Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Multi-perspective consistency checking for large language model hallucination detection: a black-box zero-resource approach

Linggang KONG[†1,2], Xiaofeng ZHONG[1,2], Jie CHEN[1,2], Haoran FU[1,2], Yongjie WANG[†‡1,2]

[1]*College of Electronic Engineering, National University of Defense Technology, Hefei 230037, China*

[2]*Anhui Province Key Laboratory of Cyberspace Security Situation Awareness and Evaluation, Hefei 230037, China*

[†]E-mail: konglinggang@nudt.edu.cn; wangyongjie17@nudt.edu.cn

Received Mar. 21, 2025; Revision accepted Oct. 8, 2025; Crosschecked Oct. 31, 2025; Published online Nov. 24, 2025

Abstract: Large language models (LLMs) have been applied across various domains due to their superior natural language processing and generation capabilities. Nonetheless, LLMs occasionally generate content that contradicts real-world facts, known as hallucinations, posing significant challenges for real-world applications. To enhance the reliability of LLMs, it is imperative to detect hallucinations within LLM generations. Approaches that retrieve external knowledge or inspect the internal states of the model are frequently used to detect hallucinations; however, this requires either white-box access to the LLM or reliable expert knowledge resources, raising a high barrier for end-users. To address these challenges, we propose a black-box zero-resource approach for detecting LLM hallucinations, which primarily leverages multi-perspective consistency checking. The proposed approach mitigates the LLM overconfidence phenomenon by integrating multi-perspective consistency scores from both queries and responses. In comparison to the single-perspective detection approach, our proposed approach demonstrates superior performance in detecting hallucinations across multiple datasets and LLMs. Notably, in one experiment, where the hallucination rate reaches 94.7%, our approach improves the balanced accuracy (B-ACC) by 2.3 percentage points compared with the single consistency approach and achieves an area under the curve (AUC) of 0.832, all without depending on any external resources.

Key words: Large language models (LLMs); LLM hallucination detection; Consistency checking; LLM security
https://doi.org/10.1631/FITEE.2500180 **CLC number:** TP18

1 Introduction

Based on user input, large language models (LLMs) can generate fluent and coherent responses by drawing upon their internalized parameter knowledge, an ability that has been widely applied across various fields such as law, medicine, cybersecurity, and education (Luo and Yang, 2024). However, practical applications have also revealed certain issues that hinder further dissemination of LLMs. One of the most significant issues is hallucination (Zhang JX et al., 2023; Farquhar et al., 2024), wherein the contents generated by LLMs contradict real-world factual information. As illustrated in Fig. 1, when a user queries the LLM about the author and publication year of a book (Where You'll Find Me: And Other Stories), it generates a response that is entirely inconsistent with reality. If users trust the content generated by the LLM, it will lead to complete misdirection and a trust crisis (Farquhar et al., 2024). Despite existing research efforts designed to alleviate hallucinations, LLM hallucinations can be mitigated but not eradicated (Zhang SL et al., 2024).

‡ Corresponding author
ORCID: Linggang KONG, https://orcid.org/0000-0002-2477-118X

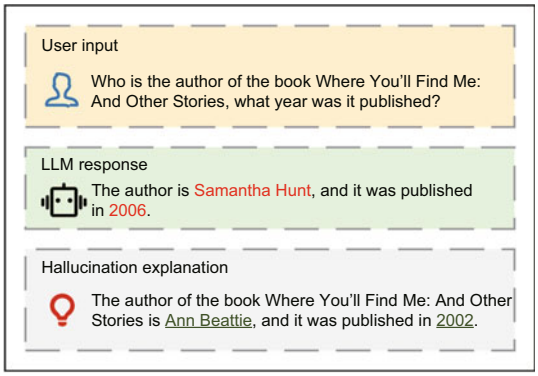

Fig. 1 LLM hallucinations occurred in LLM responses and the corresponding hallucination explanation. References to color refer to the online version of this figure

Consequently, the research of LLM hallucination detection is of significant importance.

One frequently used approach for detecting hallucinations involves using external expert knowledge bases to assist in verifying the LLM-generated content (Guan et al., 2024). Those external knowledge bases encompass resources such as knowledge graphs or more potent third-party LLMs. However, it is an intricate task to construct and maintain a completely reliable knowledge graph and prohibitively expensive to call the application programming interfaces (APIs) of other more powerful LLMs (Cheng XX et al., 2024). Furthermore, it is challenging to scale in scenarios where LLMs deployed at the edge restrict access to external resources (Li JY et al., 2024). Another approach posits that the hallucinations are embedded within the internal states of the model, such as certain latent space, activations, or attention heads (Wan et al., 2024). A classifier is often used to discriminate between the hallucinated and factual responses by analyzing the feature representations of internal states (Hu et al., 2024). Evidently, this represents a direct way of detecting hallucinations and allows for each inference output along with the predicted label. However, this requires white-box LLMs with full access granted to the users (Du et al., 2024), which creates a significant barrier for general users.

Although the aforementioned research contributes to LLM hallucination detection tasks, it is seldom applicable to scenarios where there is limited access to resources and the model is a complete black box. To address these issues, self-evaluation approaches have been proposed (Cheng FP et al., 2024; Liang et al., 2024), such as SelfCheckGPT (Manakul et al., 2023), PTrue (Kadavath et al., 2022), and Verbalized (Tian et al., 2023). The underlying mechanism is that LLMs can combat LLM hallucinations (Verspoor, 2024). Obviously, this often encounters a situation where LLMs are overconfident in their generated content, which means even though LLMs believe that their responses are factual, they are still hallucinated, as shown in Fig. 2. The reason is that those approaches detect hallucinations only from a single perspective, which is extremely limited and inadequate. To address that issue, we introduce a new hallucination detection approach from multiple perspectives to counteract the LLM overconfidence phenomenon. The proposed approach fuses the consistency scores from both the query perspective and the response perspective. The core of this approach is that when the LLM generates multiple responses based on the same query, the sampled responses are likely to be similar if the response is factual (Manakul et al., 2023; Li TJ et al., 2024). In addition, it is believed that if the response is factual, the original question can be reconstructed from the response and vice versa. Given that the query is written by users, the correctness is guaranteed compared with the uncertain and overconfident response generated by LLMs. The proposed approach detects LLM hallucinations from multiple perspectives, effectively enhancing LLM self-evaluation capabilities, and compensating for the deficiency of single-perspective approaches.

