The issue of hallucination in language models (LMs) has gained significant attention due to its negative impact on performance and the risks it introduces in various natural language processing (NLP) tasks, such as machine translation (Zhou et al., 2020), summarization (Cao et al., 2022), dialogue generation (Das et al., 2023), and question answering (Zhang et al., 2023a; Zheng et al., 2023b; Dhuliawala et al., 2023). Recent survey (Ji et al., 2023; Zhang et al., 2023c; Ye et al., 2023) and evaluation benchmarks (Liu et al., 2021; Li et al., 2023a; Yang et al., 2023) have highlighted the importance of addressing this issue. Previous research has explored hallucination evaluation using confidence-based approaches (Xiao and Wang, 2021; Varshney et al., 2023; Chen and Mueller, 2023) that require access to token-level log probability (Kuhn et al., 2023; Cole et al., 2023) or supervised tuning (Agrawal et al., 2023; Li et al., 2023b) that relies on internal states of the LM. However, these methods may not be applicable when only API access to the LM is available (Agrawal et al., 2023). Another approach involves retrieving knowledge from external databases to tackle hallucinations (Ji et al., 2022; Zheng et al., 2023a; Peng et al., 2023; Zhang et al., 2023b).

Recent studies employ self-consistency to detect hallucinations based on pretrained LMs (Manakul et al., 2023) and instruction-tuned LMs (Mündler et al., 2023). Although these methods exhibit promising accuracy on several specific tasks, potential failures (Chen et al., 2023) of self-consistency are overlooked in the current settings, as existing LMs frequently provide inconsistent responses to questions (Mitchell et al., 2022) and factual knowledge inquiries (Elazar et al., 2021; Tam et al., 2023; Gekhman et al., 2023). Our work addresses these concerns by introducing a cross-check consistency approach, aiming to bridge the gap between self consistency and factual assessment.

The essential assumption of self-consistency in factuality assessment is that if the LM has the knowledge of the concept, responses sampled from its output distribution, should be similar and consistent; conversely, if the LM lacks corresponding knowledge, the sampled responses would contain hallucinated facts that are diverged and contradictory. Although this assumption may seem reasonable, it does not always hold in practice. Specifically, we argue that solely checking the LM’s self consistency is insufficient for detecting hallucination or verifying factuality under the following two circumstances:

1. *LMs may produce consistently hallucinated facts.* We observe that for certain questions, LMs may output consistently wrong answers. For instance, when prompted with the question “Is pi smaller than 3.2?”, ChatGPT consistently generates incorrect answers. In this case, where the generated responses are consistent but non-factual, solely relying on self-consistency checking of a single model would yield false negative hallucination detection results.

*2. Even in cases when LMs generate factual statements in their original response, the stochastic sampled responses may lack veracity.* For example, the original answer (Answer 1) of ChatGPT under zero temperature is correct regarding the senator search question. However, when sampled with a higher temperature, ChatGPT generates multiple incorrect responses (Answer 2 and Answer *m*). In this scenario, where the sampled responses are inconsistent and disagree with the original response which itself is factually correct, methods that rely solely on model self-checking would produce false positives.

In summary, although the inconsistency of sampled responses has been empirically demonstrated to be correlated with hallucinated facts on certain tasks, in general, self-consistency is neither necessary nor sufficient to verify the veracity of large LMs’ statements. Therefore, methods based solely on self-consistency checking may not be able to accurately detect hallucinations in complex QA and open-domain generation tasks, which motivates us to design a more reliable and robust factuality assessment method that extends this idea.

To address question level hallucination, we introduce a mechanism that perturbs semantically equivalent questions to evaluate the consistency of LMs’ responses across variants of the same question. By examining the generated answers to these perturbed questions, we are able to identify cases where the LM consistently provides incorrect responses to a specific question, which is indicative of a question-level hallucination. Furthermore, we address model-level hallucination by introducing cross-model response consistency checking, which involves comparing the responses of different LMs to the same set of questions. By identifying discrepancies between the responses of different models, we are able to pinpoint cases where certain models exhibit hallucinations while others provide correct answers. Integrating these cross-checking extensions into our approach significantly improves its ability to detect hallucinations that go beyond self-consistency, thereby providing a more comprehensive assessment of the presence of question-level and model-level hallucinations.

consider model-specific characteristics

The above work was done with gpt-3.5-turbo.

I also tried guanaco-33b and falcon-7b used in the paper, but the session always crashed with guanaco-33b due to no available RAM any more (I have 83 GB with Google Colab). The good news is that Falcon-7b works. And I'm testing other models to see if they work.

First, I tested ChatGPT-3.5 with 50 questions as you said during our last meeting, with different number of self-responses (3, 5, 10 and 15 responses for one question in self-check) and pertubation questions (5 and 10 semantically equivalent questions). It worked. The performing  of using cross-check much better than the self-check consistency. However, the AUROC score for cross-check was much higher than the score reported in the paper (86.96 compared to 81.3). I thought it might be overfitting, so I tried several more times, but it was still very high. So, I used random sampling to test it. And it remained high. Then, I tested 100 questions, and the AUROC score for 100 questions was much lower than the score reported in the paper (77-79 compared to 81.3). I also tested 100 questions with GPT-4.0, and got similar results.

I made a big mistake on sticking to testing ChatGPT because it was costly. I did not realize it until Anastassia sent me an email saying  there had been miscalculations in my account and I had processed about 100 million tokens. I was shocked and stopped using GPT API key immediately. I sincerely apologize for this oversight and should have known the pricing policy better before using the API key.

I then started testing other models but encountered memory issues with models exceeding 10 billion parameters (e.g., Guanaco-13b and Llama-3-13b) and GPU constraints. Therefore, I decided to focus on models with fewer than 10 billion parameters, using CPU instead.

I tested Gemma-7B-Instruct, the latest Llama-3-8B-Instruct, Mistral-7B-Instruct-v0.2, Phi-3-mini-128k-instruct and Qwen1.5-7B-Chat, with parameters ranging from 3.8 billion to 8 billion. They are all open-source language models on HuggingFace, released by different companies. Since some models are not sensitive to the original prompt provided by the authors (for example, the prompt requires responding with Guess: Yes/No (if the two question-answer pairs are semantically equivalent), but Mistral-7B and Phi-3-mini always respond with answers to the questions (not Yes or No), I modified the original prompt according to the responses of different models, stressing the format of their responses or giving example question and answer in the prompt.

SAC3 worked with all five models, achieving higher AUROC scores with cross-check than with self-check consistency. Scores ranged from 56% to 94%. Phi-3-mini performed surprisingly well, with an AUROC score of 80% in self-check and 86-90% in cross-check, at much faster speeds.  I've attached two charts showing the AUROC scores:

*Chart 1: Comparing AUROC scores for 5 responses (self-check) vs. 5 perturbation questions (cross-check).*

*Chart 2: Comparing AUROC scores for 10 responses (self-check) vs. 10 perturbation questions (cross-check).*

Due to memory and GPU constraints, I could only test 10 questions per model using random sampling. I also explored different numbers of self-responses and perturbation questions. Self-check consistency had the highest AUROC scores with 5 or 10 responses to the original question, while cross-check consistency had the highest AUROC scores with 10 perturbation questions (except for Mistral-7B, which peaked at 5 and 8).

To test more data, Dylan suggested using bootstrap sampling to achieve more reliable results. I'm testing these five models with bootstrap (randomly sampling 10 questions at a time and repeating 5 times, for 50 different questions). I will send you the results as soon as I finish them.