UNDERSTANDING THE SOURCES OF UNCERTAINTY FOR LARGE LANGUAGE AND MULTIMODAL MODELS

Ziran Yang[♣] **Shibo Hao**[♣] **Hao Sun**[♢] **Lai Jiang**[♣] **Qiyue Gao**[♣] **Yian Ma**[♣] **Zhiting Hu**[♣]
[♣]UC San Diego [♢]Cambridge University

ABSTRACT

Understanding and quantifying uncertainty in large model predictions is critical for their safe and trustworthy deployment. However, existing methods that estimate the overall prediction uncertainty often fail due to miscalibration like model overconfidence. Uncertainty decomposition provides a way to focus on some specific parts in total uncertainty, removing those unrelated components. Traditional uncertainty decomposition into epistemic (model-related) and aleatoric (data-related) components is insufficient for current model usage, as additional factors like prompt phrasing and context significantly influence the model's predictions and add the source of uncertainty. We introduce a unified uncertainty decomposition framework that systematically separates uncertainty contributed by various factors such as prompting, context, and preprocessing of multimodal inputs. By quantifying each component's uncertainty, our approach identifies which uncertainty terms are well-correlated with the model's hallucination rates, thereby enhancing hallucination detection and model improvement. We validate our framework through applications in visual question answering and math reasoning, demonstrating that effective uncertainty components can serve as metrics for hallucination detection and improve model performance through self-training. Grounded in information theory and highly extensible, our framework provides a novel perspective on uncertainty decomposition in large language and multimodal models, offering valuable insights for future research.

1 INTRODUCTION

Large language models (LLMs) and multimodal models, while very capable, are prone to generating hallucinations (Bender et al., 2021; Huang et al., 2023; Dziri et al., 2024). Understanding what these models do not know or are uncertain about is important for detecting hallucinations and further safe and trustworthy deployment. Much work has focused on quantifying LLM uncertainty and using it to identify when the model's outputs are likely to be incorrect or hallucinated (Xiong et al., 2023; Kadavath et al., 2022; Farquhar et al., 2024; Kuhn et al., 2023; Hou et al., 2024). However, in real-world applications, existing methods to quantify the model's prediction uncertainty are not always reliable, e.g. the models may produce incorrect predictions very confidently, which we refer to as overconfidence (Xiong et al., 2023; Groot & Valdenegro-Toro, 2024; Yang et al., 2024). Previous work on uncertainty-based hallucination detection often overlooks or explicitly excludes this issue (Farquhar et al., 2024).

Decomposing uncertainty offers a path to more reliable estimates by attributing the prediction uncertainty to different possible sources (Liu et al., 2019a; Der Kiureghian & Ditlevsen, 2009). Traditional methods commonly decompose prediction uncertainty (or *total uncertainty*) into *epistemic uncertainty* (model's knowledge) and *aleatoric uncertainty* (inherent data randomness) components (Hüllermeier & Waegeman, 2021; Schweighofer et al., 2023); Focusing on the epistemic component rather than the total uncertainty finds success in many applications, as it excludes the irreducible data-contributed part (Charpentier et al., 2022; Osband et al., 2023; Hou et al., 2024; Ling et al., 2024). However, this formulation is limited for foundation models. On one side, measuring epistemic uncertainty in a pretrained model is challenging. On the other side, the basic decomposition does not capture new factors that affect predictions in foundation models. For example, LLM outputs can be heavily influenced by the phrasing of the prompt, the choice of in-context examples,