This paper proposes a black-box zero-resource approach for detecting LLM hallucinations, an approach that mitigates the LLM overconfidence phenomenon by integrating multi-perspective

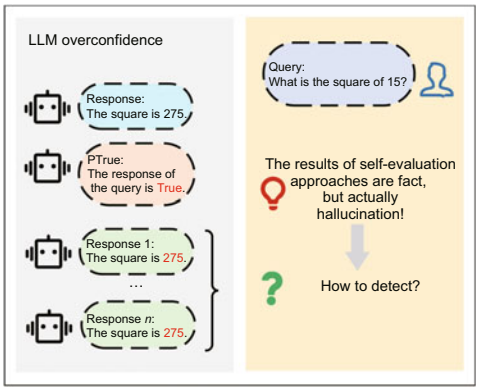

Fig. 2 LLM overconfidence via self-evaluation approaches. 1 denotes the first sampling, and n denotes the n^{th} sampling

consistency scores, thereby improving the detection accuracy. As illustrated in Fig. 3, the approach contains three parts. (1) Inference sampling: At this stage, the LLM generates the response to be evaluated based on the user query and conducts multiple samplings of the same query. Concurrently, using the same LLM and the original response, multiple samplings are also made to reconstruct the queries. The stage derives a set of samples for both the sampling responses and the reconstructed queries. (2) Consistency score calculation: The sampled queries and responses are semantically encoded first. Then, the single-perspective consistency scores of queries and responses are computed via cosine similarity. (3) Multi-perspective consistency checking: The consistency scores from the query and response perspectives are effectively integrated to collectively detect fact or hallucination. We summarize the main contributions of this paper as follows:

1. We construct two hallucination datasets on different closed-domain tasks, which are used to validate the effectiveness of our proposed approach on multiple LLMs.

2. We propose a novel multi-perspective consistency checking approach for LLM hallucination detection that counteracts the LLM overconfidence phenomenon and achieves superior performance compared with other baselines.

3. Inspired by Monte Carlo (MC) dropout in deep learning, we dynamically alter the LLM temperature coefficients and compare and analyze the results under variable and constant temperature coefficients to explore the influence of implicit uncertainty inherent in LLMs.

2 Related works

This paper primarily involves the research technical aspects of LLM hallucination detection and the consistency score between texts. Consequently, we will mainly discuss the current research approaches of those two technical points.

2.1 LLM hallucination detection

LLMs face catastrophic hallucinations that usually go unnoticed by users. This phenomenon hinders the implementation in high-stakes downstream tasks (Mündler et al., 2024). Zhang Y et al. (2023) explored the sources of hallucinations across the entire lifecycle of LLMs, including pre-training, fine-tuning, reinforcement learning from human feedback (RLHF), and inference. They found that hallucinations are likely to occur during the inference process, particularly when LLMs lack relevant knowledge, produce incorrect information due to memory errors, or overestimate their own capabilities. For LLM hallucination detection, Li JY et al. (2023) proposed HaluEval, a hallucination evaluation dataset generated by ChatGPT. They used elaborately designed prompts to guide four LLMs, i.e., GPT-3, text-davinci-002, text-davinci-003, and gpt-3.5-turbo to

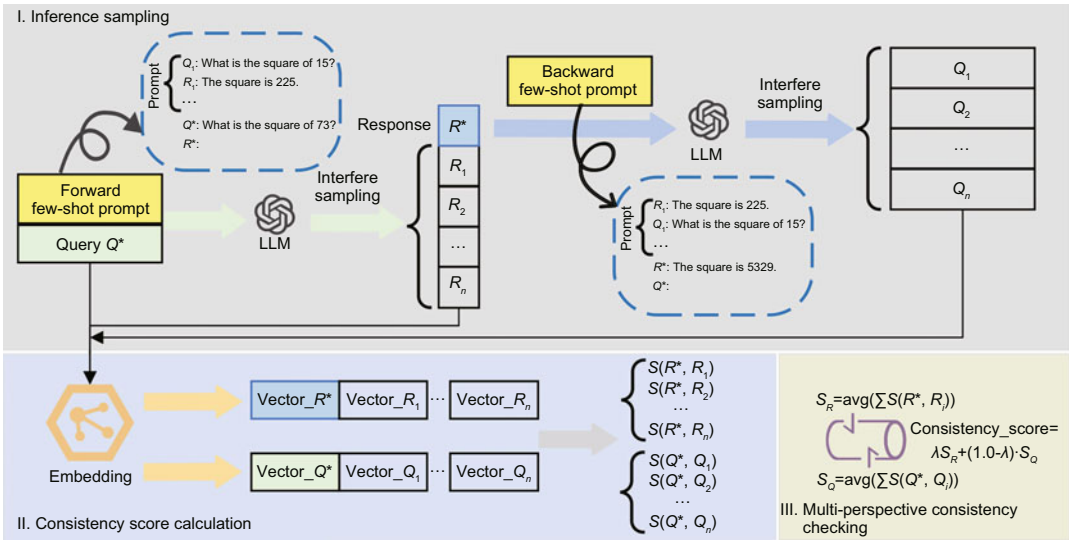

Fig. 3 Multi-perspective consistency checking for LLM hallucination detection

detect hallucinated responses within the HaluEval dataset. Meanwhile, Chern et al. (2023) linked the concept of tool use (such as Google search engine, Google Scholar, code interpreters, or even LLMs) with fact-checking to develop a unified and general framework called FacTool for cross-domain and cross-task fact verification. However, those approaches often require the use of search engines or extra powerful LLMs, which requires internet connectivity and incurs huge API usage costs. To address those issues, Du et al. (2024) trained a classifier that eliminates data collection and manual annotation, thus saving both time and high costs. However, they assumed that LLMs are fully white-box, raising a barrier for non-expert users. Due to the significant complexity of training a classifier based on the internal states, Manakul et al. (2023) proposed the first zero-resource and black-box LLM hallucination detection approach, which self-evaluates whether GPT-3-generated articles in the WikiBio dataset contain hallucinated content. However, it also overestimates the ability of the LLM, known as LLM overconfidence.

