Prompt-Guided Internal States for Hallucination Detection of Large Language Models

Anonymous ACL submission

Abstract

Large Language Models (LLMs) have demonstrated remarkable capabilities across a variety of tasks in different domains. However, they sometimes generate responses that are logically coherent but factually incorrect or misleading, which is known as LLM hallucinations. Data-driven supervised methods train hallucination detectors by leveraging the internal states of LLMs, but detectors trained on specific domains often struggle to generalize well to other domains. In this paper, we aim to enhance the cross-domain performance of supervised detectors with only in-domain data. We propose a novel framework, prompt-guided internal states for hallucination detection of LLMs, namely PRISM. By utilizing appropriate prompts to guide changes to the structure related to text truthfulness in LLMs' internal states, we make this structure more salient and consistent across texts from different domains. We integrated our framework with existing hallucination detection methods and conducted experiments on datasets from different domains. The experimental results indicate that our framework significantly enhances the cross-domain generalization of existing hallucination detection methods[1].

1 Introduction

In recent years, Large Language Models (LLMs) have demonstrated remarkable capabilities across a variety of tasks in different domains (Dinan et al., 2019; Brown et al., 2020; Zhang et al., 2022; Chowdhery et al., 2023; Touvron et al., 2023a). However, the hallucination problem in LLMs poses a potential threat to their practical application in many scenarios. LLM hallucinations refer to the cases where LLMs generate responses that are logically coherent but factually incorrect or misleading (Zhou et al., 2021; Ji et al., 2023; Su et al., 2024). These hallucinated responses may be blindly accepted, leading users to learn incorrect information or take inappropriate actions. Therefore, detecting hallucinations in LLM-generated content becomes particularly meaningful. By using hallucination detectors to help identify incorrect generated content, users can be alerted to verify the accuracy of LLMs' responses, thus preventing potential issues.

The unsupervised paradigm focuses on assessing the confidence of LLM-generated content and rejects the low-confidence outputs. For example, the probability information of each token generated by LLMs can serve as a measure of hallucination (Kadavath et al., 2022; Zhang et al., 2023; Quevedo et al., 2024; Hou et al., 2024). Additionally, the consistency among multiple responses generated by LLMs to the same question can also be used to evaluate their confidence (Manakul et al., 2023). Furthermore, Chen et al. (2024) shifts the consistency judgment from multiple responses to their corresponding activation values in LLMs. However, these methods often struggle to achieve ideal detection accuracy or require a significant amount of additional response time (Su et al., 2024).

For this reason, researchers have begun exploring the use of data-driven supervised methods for hallucination detection. These methods are generally based on the assumption that LLMs can recognize they have generated incorrect content, which is reflected in specific patterns in their internal states. Marks and Tegmark (2023) reveal that true and false statements have discernible geometric structures in LLMs' internal states, allowing us to build classifiers by learning this structure. Marks and Tegmark (2023) and CH-Wang et al. (2024) utilize linear structures to distinguish between different statements, while Azaria and Mitchell (2023) train neural networks to act as hallucination detectors.

Supervised detectors trained on specific domains often struggle to achieve good generalization per-

[1]We have open-sourced all the code and data in GitHub: https://anonymous.4open.science/r/PRISM-7E2B

formance in other domains. For example, Bürger et al. (2024) found that when distinguishing between true and false statements, the structural differences in LLMs' internal states are significantly different for affirmative and negated sentences. To address this issue, many studies focus on constructing diverse datasets or performing feature selection within LLMs' internal states to achieve better generalization performance (Chen et al., 2023; Bürger et al., 2024; Liu et al., 2024). However, these methods require collecting additional training data from other domains, which is resource-intensive. In this paper, we aim to answer the following question:

Can we enhance the cross-domain performance of supervised detectors with only in-domain data?

Driven by this research question, we propose a novel framework called PRISM, which stands for prompt-guided internal states for hallucination detection of LLMs. By utilizing appropriate prompts to guide changes to the structure related to text truthfulness in LLMs' internal states, we make this structure more salient and consistent across texts from different domains, which enables detectors trained in one domain to also perform well in others without additional data. Our approach is based on the insight that while LLMs' internal states encode rich semantic information, they are primarily optimized during pre-training to predict the next token rather than for hallucination detection. As a result, the directly extracted internal states contain lots of domain-specific information that is not related to text truthfulness, leading to detectors that are specific to certain domains and unable to achieve good cross-domain generalization performance.

Due to the powerful ability of LLMs to understand and follow instructions, we explore the use of prompt-guided methods to generate internal states more focused on text truthfulness. We employed various methods to investigate the effect of prompts from different perspectives. The findings indicate that the introduction of appropriate prompts can significantly improve the salience of the structure related to text truthfulness in LLMs' internal states and make this structure more consistent across different domain datasets. We also provide a simple and effective method to generate and select appropriate prompts for hallucination detection tasks. By combining prompt templates with the text to be evaluated and inputting them into LLMs, we can obtain internal states that are better suited as features for hallucination detection tasks. Subsequently, we can integrate the prompt-guided internal states with existing hallucination detection methods to construct more advanced detectors.