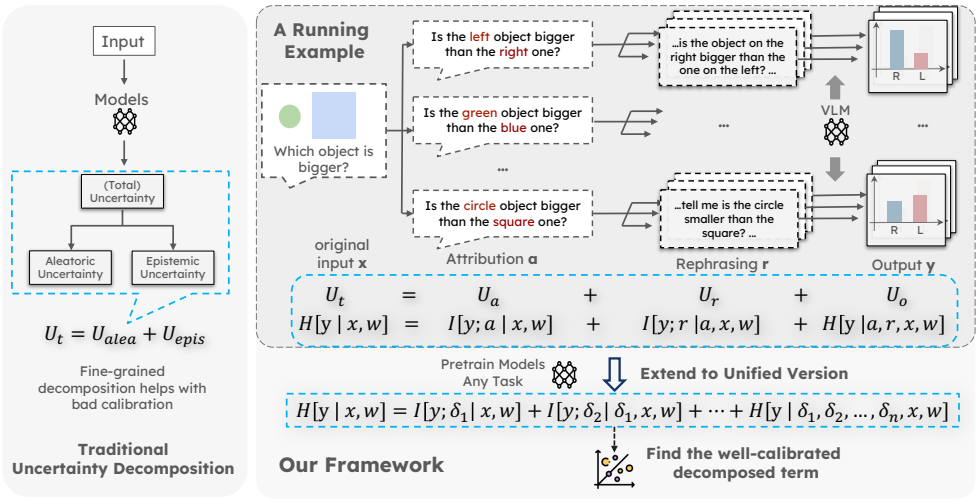


Figure 1: Outline of the paper. Traditional methods divide uncertainty into epistemic and aleatoric components (Sec. 2.1). Our framework starts with a running example (Sec. 2.2): a vision language model is queried to compare the size of two objects in the image, where the attribution used to name the objects and the phrasing of the query alternate. We decompose total uncertainty (U_t) into those contributed by attribution (U_a), rephrasing (U_r), and remaining part (U_o). Furthermore, it can be extended to a unified version of any factors δ influencing model predictions. Based on the decomposition identify those "effective calibrators" — uncertainty terms that strongly and positively correlates with error rates — which can improve model calibration and reliability(Sec. 2.3).

or how objects are referred to in the input (Wei et al., 2023; Dong et al., 2022; Wu et al., 2024). This motivates us to consider the extra uncertainty introduced by such factors.

Moving beyond the traditional aleatoric-epistemic dualism, we introduce a systematic framework for uncertainty decomposition in large language and multimodal models (Sec. 2). Our framework breaks down the total uncertainty, which is quantified by the entropy of the model's output distribution, into a sequence of mutual information between different factors, each of which measures the reduction in uncertainty the factor contributes to the model's output. An example is illustrated in Fig. 1 while the framework is also flexible enough to be extensive settings. With the new framework, we are able to precisely attribute different sources of uncertainty in the generation. And the explicit attribution allows us to obtain a better uncertainty measure, because we find certain decomposed components are more calibrated than the overall uncertainty, where calibration refers to the agreement between the estimated uncertainty and the model's actual performance (Wang, 2023; Zhang et al., 2020).

Our experiments (Sec. 3) demonstrate that certain decomposed components more reliably predict hallucinations. We call these components "effective calibrators." By focusing on them, we can have better hallucination detection accuracy and also improve the model's performance using self-training approaches (Huang et al., 2022; Mukherjee & Awadallah, 2020; Yu et al., 2022). We showcase this in both visual question-answering and reasoning tasks and observe that our decomposition gives insights into why total uncertainty is not always well correlated with true error rates. We believe this framework offers a useful perspective on how to measure and utilize uncertainty in the next generation of large language and multimodal models. We believe that these findings provide a new perspective on uncertainty quantification in the foundation model and could provide valuable insights for future research.

2 General Uncertainty Decomposition for Foundation Models

We propose a framework to decompose the uncertainty in foundation models with respect to the contributions of various factors. This section begins with an overview of traditional approaches to uncertainty quantification (Section 2.1) and introduces our new formulation (Section 2.2). We then explain the implications and applications of the methods, with a focus on their use for hallucination detection (Section 2.3). Finally, we conclude by discussing related works within the context of our framework (Section 2.4).

2.1 BACKGROUND

Uncertainty is typically decomposed into *aleatoric* and *epistemic* uncertainty (Hora, 1996; Der Kiureghian & Ditlevsen, 2009; Hüllermeier & Waegeman, 2021). *Aleatoric uncertainty* reflects the inherent randomness in data. For instance, in a coin flip, even the best model cannot provide a deterministic prediction due to the stochastic nature. In contrast, *epistemic uncertainty* arises from the model's limited knowledge, often resulting from insufficient data or imperfect learning. This type of uncertainty can in principle be reduced by incorporating additional information or data.

This decomposition is frequently discussed in the context of Bayesian inference, where the model parameters θ are treated as a probability distribution. Given a training dataset $\mathcal{D}$, an input x, the posterior predictive distribution is $p(y \mid x, \mathcal{D}) = \mathbb{E}_{\theta \sim p(\theta|\mathcal{D})} p(y \mid x, \theta)$. In this setting, the total uncertainty—quantified as the entropy of the posterior predictive distribution—can be decomposed into two components mathematically (Der Kiureghian & Ditlevsen, 2009; Schweighofer et al., 2023):

$$\underbrace{\mathbb{H}[y \mid x, \mathcal{D}]}_{\text{Total Uncertainty}} = \underbrace{\mathbb{H}[y \mid x, \theta, \mathcal{D}]}_{\text{Aleatoric Uncertainty}} + \underbrace{I[y; \theta \mid x, \mathcal{D}]}_{\text{Epistemic Uncertainty}} \tag{1}$$

The conditional entropy $\mathbb{H}[y \mid x, \theta, \mathcal{D}] = \mathbb{E}_{\theta \sim p(\theta|D)} \mathbb{H}[y \mid x, \theta]$ quantifies the uncertainty in $\mathbf{y}$ that remains even if the realization of $\boldsymbol{\theta}$ is known. As such, it is used to represent *aleatoric uncertainty*, which captures the inherent randomness in the data. The conditional mutual information $I(y; \theta \mid x, \mathcal{D})$, proposed as a measure of *epistemic uncertainty*, is a symmetric metric that quantifies the expected information gained about one variable by observing the other. Intuitively, it reflects the potential reduction in uncertainty about y obtained by observing θ.