The above approaches either require external resources or training a classifier, impeding LLMs deployed at the edge. In addition, current LLM self-evaluation approaches using only a single perspective are apt to be overconfident when exposed to hallucinations. Therefore, we propose a black-box zero-resource approach, leveraging multi-perspective consistencies to check hallucinations. The proposed approach can be applied to scenarios where there is limited access to resources and models, while mitigating the overconfidence of single-perspective approaches.

2.2 Consistency scores

Consistency scores are represented by the semantic similarity between responses, and commonly used approaches involve encoding the text and then calculating the cosine similarity between responses. Zhang TY et al. (2020) introduced the transformer-based BERTScore, which first semantically encodes the sentences and then computes the cosine similarity between the embedded vectors, addressing the limitations of n-gram matching in capturing semantics and long-distance dependencies. Reimers and Gurevych (2019) proposed the improved sentence-BERT (SBERT) based on BERT (Devlin et al., 2019), which optimizes the sentence embedding and semantic similarity calculation tasks, making it more suitable for determining the semantic similarity between sentences. In addition to numerical representation of consistency, Honovich et al. (2022) used a natural language inference (NLI) model (DeBERTa-v3-large used in Honovich et al. (2022)) and input sampled responses and the response to be evaluated; the model directly outputs entailment, neutral, or contradiction labels. Similarly, Fu et al. (2024) leveraged GPT-4 to evaluate various metrics of generated responses and used prompts to guide GPT-4 to directly output the consistency scores between the original responses and the samples. However, NLI models with good performance and GPT-4 both have many parameters and require significant computational resources, incurring high expenses. Consequently, we use SBERT to calculate consistency scores between queries and responses. The SBERT model requires less time with a smaller number of parameters compared with others.

3 Multi-perspective consistency checking for LLM hallucination detection

The contents generated by LLMs often appear plausible and fluent but contain hallucinations; end-users do not always recognize this, usually accept them as facts, and make incorrect decisions for downstream tasks (Lin et al., 2024). Consequently, we propose a black-box zero-resource approach, leveraging multi-perspective consistencies to check hallucinations and ascertain whether the response is factual or hallucinated. Notably, we treat the hallucination detection process as the subject of interrogation. We repeatedly pose the same query to the LLM, collecting multiple responses (Zhang XY et al., 2024). To mitigate overconfidence, where LLMs consistently provide incorrect responses to the same query, we employ the reconstruction of the original query multiple times based on the originally generated response, a process akin to questioning a criminal to determine whether he/she is lying in a typical interrogation scenario. Subsequently, we use the encoder SBERT (Reimers and Gurevych, 2019) to encode both the collected queries and responses, and then calculate the consistency scores between them. Finally, we implement multi-perspective consistency checking by integrating the consistency scores of the queries and responses to determine whether

the response to be evaluated is factual or hallucinated. This can mitigate overconfidence and enhance the performance of our hallucination detection approach from multi-perspective consistency checks. The complete implementation process of the multi-perspective consistency checking for LLM hallucination detection is shown in Algorithm 1.

Algorithm 1 Multi-perspective consistency checking for LLM hallucination detection

Input: LLMs, SBERT, Query Q, prompt, fusion coefficient λ, τ
Output: Fact or hallucination label

Step 1: generate the original response R^*
$R^* = \text{LLM}(\text{prompt}, Q)$
Step 2: inference sampling
for $i = 1$ to N_f and $j = 1$ to N_b **do**
  $R_i = \text{LLM}(\text{prompt}_\text{f}, Q)$
  $Q_j = \text{LLM}(\text{prompt}_\text{b}, R^*)$
end for
Step 3: consistency score calculation
$\boldsymbol{V}_{R^*} = \text{SBERT}(R^*)$, $\boldsymbol{V}_Q = \text{SBERT}(Q)$
$S_R = 0$, $S_Q = 0$
for $i = 1$ to N_f **do**
  $\boldsymbol{V}_{R_i} = \text{SBERT}(R_i)$
  $S_{R_i} = \text{cosine}(\boldsymbol{V}_{R^*} \cdot \boldsymbol{V}_{R_i})$
  $S_R = S_R + S_{R_i}$
end for
for $i = 1$ to N_b **do**
  $\boldsymbol{V}_{Q_i} = \text{SBERT}(Q_i)$
  $S_{Q_i} = \text{cosine}(\boldsymbol{V}_Q \cdot \boldsymbol{V}_{Q_i})$
  $S_Q = S_Q + S_{Q_i}$
end for
$S_R = \frac{S_R}{N_\text{f}}$, $S_Q = \frac{S_Q}{N_\text{b}}$
Step 4: multi-perspective consistency
Consistency_score $= \lambda S_R + (1.0 - \lambda) \cdot S_Q$
if Consistency_score $> \tau$ **then**
  Fact
else
  Hallucination
end if

3.1 Problem setup

Given the prompts and query Q, LLMs generate the corresponding response R^* based on their internal parameters and knowledge, adhering to the format specified by the prompt instructions, which is expressed in Eq. (1):

$$R^* = \text{LLM}(\text{prompt}, Q), \tag{1}$$

where the response R^* may fall into one of three categories: completely factual, completely hallucinated, or a mix of both facts and hallucinations. In this paper, the latter two categories are both considered as hallucinations. Therefore, the task at hand is to detect whether the LLM-generated response R^* is factual or hallucinated.