In summary, the contributions of this paper are as follows: (1) We conduct an in-depth investigation into how specific prompts guide the change to the structure related to text truthfulness in LLMs' internal states. (2) We propose PRISM, a novel framework that utilizes prompt-guided internal states to enhance hallucination detection of LLMs. (3) We integrate PRISM with existing hallucination detection methodologies, demonstrating its ability to significantly enhance the generalization performance of detectors across different domains.

2 Preliminary

In our research, to study the cross-domain hallucination detection problem, we utilized several publicly available datasets that encompass data from various domains. The first dataset we used is the True-False dataset (Azaria and Mitchell, 2023), which consists of six sub-datasets: "animals," "cities," "companies," "elements," "facts," and "inventions." These sub-datasets share similar textual structures but contain content on different topics. Each sub-dataset includes almost the same number of true and false statements, such as: 'Meta Platforms has headquarters in the United States' and 'Silver is used in catalytic converters and some electronic components.' More detailed information about this dataset can be found in Appendix A.

Another dataset we used is from Bürger et al. (2024) and consists of 24 sub-datasets with 6 different topics: "animal_class," "cities," "inventors," "element_symb," "facts," and "sp_en_trans," as well as 4 different grammatical structures: affirmative statements, negated statements, logical conjunctions, and logical disjunctions. Affirmative statements were structured similarly to the examples in the True-False dataset, while negated statements were formed by negating the affirmative statements using the word "not." The sentences in logical conjunctions and logical disjunctions were constructed by sampling sentences from the affirmative statements and then connecting them with "and" or "or." The number of true and false statements within each grammatical structure is balanced. For clarity, we refer to this dataset as LogicStruct throughout this paper and present additional information about it in Appendix A.

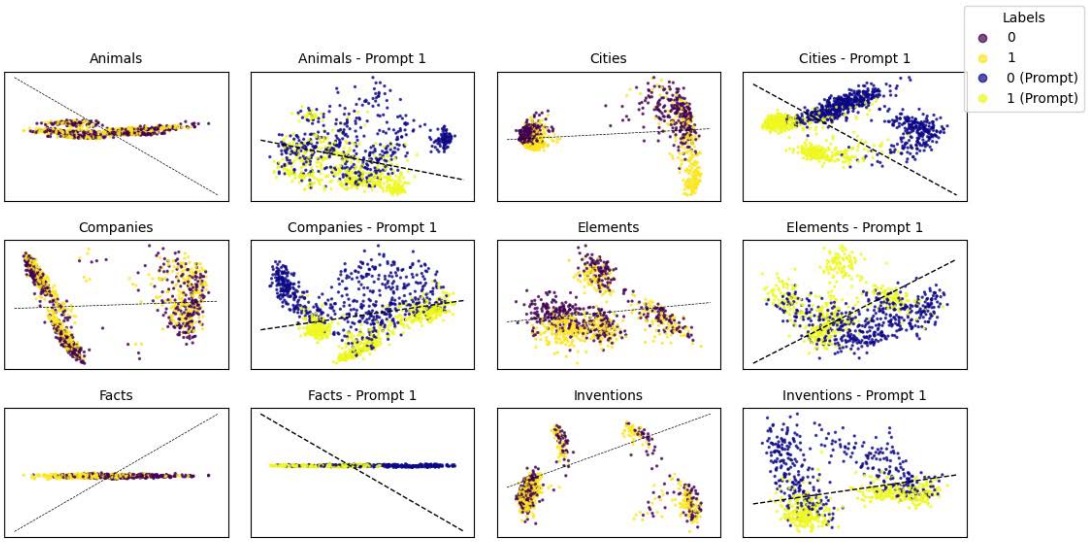

Figure 1: Visualization of the LLaMA2-7B-Chat model's internal states by using the 2-dimensional PCA on the True-False dataset before and after the introduction of Prompt 1, where label 0 represents false statements and label 1 represents true statements. Additionally, a logistic regression model was fitted to distinguish between true and false statements, with the decision boundary shown as a black dashed line.

3 Prompt-Guided Internal States

Several previous studies have focused on leveraging LLMs' internal states for hallucination detection (Azaria and Mitchell, 2023; Liu et al., 2024; Su et al., 2024). They assume that LLMs can recognize they have generated incorrect content, which is reflected in specific patterns in their internal states. Therefore, these studies typically select contextualized embeddings corresponding to specific tokens in certain layers of LLMs, and then use them as features to train hallucination detectors. However, these embeddings were originally intended to guide text generation and they encode various information of the related text. As a result, the information related to text truthfulness becomes intertwined with domain-specific details, making it difficult for detectors to identify a consistent structure related to text truthfulness in LLMs' internal states.

Therefore, we hope to guide changes in LLMs' internal states such that the structure related to text truthfulness becomes more salient and consistent across texts from different domains. This would help detectors learn this structure and enhance their generalization performance. To achieve this goal, we experimented with a simple prompt:

```
----------------------------------------------
Prompt 1:
Here is a statement: [statement]
Is the above statement correct?
----------------------------------------------
```

This prompt directly asks LLMs the truthfulness of specific statements. When we input it into LLMs, the generation of new tokens will revolve around this question, which will be reflected in LLMs' internal states.

