeable how the scores of all models, except MedL, are magnified with the expanded evaluation with divergent questions. On the other hand CoRA, presents reduced scores by about 0.11 points for GPT4, 0.42 points for MedL, 0.31 for BioML, 0.10 for BMist. Note that such a decrease is proportional to the CI score, reported in the last row of the table. By looking at the scores from BMCA(1.0) and comparing them to those of MCQA, we can clearly observe that there is a drop for all models, being GPT4o undeniably more

consistency, reaching a score of 0.73, i.e. 73% of the correct responses present $RC(i) = 1.0$. For the other models, on the other hand, less than half of the correct cases present 100% response consistency, and that gap in consistency is represented in CoRA scores that are much lower compared with MCQA. That is, MedL drops from 0.74 with MCQA to 0.32 with CorA, BioML from 0.73 to 0.42, and BMist from 0.38 to 0.28. Further evidence on the differences in consistency can be found by looking at the different scores provided by BMCA, showing that models such as MedL and BioML present a drastic decrease in accuracy as the minimum consistency requirement increase, while the decrease is not as drastic for BioML and very subtle for GPT4.

Table 2: Results on MMLU dataset. In parentheses the comparison ranking.

LLMs:	Mist	Llam	Gran	DSeek
Baselines				
MCQA	0.64 (2)	0.58 (3)	0.65 (1)	0.52 (4)
MCQA+	0.75 (1)	0.62 (3)	0.75 (1)	0.60 (4)
MV	0.78 (1)	0.61 (3)	0.76 (2)	0.58 (4)
Proposed metric				
CoRA	0.44 (2)	0.33 (3)	0.47 (1)	0.33 (3)
Secondary metrics				
BMCA(c)				
$c \geq 0.5$	0.83	0.65	0.80	0.63
$c \geq 0.6$	0.74	0.55	0.73	0.54
$c \geq 0.7$	0.65	0.46	0.66	0.44
$c \geq 0.8$	0.56	0.38	0.58	0.36
$c \geq 0.9$	0.47	0.28	0.50	0.27
$c \geq 1.0$	0.33	0.15	0.38	0.16
CI	0.69	0.57	0.73	0.64

Table 3: Results on Arc-C dataset. In parentheses the comparison ranking.

LLMs:	Mist	Llam	Gran	DSeek
Baselines				
MCQA	0.80 (2)	0.72 (3)	0.82 (1)	0.64 (4)
MCQA+	0.90 (1)	0.73 (3)	0.89 (2)	0.73 (3)
MV	0.92 (1)	0.81 (3)	0.91 (2)	0.78 (4)
Proposed metric				
CoRA	0.58 (2)	0.25 (4)	0.65 (1)	0.38 (3)
Secondary metrics				
BMCA(c)				
$c \geq 0.5$	0.94	0.84	0.92	0.82
$c \geq 0.6$	0.90	0.78	0.88	0.73
$c \geq 0.7$	0.84	0.69	0.84	0.64
$c \geq 0.8$	0.80	0.53	0.80	0.53
$c \geq 0.9$	0.70	0.32	0.73	0.42
$c \geq 1.0$	0.52	0.07	0.61	0.23
CI	0.72	0.35	0.79	0.59

The results on general-knowledge benchmarks are presented in Table 2, Table 3, and Table 4, for MMLU, Arc-C, and TruthfulQA, respectively. In this case, we observe a similar scenario apart the fact that there is not a very top-performing LLM such as GPT4o in the comparison. As it can be observed, the maximum CI score is of 0.79 in Arc-C with Gran, but CI can go as low as 0.35 with Llam in the same benchmark. That low CI score for Llam reflects a surprisingly inconsistency LLM on the Arc-C benchmark, and the resulting CoRA score reflects this lack of consistency. That is, from a difference of 0.10 MCQA points from Llam to Gran, the top performer in that benchmark, the difference becomes 0.40 points with CoRA. Notice that MCQA+ and MV keep gaps that are much closer to that of MCQA, i.e. 0.16 and 0.10, respectively. Overall, considering the best MCQA scores, that were achieved with Gran, we observe that our CoRA metric reflects at least a drop of 0.18 points in MMLU, 0.17 in Arc-C, and 0.09 in TruthfulQA. On the other hand, the drop in score for Llam, one of the worst performers in CoRA, can be as high as 0.25 points in MMLU, 0.57 in Arc-C, and 0.14 in TruthfulQA. It is intriguing that both the most and the least consistent LLM are the same for all three benchmarks, a fact that might indicate that the consistency might be a feature that generalizes among different benchmarks, but further investigation is needed to support this claim.