Recent works have adapted uncertainty decomposition to inference-time LLM settings, such as considering in-context examples (Ling et al., 2024) or adding a clarification step to the input (Hou et al., 2024). For example, in the clarification setting, an ambiguous question like x = "`In the image, is this object larger than another?`" can be clarified by a clarification c = "`The question refers to the red object.`" The authors draws an analogy to quantify the uncertainty in ambiguous inputs, categorizing those irreducible to clarification c as "aleatoric" uncertainty and the remaining as "epistemic" part:

$$\underbrace{\mathbb{H}[y \mid x, w]}_{\text{Total Uncertainty}} = \underbrace{\mathbb{H}[y \mid c, x, w]}_{\text{"Epistemic" Uncertainty}} + \underbrace{I[y; c \mid x, w]}_{\text{"Aleatoric" Uncertainty}} \tag{2}$$

Partitioning total uncertainty into the component that is reducible by clarification c ($I[y; c \mid x, w]$) and the remaining uncertainty given the clarification ($\mathbb{H}[y \mid c, x, w]$).

While inspiring, these work still rely on the traditional *epistemic-aleatoric* dualistic perspective. As the use cases for LLMs involve a broader range of influencing factors beyond a clarification step or in-context examples, we propose a more general framework and decomposing uncertainty based on the specific variables that contribute to it. And later we will show that these previous works can be explained as special cases of our proposed framework.

2.2 GENERAL UNCERTAINTY DECOMPOSITION FRAMEWORK

Running Example. We start with the running example in Fig.1 with two intermediate variables: attribution a and rephrasing r. We will explain in detail what a and r represent and the insights behind them in Sec.3; for now, you can consider a and r as two factors that modify the query's wording without changing the actual question. We quantify the uncertainty contributed by each variables by:

$$\underbrace{\mathbb{H}[y \mid x, w]}_{U_t\text{: total uncertainty}} = \mathbb{H}[y \mid a, x, w] + I[y; a \mid x, w] \tag{3}$$

$$= \underbrace{\mathbb{H}[y \mid r, a, x, w]}_{U_o\text{: observed uncertainty}} + \underbrace{I[y; r \mid a, x, w]}_{U_r\text{: contributed by rephrasing } r} + \underbrace{I[y; a \mid x, w]}_{U_a\text{: contributed by attribution } a} \tag{4}$$

Here, we adapt the similar equation to first decompose the contribution from variable a (Eq. 3). Intuitively, the uncertainty introduced by a variable is quantified by how much it influences the model's predictions. In this context, mutual information $I[y; a \mid x, w]$ captures how much additional

certainty is gained in predicting y once the variable a is known. The term $\mathbb{H}[y \mid a, x, w]$ captures the remaining unpredictability of y even when a is known. Thus we decompose this term to find the uncertainty contributed by other variables without the influence of a.

Then, considering the variable r, we decompose the uncertainty further (Eq. 4). The uncertainty contributed by r is $U_r = I[y; r \mid a, x, w]$, which measures how much additional certainty is gained by knowing r along with a, x, and w. The term $U_a = I[y; a \mid x, w]$ quantifies the uncertainty due to attribution a, reflecting how a reduces uncertainty about y given x and w. The residual uncertainty, $U_o = \mathbb{H}[y \mid r, a, x, w]$, represents the unpredictability of y when both a and r are known.

Unified Formulation. To generalize, we propose a unified framework that can handle multiple intermediate variables. Let y represent the model's prediction, x denote the input, w indicate the model. These factors are considered intermediate variables that influence the model's prediction, without altering the desired output (such as the ground truth answer in a QA task). We denote these intermediate variables as $\delta_{1:n} = \delta_1, \delta_2, \ldots, \delta_n$. The total uncertainty can then be decomposed using the chain rule of mutual information:

$$\mathbb{H}[y \mid x, w] = \mathbb{H}[y \mid \delta_{1:n}, x, w] + \sum_{i=1}^{n} I[y; \delta_i \mid \delta_{1:i-1}, x, w] \tag{5}$$

We can further elaborate on how this equation is quantified in practice. Taking all intermediate factors into account, the predictive distribution of the model is $p(y \mid \delta_{1:n}, x, w)$, which can be approximated by sampling and clustering the answers. Based on the predictive distribution we can calculate the marginal distribution $p\left(y \mid \delta_{1:i-1}, x, w\right) = \mathbb{E}_{\delta_{i:n}}\left[p\left(y \mid \delta_{1:n}, x, w\right)\right]$. Then the uncertainty contributed by δ_i is quantified using these marginal distributions:

$$I[y; \delta_i \mid \delta_{1:i-1}, x, w] = \mathbb{E}_{\delta_{1:i}}\left[D_{\mathrm{KL}}\left(p(y \mid \delta_{1:i}, x, w) \parallel p(y \mid \delta_{1:i-1}, x, w)\right)\right] \tag{6}$$