In the following two subsections, we will use various methods to demonstrate the changes in LLMs' internal states before and after the introduction of Prompt 1 and explain how these changes will effect the hallucination detection task. Our analysis is conducted on the six sub-datasets of the True-False dataset. The feature vectors we used are the embeddings corresponding to the last token in the final layer of LLaMA2-7B-Chat.

3.1 Effect of Prompt on Structural Salience

Some previous studies (Marks and Tegmark, 2023; Bürger et al., 2024) have shown that, on certain datasets, true and false statements have distinguishable geometric structures in LLMs' internal states and these structures can be directly observed after applying Principal Component Analysis (PCA) to reduce the dimensionality of relevant data. Therefore, we utilized the PCA to observe the salience of the structure related to text truthfulness before and after the introduction of Prompt 1.

From Figure 1, we can clearly observe that in all six datasets, the introduction of the prompt enables the first two principal components to effectively

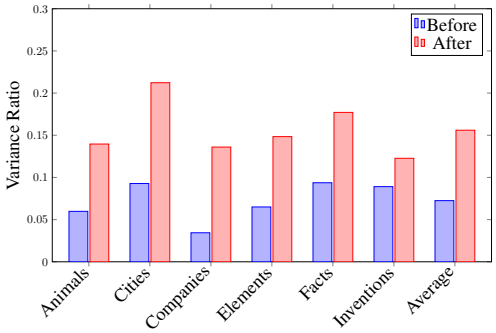

Figure 2: Comparison of the variance ratios before and after the introduction of Prompt 1 on each sub-dataset of the True-False dataset.

distinguish true and false statements. Since the first two principal components represent the directions with the largest variance among the embeddings, this indicates that the introduction of the prompt makes the structure related to text truthfulness in the embeddings more salient.

Moreover, to accurately describe the differences in structural salience, we performed a more detailed variance analysis. We believe that the proportion of the variance along a particular direction relative to the total variance can reflect the salience of the structure in that direction. Therefore, we subtracted the average embedding of false statements from the average embedding of true statements within each dataset, and referred to this direction as the 'truthfulness direction'. This was first used by Marks and Tegmark (2023) and its formula is as follows:

$$\theta = \frac{1}{N^+}\sum_{i=1}^{N^+} v_i^+ - \frac{1}{N^-}\sum_{i=1}^{N^-} v_i^-, \tag{1}$$

where θ represents the truthfulness direction in the dataset D, v_i^+ and v_i^- denote the embeddings corresponding to true and false statements, respectively, and N^+ and N^- represent their counts.

Then we performed variance analysis along this direction. Let v_i represent the embedding corresponding to a specific statement within dataset D, and let N denote the total number of statements. By calculating

$$\hat{v_i} = v_i - \frac{1}{N}\sum_{j=1}^{N} v_j, \tag{2}$$

we can use the de-centered vectors $\hat{v_i}$ to form a data matrix $X = (\hat{v_1}, \hat{v_2}, \ldots, \hat{v_N})^T$, and then compute the covariance matrix corresponding to X:

$$\Sigma = \frac{1}{N-1} X^T X. \tag{3}$$

In this way, we obtain the total variance among all embeddings in dataset D:

$$V_T = \text{Trace}(\Sigma) \tag{4}$$

and the variance along the truthfulness direction:

$$V_\theta = \frac{\theta^T \Sigma \theta}{\|\theta\|^2}. \tag{5}$$

Finally, we can use the ratio

$$R = \frac{V_\theta}{V_T} \tag{6}$$

to represent the salience of the structure related to text truthfulness in dataset D. We calculated the corresponding variance ratios on the six sub-datasets of the True-False dataset and presented the results in Figure 2.

We can observe that after the introduction of the prompt, this ratio shows a significant increase for all sub-datasets, indicating that the differences among the embeddings begin to concentrate more along the truthfulness direction. This enhances the salience of the structure related to text truthfulness in LLMs' internal states, making it easier for hallucination detectors to learn.

3.2 Effect of Prompt on Structural Consistency

When we aim to apply detectors trained in one domain to others, the consistency of this structure becomes particularly important. A more consistent structure will lead to better generalization performance. Therefore, we continued to utilize the 'truthfulness direction' defined by Equation (1) to analyze the consistency of this structure across different datasets and use the cosine similarity to represent this consistency:

$$c_{ij} = \frac{\theta_i \cdot \theta_j}{\|\theta_i\|\|\theta_j\|}, \tag{7}$$

where c_{ij} represents the cosine of the angle between two truthfulness directions.

The calculation results for the six sub-datasets of the True-False dataset are presented in Table 1. We can see that the introduction of the prompt significantly increases the cosine similarity between the truthfulness directions across different datasets, indicating that the structure related to text truthfulness between different datasets becomes more consistent. This consistency will help improve the generalization performance of related detectors, which use LLMs' internal states as input features to determine the truthfulness of statements.