Table 4: Results on TruthfulQA dataset. In parentheses the comparison ranking.

LLMs:	Mist	Llam	Gran	DSeek
Baselines				
MCQA	0.41 (1)	0.41 (1)	0.34 (3)	0.34 (3)
MCQA+	0.49 (1)	0.49 (1)	0.39 (4)	0.40 (3)
MV	0.45 (2)	0.47 (1)	0.35 (4)	0.37 (3)
Proposed metric				
CoRA	0.28 (1)	0.27 (2)	0.25 (3)	0.24 (4)
Secondary metrics				
BMCA(c)				
$c \geq 0.5$	0.48	0.51	0.36	0.40
$c \geq 0.6$	0.40	0.40	0.30	0.31
$c \geq 0.7$	0.33	0.31	0.25	0.23
$c \geq 0.8$	0.25	0.22	0.19	0.17
$c \geq 0.9$	0.16	0.14	0.14	0.11
$c \geq 1.0$	0.09	0.08	0.09	0.05
CI	0.68	0.67	0.75	0.71

6.1 Ablation Studies

In this section we present an ablation study on the set of divergent questions used to compute our metrics. As depicted in Figure 1, some methods can keep the same number of alternatives as the original question (first row in Figure 1), but other methods can either augment or reduce that sets creating uneven uniform distribution with the likelihood of pre-

dicting the correct question, which can contribute for metrics such as MCQA+ and MV to produce higher scores. For this reason, in this section not only we present results with altered questions with only the exact same number of alternatives of the original question, but we also conduct a thorough statistical analysis on the set of altered questions based on bootstrap resampling. We focus on the MedQA dataset, which presents an invariable number of alternatives for the entire dataset and can be used for both evaluations.

Results with the set of only ten divergent questions with the same number of alternatives are presented in Table 5. As somewhat expected, the results of MCQA+ and MV present a drop compared to the numbers reported in Table 1, the table containing analogous results with all 26 divergent questions. The score from CoRA, on the other hand, present slightly increases. We think that having a smaller set of divergent questions possibly reduces the impact of the consistency evaluation for these metrics, but we need further investigation to find more evidence to confirm this hypothesis.

Table 5: Results on MedQA dataset - 5-alternative questions only. In parentheses the comparison ranking.

LLMs:	GPT4o	MedL	BioML	BMist
5-alternative-only divergent questions				
MCQA+	0.86 (1)	0.62 (3)	0.67 (2)	0.43 (4)
MV	0.85 (1)	0.62 (3)	0.67 (2)	0.38 (4)
CoRA	0.77 (1)	0.37 (3)	0.48 (2)	0.28 (4)
difference from Table 1				
MCQA+	-0.04	-0.07	-0.08	-0.15
MV	-0.06	-0.11	-0.10	-0.20
CoRA	0.03	0.05	0.06	0.00

We focus in understanding better the sensitiveness of the metric to the set of altered question, independently of the distribution and number of alternatives. For that we conducted a bootstrap resampling analysis, by generating 10,000 evaluations with 100 altered questions randomly selected with replacement from the 26 divergent questions created with MedQA.

The means of the scores computed with the 10,000 bootstrapped resamplings are presented in Table 6. First, it is eye catching the usually low standard deviations, demonstrating an interesting stability of the score independently of the set of altered choices. Notice also that all methods present very small differences when compared to the seed divergence set with 26 variations, as reported in the bottom portion of the table. That is quite interesting since it indicates that the variations presented in Section 5 are relatively robust for the evaluation of response consistency, and our proposed method to generate variations can be used to evaluate LLMs without much concern with intrinsic variability from the set of altered choices.

Table 6: Results on MedQA dataset - means of 10,000 resamplings of 100-samples bootstrapped divergent questions. In parentheses the standard deviation multiplied by 10^3.