3.2 Inference sampling

To detect the hallucination in response R^*, we conduct N_f forward inference samplings using the same query Q, yielding N_f responses, which is expressed in Eq. (2):

$$R_i = \text{LLM}(\text{prompt}_\text{f}, Q),\ i \in [1, N_\text{f}], \tag{2}$$

where i denotes the i^{th} inference and N_f represents the total number of forward samplings. prompt_f is the forward prompt that instructs the LLM to generate responses.

Similarly, we conduct N_b backward inference samplings using the response to be evaluated R^*, yielding N_b reconstructed queries, which is expressed in Eq. (3):

$$Q_i = \text{LLM}(\text{prompt}_\text{b}, R^*),\ i \in [1, N_\text{b}], \tag{3}$$

where i denotes the i^{th} backward inference and N_b represents the total number of backward samplings. prompt_b indicates the backward prompt that instructs the LLM to generate queries.

Additionally, motivated by MC dropout in deep learning models, we also vary the temperature coefficients in both inference processes to incorporate the uncertainty inherent in the LLMs. We dynamically modulate the temperature coefficients to increase with the number of sampling iterations, which is expressed in Eq. (4):

$$T_i = \frac{T_0 N + (1.0 - T_0) \cdot i}{N}, \tag{4}$$

where T_i represents the temperature coefficient at the i^{th} LLM inference, T_0 represents the initial value of default temperature, and N represents the number of sampling iterations. As the temperature coefficient increases, the generated results become more creative and diverse, resulting in a higher likelihood that the generated contents deviate from the factual ones.

3.3 Consistency score calculation

Given the original query Q and the sampled queries Q_i, as well as the response to be evaluated R^* and the sampled responses R_i, we first encode them using the SBERT model, which are expressed in Eqs. (5) and (6):

$$\boldsymbol{V}_R=\begin{cases}\text{SBERT}(R^*),\\ \text{SBERT}(R_i),\ i\in[1,N_\text{f}],\end{cases} \tag{5}$$

$$\boldsymbol{V}_Q=\begin{cases}\text{SBERT}(Q),\\ \text{SBERT}(Q_i),\ \ i\in[1,N_\text{b}].\end{cases} \tag{6}$$

Subsequently, we employ cosine similarity to calculate the consistency scores between them individually. According to the number of sampling iterations, we obtain multiple consistency scores, which are expressed in Eqs. (7) and (8):

$$S_{R_i}=\text{cosine}(\boldsymbol{V}_{R^*}\cdot\boldsymbol{V}_{R_i}), i\in[1,N_\text{f}], \tag{7}$$

$$S_{Q_i}=\text{cosine}(\boldsymbol{V}_{Q}\cdot\boldsymbol{V}_{Q_i}), i\in[1,N_\text{b}], \tag{8}$$

where $\text{cosine}(\boldsymbol{A}\cdot\boldsymbol{B})$ represents the cosine similarity between vectors $\boldsymbol{A}$ and $\boldsymbol{B}$, and S_{R_i} and S_{Q_i} represent consistency scores of the i^th response and query, respectively.

3.4 Multi-perspective consistency checking

To integrate the consistency scores from multiple perspectives, we first determine the average of the consistency scores for both the queries and responses, which are expressed in Eqs. (9) and (10):

$$S_R=\frac{1}{N_\text{f}}\sum_{i=1}^{N_\text{f}}S_{R_i}, \tag{9}$$

$$S_Q=\frac{1}{N_\text{b}}\sum_{i=1}^{N_\text{b}}S_{Q_i}, \tag{10}$$

where S_R and S_Q denote the average consistency scores of responses and queries, respectively. In this paper, for the convenience of implementation and operation in limited-resource scenarios, we directly linearly integrate the multi-perspective consistency scores following Sadat et al. (2023), which is expressed in Eq. (11):

$$\text{Consistency\_score}=\lambda S_R+(1.0-\lambda)\cdot S_Q. \tag{11}$$

4 Experiment

To validate the effectiveness of our proposed multi-perspective consistency checking for hallucination detection, we create two datasets based on two types of tasks and conduct experimental validation on them, notably because most current datasets are composed of open-domain questions, which are not applicable to our settings. Also, those datasets are generated by ChatGPT or GPT-3, which are not suitable for scenarios we aim to research in this paper. Therefore, we deliberately design prompts and queries based on the different tasks and use three smaller-parameter LLMs (i.e., Llama2-7B, Qwen2-7B, and Mistral-7B) to validate our approach. Importantly, our approach does not require the retrieval of any external knowledge, making it more applicable for scenarios where LLMs are deployed at the edge and have limited access to resources.

4.1 Dataset

We concentrate on two distinct tasks: the knowledge question–answer (QA) task in the closed domain and the basic arithmetic task. The knowledge QA dataset is designed based on the Books dataset from Kaggle, whereas the arithmetic dataset is constructed based on the square number calculation of natural numbers.

The Books dataset: The Books dataset is composed of over 200 000 literary books from Amazon and has been used for diverse data analyses and research purposes. We randomly select 1000 books from the dataset that are published before the LLMs are trained. Each item record includes the book title, authors, publisher, and publication year. We use the book titles as the premises on which to query LLMs about the authors and publication year, yielding 1000 corresponding generated responses to be evaluated. Subsequently, to fit the mode of daily use, we meticulously design prompts and use few-shot to instruct the generated content.

The Square Num dataset: The main task involves calculating the arithmetic square of a random natural number. Given the ability of smaller-parameter LLMs, the random numbers are constrained to the integral numbers between 1 and 500; hence, the dataset contains 500 queries and 500 corresponding responses generated by LLMs.

We manually annotate the LLM-generated responses as either facts or hallucination labels, and the hallucination rates of the three LLMs on two different tasks are presented in Table 1. It is observed that different LLMs exhibit varying hallucination rates across different datasets, further emphasizing the importance of detecting factual or hallucinated content. In addition, users are guided by the detection results of the responses to make further decisions on the downstream tasks, which will help catch errors early in the process. Therefore, high accuracy of the results is of paramount importance, which is also our aim.