Table 1: The cosine similarity between the truthfulness directions across different topics in the True-False dataset, before and after Prompt 1 addition.

	Cosine Similarity Between Truthfulness Directions						Cosine Similarity After **Prompt 1** Addition					
Datasets	**Animals**	**Cities**	**Comp.**	**Elements**	**Facts**	**Invent.**	**Animals**	**Cities**	**Comp.**	**Elements**	**Facts**	**Invent.**
Animals	1.0000	0.4368	0.4668	0.5498	0.6950	0.3786	1.0000	0.8037	0.7589	0.7284	0.8345	0.8333
Cities	0.4368	1.0000	0.4122	0.3300	0.4707	0.2520	0.8037	1.0000	0.8349	0.7253	0.7274	0.8540
Comp.	0.4668	0.4122	1.0000	0.4302	0.5691	0.4102	0.7589	0.8349	1.0000	0.7717	0.7255	0.8210
Elements	0.5498	0.3300	0.4302	1.0000	0.6258	0.3301	0.7284	0.7253	0.7717	1.0000	0.8175	0.8071
Facts	0.6950	0.4707	0.5691	0.6258	1.0000	0.4757	0.8345	0.7274	0.7255	0.8175	1.0000	0.7950
Invent.	0.3786	0.2520	0.4102	0.3301	0.4757	1.0000	0.8333	0.8540	0.8210	0.8071	0.7950	1.0000
Average	0.5878	0.4836	0.5481	0.5443	0.6394	0.4744	**0.8265**	**0.8242**	**0.8187**	**0.8083**	**0.8167**	**0.8517**

4 Methodology

The analyses in sections 3 indicate that the introduction of appropriate prompts can guide changes to the structure related to text truthfulness in LLMs' internal states, and these changes will facilitate our use of LLMs' internal states as features to perform the hallucination detection task. Based on this insight, we propose a novel framework **PRISM**, i.e., PRompt-guided Internal States for hallucination detection of Large Language Models. Our framework consists of two components: First,

Our framework consists of two components:

First, we generate a large number of candidate prompt templates and select the one that best fits the current task. We manually construct a prompt template P related to the hallucination detection task, and then we use the language model L to generate multiple candidate prompt templates $\{P_i\}_{i=1,...,N}$ that are similar in meaning but different in structure. For each P_i, we combine it with the text a_j from the labeled dataset A to obtain the text $P_i(a_j)$, and then input it into L to obtain the corresponding feature vector v_{ij} in the internal states. Subsequently, we use all feature vectors $\{v_{ij}\}_{j=1,...,M}$ to calculate the structure salience index R_i corresponding to P_i on A and select the prompt template with the highest index as our final prompt template P_s.

Next, we integrate the prompt-guided internal states with existing hallucination detection methods to construct a more advanced detector. We train the hallucination detector D on the dataset A using all feature vectors $\{v_{sj}\}_{j=1,...,M}$ and their labels. For new text T to be detected, we similarly combine it with the prompt template P_s to obtain the text $P_s(T)$ and input it into the language model L to obtain the corresponding feature vector v. Finally, we input v into the hallucination detector D to obtain its hallucination label H.

In the following subsections, we will provide a more detailed explanation of some key steps.

4.1 Prompt Generation

To obtain suitable candidate prompt templates for the hallucination detection task, we can use LLMs to assist in this process. First, we manually construct a simple prompt, such as Prompt 1 in Section 3. Next, we use this prompt to construct the following one:

```
------------------------------------------------
Prompt 2:
"Here is a statement: '[statement]'
Is the above statement correct?"
This is a universal prompt template. Please
generate templates with similar meanings but
diverse forms. The template should include the
embedding position of [statement].
------------------------------------------------
```

Finally, we input the above prompt into LLMs to obtain a large number of prompt templates similar to Prompt 1. For example, we used Prompt 2 to query GPT-4o, resulting in 10 prompt templates that have similar meanings but different formats. The relevant content is provided in Appendix C.

4.2 Prompt Selection

Once we have a large number of suitable prompt templates, we need to select the most appropriate one among them to perform our hallucination detection task. A simple and effective method is to calculate the variance ratio corresponding to each prompt template on the labeled dataset using Equation (6) and then leverage this ratio to guide our prompt selection. A higher ratio indicates that the prompt is more likely to achieve better results.

It is worth noting that this process relies on the selection of feature vectors from the internal states of LLMs. We selected the contextualized embedding corresponding to the last token in the final layer of LLaMA2-Chat-7B as the feature vector and calculated the corresponding variance ratio for each of the 10 prompt templates in Appendix C. We computed the variance ratios for each prompt template across the six sub-datasets of the True-False

Table 2: Average variance ratios for the 10 candidate prompt templates along with their ranking.

Prompt	I	II	III	IV	V	VI	VII	VIII	IX	X	Without
Ratio	0.1093	0.1312	0.1291	0.1407	0.0875	0.1165	0.1041	0.1265	0.1228	**0.1513**	0.0725
Ranking	8	3	4	2	10	7	9	5	6	**1**	11

dataset, averaged the results, and ranked them, as shown in Table 2.