LLMs:	GPT4o	MedL	BioML	BMist
10 bootstrapped divergent questions				
MCQA+	0.90 (3)	0.69 (11)	0.75 (9)	0.58 (13)
MV	0.91 (4)	0.73 (15)	0.78 (12)	0.59 (21)
CoRA	0.75 (1)	0.33 (3)	0.42 (2)	0.28 (1)
difference to Table 1				
MCQA+	0.00	0.00	0.00	0.00
MV	0.00	0.00	0.01	0.01
CoRA	0.01	0.01	0.00	0.00

7 Conclusions and Future Work

In this work we proposed the CoRA metric to enhance the way LLMs are evaluated on MC benchmarks, which explores the concept of response consistency to rebalance the scores computed from the ratio for hits of an LLM on MC benchmarks and provide more faithfully the capabilities of such models. And our evaluations on well-known benchmarks show that CoRA is able to redistribute the scores according to the consistency of the LLM, which is demonstrated with the CI scores, improving the reliability of LLM evaluation compared to state-of-the-art metrics that do not reflect any aspect of response consistency in the scores. Furthermore, we conducted an ablation study focused at evaluating the sensitiveness of to the set of altered answer choices and demonstrate that our proposed generation method is relatively robust, practically equivalent to the scores obtained with bootstrap resampling.

As future work we believe we can improve the methodology in different ways. There is room to investigate and increase the set of divergent questions, and also in exploring further simpler and more general methods such as shuffling. Another direction lies in revisiting the way consistency is used for rebalancing scores, for instance by taking more advantage of BMCA computed on multiple values for c. Lastly, it is key to understand how this work can be expanded to other types of benchmarks beyond multiple choice, and possibly how these ideas can be used to make LLMs safer in real time, during the inference.

8 Limitations

The first limitation of this work is the focus on multiple choice benchmarks only, so the results from this paper do not directly transfer to other types of benchmarks such as open-ended questions. Also, the evaluation comprises a limited set of benchmarks, so further experiments shall be conducted to validate the generalization of our methods. Another limitation is that set of LLMs that we evaluated, given that the size of the models usually tops at around 7B to 8B parameters, so experiments with larger LLMs should also be conducted in the future. Finally, we have not provided any deep discussion on the computational complexity increase of our method, but we decided to not delve into that discussion since, in general terms, the complexity for computing CoRA is roughly equivalent to that of both MCQA+ and MV.

9 Ethical Statement

We have not identified any ethical issue, since the LLMs and benchmarks are publicly available and we just followed commonly-used practices.

References

Aaron Grattafiori et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Samuel Ackerman, Ella Rabinovich, Eitan Farchi, and Ateret Anaby Tavor. 2024. A novel metric for measuring the robustness of large language models in non-adversarial scenarios. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 2794–2802, Miami, Florida, USA. Association for Computational Linguistics.

Nishant Balepur, Abhilasha Ravichander, and Rachel Rudinger. 2024. Artifacts or abduction: How do LLMs answer multiple-choice questions without the question? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10308–10330, Bangkok, Thailand. Association for Computational Linguistics.

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:1803.05457v1*.

Donald A. Curtis, Susanne L. Lind, Christy K. Boscardin, and Mark Dellinges. 2013. Does student confidence on multiple-choice question assessments provide useful information? *Medical Education*, 47(6):578–584.

DeepSeek. 2024. Deepseek 7b chat. https://huggingface.co/deepseek-ai/deepseek-llm-7b-chat.

DeepSeek-AI and Aixin Liu et al. 2024. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Abhimanyu Dubey and et al. 2024. The Llama 3 herd of models. *Preprint*, arXiv:2407.21783.

IBM Granite Team. 2024. Granite 3.0 language models.

Renata Grazziotin-Soares, Coca Blue, Rachel Feraro, Kristen Tochor, Thiago Machado Ardenghi, Donald Curtis, and Diego Machado Ardenghi. 2021. The interrelationship between confidence and correctness in a multiple-choice assessment: pointing out misconceptions and assuring valuable questions. *BDJ Open*, 7(1):10.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2020. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *arXiv preprint arXiv:2009.13081*.