Table 1 Hallucination rates of LLMs on different datasets

LLM	Hallucination rate (%)	
	Books	Square Num
Llama2-7B	94.7	90.2
Qwen2-7B	95.3	11.4
Mistral-7B	92.9	78.2

4.2 Implementation details

In all the experiments in this paper, we maintain N_{f} and N_{b} with a value of 5. The initial sampling temperature coefficient T_0 is set to 0.6 by default. The embedding model employed in our approach is the SBERT. Consistency scores between queries (or responses) are obtained by calculating the cosine similarity of the encoded vectors. The parameter λ to fuse the multi-perspective consistency scores is set to 0.8. It is noteworthy that the datasets generated by various LLMs are all inferred on a single A800 graphics processing unit (GPU). Generating the Books dataset takes approximately 10 h on one LLM, whereas generating the mathematics arithmetic benchmark takes about 4.5 h on one LLM, with a total time expenditure of roughly 2.5 d on all three LLMs.

4.3 Evaluation metric

As defined in Section 3.1, we categorize the detection results into two classes: fact or hallucination. Consequently, the evaluation metrics for hallucination detection approaches are the same as those used for binary classification problems. As indicated in Table 1, the proportion of factual and hallucinated responses on the two datasets is unevenly distributed. Therefore, we employ two common evaluation metrics, balanced accuracy (B-ACC) and area under the curve (AUC) (Manakul et al., 2023), to assess all the hallucination detection approaches. Notably, the higher the values of B-ACC and AUC, the better the performance of LLM hallucination detection approaches.

4.4 Hallucination detection results

To demonstrate the effectiveness of our multi-perspective consistency checking approach, we compare it with five baselines, i.e., the SBERT_cosine approach and four single-perspective approaches, PTrue (Kadavath et al., 2022), Verbalized (Tian et al., 2023), forward-perspective (SelfCheckGPT) (Manakul et al., 2023), and the backward-perspective approach. Because our approach is zero-resource and black-box, we only compare similar baselines. The SBERT_cosine approach directly calculates the consistency score between the query and the response to determine whether it is factual or hallucinated. The prompt instructions used in PTrue (Kadavath et al., 2022) and Verbalized (Tian et al., 2023) are described in Tables A1 and A2 in Appendix A, respectively.

We employ Bootstrap (Efron and Tibshirani, 1993) resampling (number of samples $B = 1000$) to estimate the uncertainty of performance metrics and assess the statistical significance tests between our approach and other baselines, with false discovery rate (FDR) correction for multiple comparisons. As shown in Table 2, our approach shows great improvement both in B-ACC and AUC over most baselines. Compared to SelfCheckGPT (the best baseline), our improvement of 2.3 percentage points (PPs) in B-ACC is statistically significant ($p < 0.001$ after FDR correction), with a 95% confidence interval of [2.1 PPs, 2.5 PPs] for the Books dataset on Llama2-7B. Our approach improves the AUC by 0.007 for the Square Num dataset on Llama2-7B. We see that the hallucination rate of Qwen2-7B on the Square Num dataset is rather low, with only 11.4% from Table 1; however, the B-ACC and AUC of all single-perspective approaches are markedly less than our approach by 2.2 PPs and 0.004, respectively, again elucidating the shortcomings of those single-perspective hallucination detection approaches. The advantage of our approach is that it regards the detection problems from multiple perspectives, which diminishes the overconfidence and overestimation of

Table 2 Hallucination detection results on all LLMs and datasets with $N_f = N_b = 5$ and under a constant temperature coefficient

LLM	Approach	Books		Square Num	
		B-ACC (%) ↑	AUC ↑	B-ACC (%)↑	AUC ↑
Llama2-7B	SBERT_cosine	49.8±0.1	0.497±0.031	50.0±0.0	0.583±0.037
	PTrue	53.1±0.4	0.531±0.004	62.4±1.0	0.624±0.010
	Verbalized	54.0±2.8	0.634±0.031	**66.5±3.7**	0.691±0.042
	Forward (SelfCheckGPT)	75.5±2.4	0.828±0.018	61.0±3.6	0.720±0.037
	Backward	70.9±3.4	0.732±0.039	51.7±1.4	0.624±0.039
	Multi-perspective consistency	**77.8±2.4**	**0.832±0.019**	61.4±3.6	**0.727±0.036**
Qwen2-7B	SBERT_cosine	50.0±0.0	0.523±0.037	50.0±0.0	0.494±0.038
	PTrue	53.8±0.4	0.538±0.004	49.9 ±0.1	0.499±0.001
	Verbalized	53.1±2.1	0.535±0.026	56.3±3.4	0.590±0.039
	Forward (SelfCheckGPT)	80.9±2.0	**0.869±0.016**	70.1±3.4	0.838±0.028
	Backward	58.7±3.3	0.724±0.043	57.9±3.2	0.560±0.035
	Multi-perspective consistency	**81.8±2.3**	0.866±0.017	**72.3±3.4**	**0.842±0.026**
Mistral-7B	SBERT_cosine	48.9±0.2	0.540±0.032	50.0±0.0	0.407±0.033
	PTrue	56.4±1.3	0.564±0.014	**66.2±1.6**	0.662±0.016
	Verbalized	48.7±0.2	0.490±0.010	58.2±2.6	0.560±0.036
	Forward (SelfCheckGPT)	76.1±1.8	**0.814±0.018**	60.2±2.3	0.659±0.032
	Backward	65.8±2.9	0.719±0.030	62.7±2.6	0.662±0.033
	Multi-perspective consistency	**77.2±1.8**	0.804±0.018	61.3±2.3	**0.666±0.032**

Best B-ACC and AUC values are in bold. ↑ represents a higher value indicating better performance

those single-perspective approaches, thereby improving the performance of hallucination detection.