According to Table 2, the 10th prompt template achieved the highest variance ratio and we present its form below:

```
-----------------------------------------------
Prompt 3:
Does the statement '[statement]' accurately
reflect the truth?
-----------------------------------------------
```

We selected this prompt template for our framework in the experiments presented in Section 5. We also conducted the same analysis on this prompt as we did in Section 3 and presented the results in Appendix B.

5 Experiment

5.1 Datasets and Metric

We evaluate our PRISM and other baselines on the True-False and LogicStruct datasets which are detailed in section 2. Since the ratio of the two classes of labels in these datasets is close to 1:1, classification accuracy is an appropriate metric.

5.2 Baselines

We choose the following methods as baselines:

- **LN-PP** (Manakul et al., 2023) detects hallucinations based on the average probabilities of tokens generated by LLMs. We first determine a threshold, and then use it for label classification.

- **SAPLMA** (Azaria and Mitchell, 2023) is an effective supervised learning method. It uses activation values in LLMs as feature vectors to train a Multilayer Perceptron (MLP) network as the hallucination detector.

- **MIND** (Su et al., 2024) is an unsupervised learning framework. It automatically generates labeled datasets from Wikipedia articles for training the hallucination detector. Other settings are consistent with the SAPLMA method.

- **MM**(Marks and Tegmark, 2023) uses a simple mass-mean probe to distinguish between true and false statements. This method selects specific activation values as feature vectors and uses Equation (1) to calculate the truthfulness direction on the labeled dataset. For the text to be detected, it first projects the feature vector onto the truthfulness direction by taking the dot product, then applies the sigmoid function, and finally rounds the result to obtain the label.

- **PRISM**. Our framework selects Prompt 3 from Section 4.2 as the prompt template, and integrates with the SAPLMA and MM methods to obtain the PRISM-SAPLMA and PRISM-MM methods.

5.3 Implementation Details

We conducted our experiments using LLaMA2-7B-Chat and LLaMA2-13B-Chat (Touvron et al., 2023b). The activation values were selected from the contextualized embeddings corresponding to the last token in the final layer of the LLMs. In the SAPLMA and MIND methods, the dimensions of the detectors for each layer were 256, 128, 64, and 2, with a dropout rate of 20% applied in the first layer. The activation function was ReLU, and the optimizer was Adam. These settings are the same as in the original paper (Su et al., 2024).

Our experiments focused on the generalization performance of different methods across different domains. For the SAPLMA method, we trained the detector separately on each sub-dataset of the True-False dataset and tested it on all other sub-datasets. Thus, for a single sub-dataset, we obtained multiple test results from detectors trained on other sub-datasets, and we averaged these results to obtain the final classification accuracy for this sub-dataset. In the LogicStruct dataset, we concentrated on training with affirmative statements and testing on other grammatical structures, as previous studies have indicated that achieving good generalization results in this scenario is challenging (Marks and Tegmark, 2023; Bürger et al., 2024). Both the MM and LN-PP methods employed the same evaluation approach as the SAPLMA. The MIND method was trained on its automatically generated dataset and tested on each sub-dataset of the True-False and LogicStruct datasets. The settings of the methods under PRISM framework remain consistent with those of the original methods.

For all detectors that required training, we split a validation set from the training set at a 4:1 ratio and

Table 3: The experimental results on the True-False dataset using LLaMA2-7B-Chat.

Baselines	Animals	Cities	Comp.	Elements	Facts	Invent.	Average
LN-PP	0.5496	0.5653	0.5357	0.5727	0.5948	0.5527	0.5618
MIND	0.4626	0.4664	0.4997	0.4850	0.5025	0.5023	0.4864
MM	0.5300	0.5688	0.5648	0.5581	0.6137	0.5075	0.5572
SAPLMA	0.6539	0.6700	0.6262	0.6057	**0.7585**	0.6233	0.6563
PRISM-MM	0.7004	**0.9060**	0.7908	0.6385	0.7031	0.7756	0.7524
PRISM-SAPLMA	**0.7147**	0.8936	**0.8279**	**0.6705**	0.7539	**0.7804**	**0.7735**

Table 4: The experimental results on the True-False dataset using LLaMA2-13B-Chat.

Baselines	Animals	Cities	Comp.	Elements	Facts	Invent.	Average
LN-PP	0.5538	0.5564	0.5397	0.5800	0.5954	0.5612	0.5644
MIND	0.4957	0.4705	0.4372	0.5011	0.4986	0.4502	0.4756
MM	0.5330	0.5366	0.5438	0.5327	0.6196	0.5085	0.5457
SAPLMA	0.6584	0.7065	0.6721	0.6472	**0.8054**	0.6253	0.6858
PRISM-MM	0.7111	0.8837	**0.8498**	**0.7092**	0.7804	0.7993	0.7889
PRISM-SAPLMA	**0.7405**	**0.8960**	0.8435	0.6997	0.7918	**0.8209**	**0.7987**

selected the model parameters with the highest accuracy on the validation set over 10 training epochs as the test parameters. The final results presented are the averages obtained from training under three different random seeds.