Abhishek Kumar, Robert Morabito, Sanzhar Umbet, Jad Kabbara, and Ali Emami. 2024. Confidence under the hood: An investigation into the confidence-probability alignment in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 315–334, Bangkok, Thailand. Association for Computational Linguistics.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. Biomistral: A collection of open-source pretrained large language models for medical domains. *Preprint*, arXiv:2402.10373.

Fangyu Lei, Qian Liu, Yiming Huang, Shizhu He, Jun Zhao, and Kang Liu. 2024. S3Eval: A synthetic, scalable, systematic evaluation suite for large language model. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1259–1286, Mexico City, Mexico. Association for Computational Linguistics.

Ellen Lesage, Martin Valcke, and Elien Sabbe. 2013. Scoring methods for multiple choice assessment in higher education – is it still a matter of number right scoring or negative marking? *Studies in Educational Evaluation*, 39(3):188–193.

Wangyue Li, Liangzhi Li, Tong Xiang, Xiao Liu, Wei Deng, and Noa Garcia. 2024. Can multiple-choice questions really be useful in detecting the abilities of LLMs? In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2819–2834, Torino, Italia. ELRA and ICCL.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Yen-Ting Lin and Yun-Nung Chen. 2023. LLM-eval: Unified multi-dimensional automatic evaluation for open-domain conversations with large language models. In *Proceedings of the 5th Workshop on NLP for Conversational AI (NLP4ConvAI 2023)*, pages 47–58, Toronto, Canada. Association for Computational Linguistics.

Probe Medical. 2024. Medllama3 v20. https://huggingface.co/ProbeMedicalYonseiMAILab/medllama3-v20.

Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. 2024. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models. *Preprint*, arXiv:2410.05229.

Aidar Myrzakhan, Sondos Mahmoud Bsharat, and Zhiqiang Shen. 2024. Open-llm-leaderboard: From multi-choice to open-style questions for llms evaluation, benchmark, and arena. *Preprint*, arXiv:2406.07545.

Harsha Nori, Yin Tat Lee, Sheng Zhang, Dean Carignan, Richard Edgar, Nicolo Fusi, Nicholas King, Jonathan Larson, Yuanzhi Li, Weishung Liu, Renqian Luo, Scott Mayer McKinney, Robert Osazuwa Ness, Hoifung Poon, Tao Qin, Naoto Usuyama, Chris White, and Eric Horvitz. 2023. Can generalist foundation models outcompete special-purpose tuning? case study in medicine. *Preprint*, arXiv:2311.16452.

Pouya Pezeshkpour and Estevam Hruschka. 2024. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2006–2017, Mexico City, Mexico. Association for Computational Linguistics.

Yuval Reif and Roy Schwartz. 2024. Beyond performance: Quantifying and mitigating label bias in LLMs. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6784–6798, Mexico City, Mexico. Association for Computational Linguistics.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, Mike Schaekermann, Amy Wang, Mohamed Amin, Sami Lachgar, Philip Mansfield, Sushant Prakash, Bradley Green, Ewa Dominowska, Blaise Aguera y Arcas, Nenad Tomasev, Yun Liu, Renee Wong, Christopher Semturs, S. Sara Mahdavi, Joelle Barral, Dale Webster, Greg S. Corrado, Yossi Matias, Shekoofeh Azizi, Alan Karthikesalingam, and Vivek Natarajan. 2023. Towards expert-level medical question answering with large language models. *Preprint*, arXiv:2305.09617.

Haochun Wang, Sendong Zhao, Zewen Qiang, Nuwa Xi, Bing Qin, and Ting Liu. 2024a. Beyond the answers: Reviewing the rationality of multiple choice question answering for the evaluation of large language models. *Preprint*, arXiv:2402.01349.

Haochun Wang, Sendong Zhao, Zewen Qiang, Nuwa Xi, Bing Qin, and Ting Liu. 2025. LLMs may perform MCQA by selecting the least incorrect option. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 5852–5862, Abu Dhabi, UAE. Association for Computational Linguistics.

Weixuan Wang, Barry Haddow, Alexandra Birch, and Wei Peng. 2024b. Assessing factual reliability of large language model knowledge. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 805–819, Mexico City, Mexico. Association for Computational Linguistics.