4.5 Ablation study

We conduct an ablation study to examine the impact of different inference iterations (N_f and N_b), the influence of temperature coefficients in Eq. (4) during the LLM inference process, and the effect of λ in Eq. (11) for the multi-perspective consistency scores in the proposed approach in the following subsections.

4.5.1 Inference iterations

We evaluate the impact of different values of inference iterations (i.e., different values of N_f and N_b) on the performance of our approach. The results on the Books and Square Num datasets are presented in Table 3 and Table B1 in Appendix B, respectively. Specifically, we dynamically assess N_f and N_b ranging from 1 to 5 (higher values of N_f and N_b correspond to larger costs of computational resources). Again, we set $N_f = N_b$ in all detection experiments, replacing with N_{both} for ease of description. Table 3 shows that setting $N_{both} > 1$ is crucial for the proposed approach across all three LLMs on the Books dataset. Notably, the best results are consistently obtained with $N_{both} = 5$, yielding the best detection performance in most cases.

Therefore, we conclude that increasing the number of inference iterations improves the detection results and performance, but this comes with additional computational costs for inference. The trade-off between performance and computational costs is an important consideration when deploying the multi-perspective consistency checking approach in practical applications.

4.5.2 Temperature coefficients

We evaluate the impact of temperature coefficients during the inference processes on the performance of our approach, with the results on the Books and Square Num datasets presented in Table 4. Specifically, we consider the temperature coefficients as constant and variable temperatures. The constant temperature refers to the temperature coefficient that is the same, by default, across all sampling iterations, whereas variable temperatures increase with the number of sampling iterations in Eq. (4). Table 4 reveals that the hallucination detection performance on most LLMs is better with a constant temperature compared to that with a variable temperature. Further analysis suggests that

Table 3 Results for the Books dataset, with N_{both} values ranging from 1 to 5, and under a constant temperature coefficient

LLM	$N_{\text{both}}=1$		$N_{\text{both}}=2$		$N_{\text{both}}=3$		$N_{\text{both}}=4$		$N_{\text{both}}=5$	
	B-ACC (%)	AUC	B-ACC (%)	AUC	B-ACC (%)	AUC	B-ACC (%)	AUC	B-ACC (%)	AUC
Llama2-7B_Multi-perspective consistency	**77.9**	0.819	77.4	0.832	75.1	0.825	76.4	0.830	77.5	**0.833**
Qwen2-7B_Multi-perspective consistency	79.8	0.858	81.1	0.868	83.6	**0.871**	82.0	0.864	**84.1**	0.865
Mistral-7B_Multi-perspective consistency	70.0	0.796	76.9	0.804	**78.3**	0.807	78.0	**0.808**	78.0	0.806

Best B-ACC and AUC values are in bold

Table 4 Results for the Books and Square Num datasets, with $N_{\text{both}}=5$, under both constant and variable temperature coefficient

LLM	Books				Square Num			
	Constant temperature		Variable temperature		Constant temperature		Variable temperature	
	B-ACC (%)	AUC	B-ACC (%)	AUC	B-ACC (%)	AUC	B-ACC (%)	AUC
Llama2-7B_Multi-perspective consistency	**77.5**	0.833	77.2	**0.843**	**63.0**	**0.733**	59.6	0.723
Qwen2-7B_Multi-perspective consistency	**84.1**	**0.865**	80.3	0.858	72.1	**0.840**	**79.8**	0.835
Mistral-7B_Multi-perspective consistency	78.0	0.806	**78.5**	**0.817**	**61.3**	**0.670**	58.2	0.585

Better B-ACC and AUC values are in bold

higher inference temperatures lead to more creative responses, which causes the responses to be too divergent from the original response, potentially mislabeling factual responses as hallucinations. It is also the reason that a constant temperature coefficient is used in SelfCheckGPT (Manakul et al., 2023).

However, from the perspective of evaluating the uncertainty of responses, variable temperature takes into account the uncertainty inherent in LLMs (Liu et al., 2025), which helps the user assess the reliability of responses (Huang et al., 2024) and will be explored further in our future work.

4.5.3 Fusion coefficients

Finally, we evaluate the impact of different parameters λ to fuse multi-perspective consistency scores on the performance of our approach, with the results on the Books and Square Num datasets presented in Fig. 4 and Fig. C1 in Appendix C, respectively. Specifically, let $\lambda \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$ for the convenience of implementation and assess the detection results for fusing multi-perspective consistency scores based on variable λ. As shown in Fig. 4, for most LLMs, B-ACC and AUC are superior when $\lambda = 0.8$ compared with others. It is also seen in Table 2 that the hallucination detection results via SelfCheckGPT outperform the backward approach, which means the forward perspective is given more weight than the backward perspective. Consequently, the best result gives more weight to the forward-perspective consistency scores when λ is set to 0.8, while the backward-perspective approach, which serves as an auxiliary, is also essential. In summary, hallucination detection using multi-perspective consistency checking is indispensable and supportive of improvement in detection accuracy.

5 Conclusions

In this paper, we propose a black-box zero-resource approach for LLM hallucination detection via multi-perspective consistency checking and fusing. We treat the hallucination detection process as the subject of interrogation by repeatedly posing the same query to the LLM and collecting multiple responses. To mitigate overconfidence, we reconstruct of the original query multiple times based on the original response, similar to questioning about

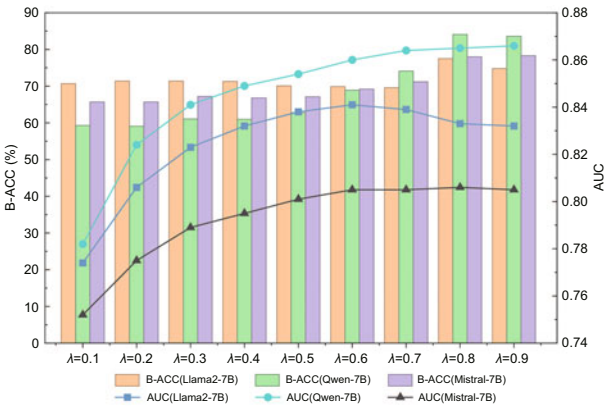

Fig. 4 Results for the Books dataset, with different λ's, and under the constant temperature coefficient

lying. Then we fuse the multi-perspective interrogation results to obtain a more accurate detection result. Compared with other single-perspective approaches, our approach demonstrates superior performance across multiple datasets and LLMs. Notably, in one experiment where the hallucination rate reached 94.7%, our approach improves B-ACC by 2.3 PPs compared with the single consistency approach and achieves an AUC of 0.832, all without depending on any external resources.