5.4 Experimental Results

We can see from Table 3 and 4 that our framework significantly outperforms other baselines in terms of the classification accuracy on the True-False dataset. **On almost every sub-dataset of different topics (5 out of 6 topics), our framework substantially improves the generalization performance of the original methods.** Additionally, there is a significant difference in the performance of the original MM and SAPLMA methods, which may be due to the simpler structure of the detector used in the MM method compared to that in the SAPLMA method. However, after integrating both into our framework, these two methods achieve similar performance. According to Section 3.1, this may be because the introduction of the prompt makes the structure related to text truthfulness more salient in the LLMs' internal states, making it easier for detectors to capture this structure. On the Facts sub-dataset, the original SAPLMA method and our framework achieve nearly identical accuracy. As shown in Figure 1, before the introduction of prompts, the true and false statements in the Facts sub-dataset already exhibit clear separability, so it is difficult to further enhance this structure by appropriate prompts.

Table 5 presents the experimental results on the LogicStruct dataset. We can observe that, except for our framework, other baselines struggle to generalize the training results on affirmative statements to other grammatical structures. According to the observations made by Bürger et al. (2024), this is because the structure related to text truthfulness in affirmative statements is different from the corresponding structure in other grammatical structures. However, **our framework achieves significant generalization performance across different grammatical structures.** This indicates that the introduction of the prompt indeed makes the structure related to text truthfulness more consistent across different datasets, which aligns with the observation in Section 3.2.

5.5 Ablation Studies

5.5.1 Impact of Prompt Selection

In this section, we focus on the impact of prompt selection. We conducted experiments using the PRISM-MM method on all 10 prompt templates presented in Appendix C, with the results presented in Table 6. We can observe that all the prompt templates significantly improved the average accuracy compared to the result of the original MM method. This indicates that the impact of prompts on improving the generalization performance of hallucination detectors is general and stable.

In addition, by combining the ranking information from Table 2 and Table 6, we can see that the top 4 prompt templates selected based on the variance ratios all perform within the top 5 in actual experiments, while the two worst-performing prompt templates correspond to the two lowest variance ratios. These results indicate that the variance ratio

Table 5: The overall experimental results on the LogicStruct dataset.

	LLaMA2-7B-Chat				LLaMA2-13B-Chat			
Baselines	**Neg.**	**Conj.**	**Disj.**	**Average**	**Neg.**	**Conj.**	**Disj.**	**Average**
LN-PP	0.4108	0.5278	**0.5743**	0.5043	0.4263	0.5243	0.5863	0.5123
MIND	0.5219	0.4318	0.4969	0.4835	0.4849	0.4873	0.5038	0.4920
MM	0.4948	0.4937	0.5013	0.4966	0.5071	0.4895	0.5020	0.4995
SAPLMA	0.4864	0.5196	0.5030	0.5030	0.5155	0.6788	0.5186	0.5710
PRISM-MM	0.5548	**0.8087**	0.5580	0.6405	0.5260	**0.8734**	**0.6207**	0.6734
PRISM-SAPLMA	**0.7345**	0.7529	0.5329	**0.6734**	**0.7389**	0.8112	0.5722	**0.7074**

Table 6: The experimental results of all 10 prompt templates on the True-False dataset using LLaMA2-7B-Chat.

Prompt	**I**	**II**	**III**	**IV**	**V**	**VI**	**VII**	**VIII**	**IX**	**X**	**Without**
Acc.	0.7217	0.7678	0.7297	0.7398	0.6997	0.7749	0.6931	0.7162	0.7003	0.7524	0.5572
Ranking	6	2	5	4	9	1	10	7	8	3	11

can help us select prompt templates that achieve better generalization results in actual experiments while avoiding poorly performing ones.

5.5.2 Impact of Layer Selection

In addition to selecting the contextualized embeddings corresponding to the last token in the final layer of LLMs as feature vectors, we also extracted the embeddings corresponding to the last token in the middle layer (16th) of LLaMA2-7B-Chat to evaluate our framework. We conducted the same experiments on the True-False dataset with our framework and the corresponding original methods, as shown in the last four rows of Table 7.

Compared to the results in Table 3, the performance of the original methods has worsened, while our framework has achieved better results. This indicates that our framework exhibits good stability with respect to different layer selections.

5.5.3 Impact of Internal States

After introducing the prompt, is it possible to assess the truthfulness of the text directly based on LLMs' responses without relying on their internal states? To investigate this question, after inputting the prompt into LLaMA2-7B-Chat, we directly extracted the token probabilities for " Yes" and " No" when the LLM was going to generate the next token and calculated their ratio $p[3869]/p[1939]$, where p represents the vocabulary probabilities for generating the next token. We used whether the ratio is greater than 1 to determine the label of the original text. The average accuracy on the True-False dataset is shown in the first row of Table 7.