Fangyun Wei, Xi Chen, and Lin Luo. 2024. Rethinking generative large language model evaluation for semantic comprehension. *Preprint*, arXiv:2403.07872.

Sarah Wiegreffe, Oyvind Tafjord, Yonatan Belinkov, Hannaneh Hajishirzi, and Ashish Sabharwal. 2024. Answer, assemble, ace: Understanding how transformers answer multiple choice questions. *Preprint*, arXiv:2407.15018.

Fanghua Ye, Mingming Yang, Jianhui Pang, Longyue Wang, Derek F. Wong, Emine Yilmaz, Shuming Shi, and Zhaopeng Tu. 2024. Benchmarking llms via uncertainty quantification. *Preprint*, arXiv:2401.12794.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2024. Large language models are not robust multiple choice selectors. *Preprint*, arXiv:2309.03882.

A Guessing on Benchmarks

Requiring consistency of correct answers on multiple, independent answers to the same question of a benchmark is, intuitively, a way to assure that models which are doing, to some extent, random guessing in a multiple choice benchmark. Here we explore the impact of requirement consistency in M trials in terms of determining minimum values for the guessing rate to assure that a given consistency level is met.

We start considering a single multiple-choice question q of k choices which is repeated evaluated by a model M times, yielding answers $llm(q_i), 1 \leq i \leq M$, where $llm(q_i) = 1$ if, and only if, the model produces the correct alternative.

In M trials, the number of possible arrangement of choices where exactly p are correct, $C_M(p)$ is:

$$C_M(p) = \left(\begin{array}{c} M \\ p \end{array} \right) = \frac{M!}{(M-p)!\,p!} \quad (8)$$

Given a guessing rate r, where $r = 1/p$ if it is a purely random guess, the probability of guessing correctly exactly p of the M trials, $T_r^M(p)$ is:

$$P(T_r^M(p)) = C_M(p) r^p (1-r)^{(M-p)} \quad (9)$$

Following, the probability of guessing correctly p or more answers in M trials, $\overline{T}_r^M(p)$ is, clearly:

$$P(\overline{T}_r^M(p)) = \sum_{j=p}^{M} C_M(j) r^j (1-r)^{(M-j)} \quad (10)$$

The two leftmost columns of table A show, for different values of p, the value of $\overline{T}_r^M(p))$ for $M = 10$ trials of $k = 5$ choices, when the guessing rate is purely random, $r = 1/k$. For instance, the probability of obtaining 10 correct answers in 10 trials of a question if the model is randomly guessing is 0.0000001.

Conversely, now imagine that the model as an "oracle" which guesses the correct answer at a certain success rate, SGR. We can then compute the minimum success needed to always get at least p correct answers in M, what we call the *minimum success guessing rate*, $MSGR(p)$. We computed numerically such values, and the rightmost column of table A displays the values for $M = 10$ trials of $k = 5$ choices. It shows, for instance, that that a model has to be guessing at least of a success rate of 0.93 to achieve 6 out of 10 correct answers ($MSGR(6)$), which is equivalent to the requirements of the metric MV proposed in (Pezeshkpour and Hruschka, 2024).

In our view, a $MSGR(6) = 0.93$ is still insufficient to guarantee that a model actually knows the contents of a multiple-choice benchmark. However, requiring that the model is consistent in 10 out of 10 trials ($MSGR(10)$) warrants that a model can only successfully guess if its success rate is above 0.9999, which we consider a reasonable requirement to consider a model knowledgeable in a subject.

Table 7: Guessing probabilities of p or greater correct answers in $M = 10$ trials of $k = 5$ choices, for random guessing; and minimum success guessing rate $MSGR(p)$.

$\overline{T}_r^{10}(p), k = 5$		
p	random ($r = 1/k$)	$MSGR(p)$
0	1.000	0.2
1	0.893	0.54
2	0.624	0.66
3	0.322	0.75
4	0.121	0.82
5	0.033	0.88
6	0.006	0.93
7	0.0009	0.96
8	0.00008	0.98
9	0.000004	0.99
10	0.0000001	0.9999