However, there are still some limitations to be addressed in our future work. (1) In the experiment, we use only two available perspectives to detect hallucinations. However, the workflow is similar when extending more perspectives, which will be explored in our next work step. (2) Additionally, a broader suite of consistency metrics will be investigated to further demonstrate the generalization of our approach. (3) Ultimately, to enhance performance, more non-linear models to fuse consistency scores will be further explored in our subsequent work.

Contributors

Linggang KONG designed the research. Linggang KONG and Haoran FU processed the data. Linggang KONG drafted the paper. Xiaofeng ZHONG helped organize the paper. Jie CHEN and Yongjie WANG revised and finalized the paper.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Cheng FP, Zouhar V, Arora S, et al., 2024. RELIC: investigating large language model responses using self-consistency. Proc CHI Conf on Human Factors in Computing Systems, Article 647. https://doi.org/10.1145/3613904.3641904

Cheng XX, Li JY, Zhao WX, et al., 2024. Small agent can also rock! Empowering small language models as hallucination detector. Proc Conf on Empirical Methods in Natural Language Processing, p.14600-14615. https://doi.org/10.18653/v1/2024.emnlp-main.809

Chern IC, Chern S, Chen SQ, et al., 2023. FacTool: factuality detection in generative AI—a tool augmented framework for multi-task and multi-domain scenarios. https://arxiv.org/abs/2307.13528

Devlin J, Chang MW, Lee K, et al., 2019. BERT: pre-training of deep bidirectional transformers for language understanding. Proc Conf of the North American Chapter of the Association for Computational Linguistics, p.4171-4186. https://doi.org/10.18653/v1/N19-1423

Du XF, Xiao CW, Li YX, 2024. HaloScope: harnessing unlabeled LLM generations for hallucination detection. Proc 38th Int Conf on Neural Information Processing Systems, Article 3270.

Efron B, Tibshirani RJ, 1993. An Introduction to the Bootstrap. Chapman & Hall, New York, USA.

Farquhar S, Kossen J, Kuhn L, et al., 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625-630. https://doi.org/10.1038/s41586-024-07421-0

Fu JL, Ng SK, Jiang ZB, et al., 2024. GPTScore: evaluate as you desire. Proc Conf of the North American Chapter of the Association for Computational Linguistics, p.6556-6576. https://doi.org/10.18653/v1/2024.naacl-long.365

Guan XY, Liu YJ, Lin HY, et al., 2024. Mitigating large language model hallucinations via autonomous knowledge graph-based retrofitting. Proc 38th Annual AAAI Conf on Artificial Intelligence, p.18126-18134. https://doi.org/10.1609/aaai.v38i16.29770

Honovich O, Aharoni R, Herzig J, et al., 2022. TRUE: re-evaluating factual consistency evaluation. Proc Conf of the North American Chapter of the Association for Computational Linguistics, p.3905-3920. https://doi.org/10.18653/v1/2022.naacl-main.287

Hu XM, Zhang YM, Peng R, et al., 2024. Embedding and gradient say wrong: a white-box method for hallucination detection. Proc Conf on Empirical Methods in Natural Language Processing, p.1950-1959. https://doi.org/10.18653/v1/2024.emnlp-main.116

Huang XM, Li S, Yu MX, et al., 2024. Uncertainty in language models: assessment through rank-calibration. Proc Conf on Empirical Methods in Natural Language Processing, p.284-312. https://doi.org/10.18653/v1/2024.emnlp-main.18

Kadavath S, Conerly T, Askell A, et al., 2022. Language models (mostly) know what they know. https://arxiv.org/abs/2207.05221

Li JY, Cheng XX, Zhao WX, et al., 2023. HaluEval: a large-scale hallucination evaluation benchmark for large language models. Proc Conf on Empirical Methods in Natural Language Processing, p.6449-6464. https://doi.org/10.18653/v1/2023.emnlp-main.397

Li JY, Chen J, Ren RY, et al., 2024. The dawn after the dark: an empirical study on factuality hallucination in large language models. Proc 62nd Annual Meeting of the Association for Computational Linguistics, p.10879-10899. https://doi.org/10.18653/v1/2024.acl-long.586

Li TJ, Li Z, Zhang Y, 2024. Improving faithfulness of large language models in summarization via sliding generation and self-consistency. Proc Joint Int Conf on Computational Linguistics, p.8804-8817.

Liang X, Song SC, Zheng ZF, et al., 2024. Internal consistency and self-feedback in large language models: a survey. https://arxiv.org/abs/2407.14507

Lin ZC, Guan SY, Zhang WD, et al., 2024. Towards trustworthy LLMs: a review on debiasing and dehallucinating in large language models. *Artif Intell Rev*, 57(9):243. https://doi.org/10.1007/s10462-024-10896-y

Liu HF, Huang HG, Gu XM, et al., 2025. On calibration of LLM-based guard models for reliable content moderation. Proc 13th Int Conf on Learning Representations.

Luo YW, Yang Y, 2024. Large language model and domain-specific model collaboration for smart education. *Front Inform Technol Electron Eng*, 25(3):333-341. https://doi.org/10.1631/FITEE.2300747

Manakul P, Liusie A, Gales M, 2023. SelfCheckGPT: zero-resource black-box hallucination detection for generative large language models. Proc Conf on Empirical Methods in Natural Language Processing, p.9004-9017. https://doi.org/10.18653/v1/2023.emnlp-main.557

Mündler N, He JX, Jenko S, et al., 2024. Self-contradictory hallucinations of large language models: evaluation, detection and mitigation. Proc 12th Int Conf on Learning Representations.