Table 7: The experimental results on the True-False datase using feature vectors from the middle layer of LLaMA2-7B-Chat and the next token probabilities from LLaMA2-7B-Chat.

Baselines	**Average Accuracy**
Yes/No	0.6166
MM(middle)	0.5166
SAPLMA(middle)	0.5766
PRISM-MM(middle)	0.7924
PRISM-SAPLMA(middle)	**0.7938**

We can observe a significant accuracy difference between directly using the responses generated by the LLM and training detectors based on the prompt-guided internal states. This indicates that the LLM's internal states contain more information related to text truthfulness, which is not observable from the LLM's responses but still plays a crucial role in hallucination detection.

6 Conclusions

In this paper, we first investigated the effect of specific prompts on the structure related to text truthfulness in LLMs' internal states. We found that suitable prompts can make this structure more salient and consistent across datasets from different domains, facilitating hallucination detectors in learning this structure and improving their generalization performance. Based on this insight, we introduced a novel framework, prompt-guided internal states for hallucination detection of large language models, namely PRISM. This framework integrates the prompt-guided internal states with existing hallucination detection methods to obtain more advanced detectors. Finally, we conducted experiments across various baselines on datasets from different domains, and the results demonstrated that our framework can significantly enhance the generalization performance of existing hallucination detection methods.

7 Limitations

We acknowledge that our research still has some limitations. When detecting hallucinations in the content generated by LLMs, we heavily rely on their internal states, which may be challenging to access in some cases, requiring us to use a proxy model. Additionally, our framework does not leverage other information produced by LLMs during text generation, such as probability information and the generated text itself. In future research, we will explore fully utilizing various types of information produced by LLMs during text generation to achieve more effective hallucination detection.

8 Ethics Statement

The development and application of the PRISM framework for Large Language Models hallucination detection are guided by the principle of assessing the reliability and credibility of AI-generated content. PRISM aims to reduce the risk of users being misled by AI-generated content, thereby contributing to safer and more trustworthy AI applications. Additionally, the datasets used in this project are publicly available and do not involve personal or sensitive data, ensuring privacy and security. We need to recognize that our framework classifies text based on the model's internal states, and thus may inherit biases from the language model on certain issues, potentially leading to incorrect judgments. Therefore, we encourage further research and collaboration to develop ethical, fair, and trustworthy AI systems.

References

Amos Azaria and Tom Mitchell. 2023. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, Singapore. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Lennart Bürger, Fred A. Hamprecht, and Boaz Nadler. 2024. Truth is universal: Robust detection of lies in LLMs. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Sky CH-Wang, Benjamin Van Durme, Jason Eisner, and Chris Kedzie. 2024. Do androids know they're only dreaming of electric sheep? In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4401–4420, Bangkok, Thailand. Association for Computational Linguistics.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. 2024. INSIDE: LLMs' internal states retain the power of hallucination detection. In *The Twelfth International Conference on Learning Representations*.

Yuyan Chen, Qiang Fu, Yichen Yuan, Zhihao Wen, Ge Fan, Dayiheng Liu, Dongmei Zhang, Zhixu Li, and Yanghua Xiao. 2023. Hallucination detection: Robustly discerning reliable answers in large language models. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, CIKM '23, page 245–255, New York, NY, USA. Association for Computing Machinery.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents. In *International Conference on Learning Representations*.

Bairu Hou, Yang Zhang, Jacob Andreas, and Shiyu Chang. 2024. A probabilistic framework for llm hallucination detection via belief tree propagation. *arXiv preprint arXiv:2406.06950*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

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

Junteng Liu, Shiqi Chen, Yu Cheng, and Junxian He. 2024. On the universal truthfulness hyperplane inside LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 18199–18224, Miami, Florida, USA. Association for Computational Linguistics.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore. Association for Computational Linguistics.

Samuel Marks and Max Tegmark. 2023. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*.

Ernesto Quevedo, Jorge Yero, Rachel Koerner, Pablo Rivas, and Tomas Cerny. 2024. Detecting hallucinations in large language model generation: A token probability approach. *arXiv preprint arXiv:2405.19648*.

Weihang Su, Changyue Wang, Qingyao Ai, Yiran Hu, Zhijing Wu, Yujia Zhou, and Yiqun Liu. 2024. Unsupervised real-time hallucination detection based on the internal states of large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14379–14391, Bangkok, Thailand. Association for Computational Linguistics.

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

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

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

Tianhang Zhang, Lin Qiu, Qipeng Guo, Cheng Deng, Yue Zhang, Zheng Zhang, Chenghu Zhou, Xinbing Wang, and Luoyi Fu. 2023. Enhancing uncertainty-based hallucination detection with stronger focus. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 915–932, Singapore. Association for Computational Linguistics.

Chunting Zhou, Graham Neubig, Jiatao Gu, Mona Diab, Francisco Guzmán, Luke Zettlemoyer, and Marjan Ghazvininejad. 2021. Detecting hallucinated content in conditional neural sequence generation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1393–1404, Online. Association for Computational Linguistics.

Table 8: The information of the True-False dataset, where T=0 indicates that the sentence is annotated as false, and T=1 indicates true.

Topic	#Sentence	#T=0	#T=1
Animals	1008	504	504
Cities	1458	729	729
Companies	1200	600	600
Elements	930	465	465
Facts	613	307	306
Inventions	876	412	464

A Dataset Details

The following are some examples of statements from the True-False dataset:

- Animals: "The gazelle has distinctive orange and black stripes and is an apex predator."
- Cities: "Oranjestad is a city in Aruba."
- Companies: "Meta Platforms has headquarters in United States."
- Elements: "Silver is used in catalytic converters and some electronic components."
- Facts: "The planet Jupiter has many moons."
- Inventions: "Narinder Singh Kapany invented the programming language (theoretical)."

We can observe that the displayed statements contain both true and false information, and the relevant details are shown in Table 8.

Examples of sentences with different grammatical structures in the LogicStruct dataset are as follows:

- Affirmative statements: "The Spanish word 'dos' means 'enemy'."
- Negated statements: "The Spanish word 'dos' does not mean 'enemy'."
- Logical conjunctions: It is the case both that [statement 1] and that [statement 2].
- Logical disjunctions: It is the case either that [statement 1] or that [statement 2].

In both Logical conjunctions and Logical disjunctions, statement 1 and statement 2 are derived from Affirmative statements and sampled in a way that ensures label balance across each dataset. Additional information is presented in Table 9.

Table 9: The information of the LogicStruct dataset, where {Topic} in the Name refers to "animal_class," "cities," "inventors," "element_symb," "facts," and "sp_en_trans," and #Sentence represents the total number of data across all topics for each grammatical structure.

Name	Description	#Sentence
{Topic}	Affirmative statements	3167
{Topic}_neg	Negated statements	3167
{Topic}_conj	Logical conjunctions	3998
{Topic}_disj	Logical disjunctions	3000

Table 10: The cosine similarity between the truthfulness directions across different topics in the True-False dataset after Prompt 3 addition.

Datasets	Animals	Cities	Comp.	Elements	Facts	Invent.
Animals	1.0000	0.8370	0.8043	0.8144	0.8519	0.8811
Cities	0.8370	1.0000	0.8113	0.7835	0.8570	0.8613
Comp.	0.8043	0.8113	1.0000	0.8012	0.7927	0.8192
Elements	0.8144	0.7835	0.8012	1.0000	0.8327	0.8154
Facts	0.8519	0.8570	0.7927	0.8327	1.0000	0.8522
Invent.	0.8811	0.8613	0.8192	0.8154	0.8522	1.0000
Average	**0.8265**	**0.8242**	**0.8187**	**0.8083**	**0.8167**	**0.8517**

B Analysis of Prompt 3

Consistent with the analysis of Prompt 1 in Section 3, we calculated the variance ratios of Prompt 3 on each sub-dataset of the True-False dataset. The results before and after the introduction of Prompt 3 are presented in Figure 3. We also calculated the cosine similarity between the truthfulness directions across these sub-datasets after the introduction of Prompt 3, as shown in Table 10.

We can observe that Prompt 3 significantly enhances the salience and consistency of the structure related to text truthfulness in the embeddings. This is consistent with the conclusion in Section 3, indicating that using prompts to guide changes in LLMs' internal states is generally effective.

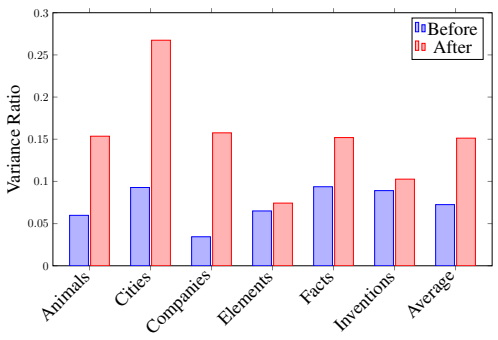

Figure 3: Comparison of the variance ratios before and after the introduction of Prompt 3 on each sub-dataset of the True-False dataset.

C Generation of Prompt Templates

Question:
"Here is a statement: ’[statement]’
Is the above statement correct?"
This is a universal prompt template. Please generate templates with similar meanings but diverse forms. The template should include the embedding position of [statement].
GPT-4o:
Here are various templates with diverse forms that ask about the correctness or validity of a given statement:
1. "Is it accurate to say that ’[statement]’?"
2. "Would you consider the statement ’[statement]’ to be correct?"
3. "Can we confirm that ’[statement]’ is true?"
4. "Does the statement ’[statement]’ hold true?"
5. "Is ’[statement]’ a valid statement?"
6. "Is there accuracy in the claim ’[statement]’?"
7. "Could ’[statement]’ be considered a factual statement?"
8. "Is it correct to assume that ’[statement]’ is true?"
9. "Would it be right to say ’[statement]’ is accurate?"
10. "Does the statement ’[statement]’ accurately reflect the truth?"
Each version maintains the focus on verifying the truth or accuracy of the statement in question.