Reimers N, Gurevych I, 2019. Sentence-BERT: sentence embeddings using Siamese BERT-networks. Proc Conf on Empirical Methods in Natural Language Processing and the 9th Int Joint Conf on Natural Language Processing, p.3982-3992. https://doi.org/10.18653/v1/d19-1410

Sadat M, Zhou ZY, Lange L, et al., 2023. DelucionQA: detecting hallucinations in domain-specific question answering. Proc Findings of the Association for Computational Linguistics, p.822-835. https://doi.org/10.18653/v1/2023.findings-emnlp.59

Tian K, Mitchell E, Zhou A, et al., 2023. Just ask for calibration: strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. Proc Conf on Empirical Methods in Natural Language Processing, p.5433-5442. https://doi.org/10.18653/v1/2023.emnlp-main.330

Verspoor K, 2024. 'Fighting fire with fire'—using LLMs to combat LLM hallucinations. *Nature*, 630(8017):569-570. https://doi.org/10.1038/d41586-024-01641-0

Wan FQ, Huang XT, Cui LY, et al., 2024. Knowledge verification to nip hallucination in the bud. Proc Conf on Empirical Methods in Natural Language Processing, p.2616-2633. https://doi.org/10.18653/v1/2024.emnlp-main.152

Zhang JX, Li ZH, Das K, et al., 2023. SAC^3: reliable hallucination detection in black-box language models via semantic-aware cross-check consistency. Proc Findings of the Association for Computational Linguistics, p.15445-15458. https://doi.org/10.18653/v1/2023.findings-emnlp.1032

Zhang SL, Yu T, Feng Y, 2024. TruthX: alleviating hallucinations by editing large language models in truthful space. Proc 62nd Annual Meeting of the Association for Computational Linguistics, p.8908-8949. https://doi.org/10.18653/v1/2024.acl-long.483

Zhang TY, Kishore V, Wu F, et al., 2020. BERTScore: evaluating text generation with BERT. Proc 8th Int Conf on Learning Representations.

Zhang XY, Peng BL, Tian Y, et al., 2024. Self-alignment for factuality: mitigating hallucinations in LLMs via self-evaluation. Proc 62nd Annual Meeting of the Association for Computational Linguistics, p.1946-1965. https://doi.org/10.18653/v1/2024.acl-long.107

Zhang Y, Li YF, Cui LY, et al., 2023. Siren's song in the AI ocean: a survey on hallucination in large language models. https://arxiv.org/abs/2309.01219

Appendix A: Prompts

1. PTrue prompts

Table A1 PTrue prompt instructions

Provide a True or False response on whether the answer to the following query is correct. Give only a True or False answer, no other words or explanation.
The question is: {query} The answer is: {response} The answer is:

2. Verbalized prompts

Table A2 Verbalized prompt instructions

Provide the probability between 0.0 and 1.0 that the answer for the following query is correct. Give only the probability value between 0.0 and 1.0, no other words or explanation.
The question is: {query} The answer is: {response} The probability of the answer is correct:

Appendix B: More results for variable inference iterations

Table B1 demonstrates the performance of our proposed multi-perspective consistency checking approach on the Square Num dataset with variable N_{both} values. As can be seen, higher values of N_{both} result in larger B-ACC and AUC values.

Appendix C: More results for variable fusion coefficients

As depicted in Fig. C1, the values of B-ACC and AUC on the Square Num dataset are larger when λ is set to 0.8.

Table B1 Results for the Square Num dataset with N_{both} values ranging from 1 to 5

LLM	$N_{\text{both}}=1$	
	B-ACC (%)	AUC
Llama2-7B_Multi-perspective consistency	70.3	0.745
Qwen2-7B_Multi-perspective consistency	68.2	0.739
Mistral-7B_Multi-perspective consistency	62.2	0.650
LLM	$N_{\text{both}}=2$	
	B-ACC (%)	AUC
Llama2-7B_Multi-perspective consistency	**71.4**	**0.791**
Qwen2-7B_Multi-perspective consistency	**72.3**	0.796
Mistral-7B_Multi-perspective consistency	60.3	0.622
LLM	$N_{\text{both}}=3$	
	B-ACC (%)	AUC
Llama2-7B_Multi-perspective consistency	61.4	0.725
Qwen2-7B_Multi-perspective consistency	68.2	0.809
Mistral-7B_Multi-perspective consistency	62.4	0.658
LLM	$N_{\text{both}}=4$	
	B-ACC (%)	AUC
Llama2-7B_Multi-perspective consistency	61.2	0.724
Qwen2-7B_Multi-perspective consistency	70.4	0.830
Mistral-7B_Multi-perspective consistency	**62.5**	0.663
LLM	$N_{\text{both}}=5$	
	B-ACC (%)	AUC
Llama2-7B_Multi-perspective consistency	63.0	0.733
Qwen2-7B_Multi-perspective consistency	72.1	**0.840**
Mistral-7B_Multi-perspective consistency	61.3	**0.670**

Best B-ACC and AUC values are in bold

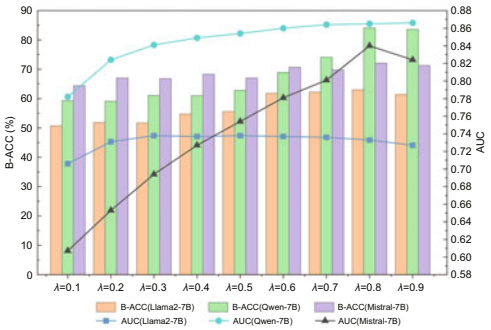

Fig. C1 Results for the Square Num dataset with different λ's and under a constant temperature coefficient