Mutual information here measures the expected reduction in uncertainty about y given knowledge of δ_i and it can be expressed as the expected KL divergence between the conditional and marginal distributions. The KL divergence quantifies the decrease in uncertainty (or increase of surprisal) when updating our belief from the marginal distribution to the conditional distribution, thereby capturing how additional information about δ_i influences the model's predictions. This decomposition allows us to quantify the individual contributions of each intermediate variable to the total uncertainty. In the running example, $n = 2, \delta_1 = a$, and $\delta_2 = r$. A practical pipeline for implementing the decomposition is shown in Algorithm 1.

Algorithm 1 Practical Pipeline for General Uncertainty Decomposition

Input: Input x, model w and n pre-selected intermediate variables $\delta_i|_{i=1}^n$.
Output: Uncertainty measures $U_{\delta_1}(x), U_{\delta_2}(x), \cdots, U_{\delta_n}(x)$ and $U_t(x)$ on input x.
Sample candidates $\delta_{1:n}$ from their joint distribution (assumed known, see Sec. 3);
foreach *sampled candidates* $\delta_{1:n}$ **do**
  Query and sample responses from the model predictive distribution;
  Cluster the responses and save distribution $p(y \mid \delta_{[1:n]}, x, w)$ for each r, a;
end
Compute marginal distribution $p(y \mid \delta_{[1:i]}, x, w)$, $\forall i \in [1, n-1]$ and $p(y \mid x, w)$;
Compute $U_{\delta_i}(x) = I[y; \delta_i \mid \delta_{1:i-1}, x, w] = \mathbb{E}_{\delta_{1:i}}\left[D_{\mathrm{KL}}\left(p(y \mid \delta_{1:i}, x, w) \parallel p(y \mid \delta_{1:i-1}, x, w)\right)\right]$;
Compute total uncertainty $U_t(x) = \mathbb{H}[y \mid \delta_{1:n}, x, w] + \sum_{i=1}^{n} U_{\delta_i}(x)$;
return $U_{\delta_1}(x), U_{\delta_2}(x), \cdots, U_{\delta_n}(x)$ *and* $U_t(x)$

Examples of intermediate variables include different network modules, prompt rephrasings, contextual information, etc (see Sec.2.4). When it comes to the practical decomposition order of intermediate variables, if a variable δ_j is generated conditionally based on δ_i, it is natural to decompose δ_i before δ_j; Otherwise, we can just assume their joint distribution is known. For further discussion please see Appendix B. Regardless of the order of decomposition, the decomposition and analysis principles remain consistent, allowing us to systematically quantify the uncertainty contributed by each variable. This flexibility enables our framework to adapt to a wide range of models and applications by selecting relevant intermediate variables based on the specific setting.

2.3 UNDERSTANDING THE DECOMPOSED UNCERTAINTY

As discussed in Sec.1, calibration refers to the alignment between measured uncertainties and actual error rates in this context (Gruber & Buettner, 2022). Because a well-calibrated uncertainty metric helps in identifying when the model is likely to produce inaccurate or hallucinated outputs, calibration is critical for applications such as hallucination detection and self-training (Farquhar et al., 2024). Existing uncertainty quantification methods, including total uncertainty in our framework, may not always be well-calibrated (Vashurin et al., 2025), leading to situations where a model is overconfident despite high uncertainty.

Calibration Test. We use a simple method to help with ill-calibrated uncertainty measures. Our decomposition framework offers a way to measure the individual components in total uncertainty that are contributed by different variables. We then empirically assess the relationship between different uncertainty components and hallucinations. As we will see in Sec. 3, some uncertainty components correlate well with model hallucinations and thus provide better calibration results and help in hallucination detection and self-training. From this view, we call different uncertainty components calibrators.

For example, in the case in the running example (Fig. 1 and Sec. 3.1), we evaluate the effectiveness of each calibrator by examining how well each uncertainty component (e.g., U_a, U_r and U_o) correlates with actual prediction hallucination rates. A strong positive correlation between an uncertainty component and hallucination rate indicates that the uncertainty component is a good predictor of model hallucination. Therefore we call it an effective calibrator and can then be used in hallucination detection, self-training, and other applications where understanding model uncertainty is essential for improving reliability and trust. Conversely, uncertainty components with weak or negative correlations fail to consistently predict hallucinations and are less useful for calibration.

2.4 USE CASES AND RELATED WORK

Table 1: Uncertainty decomposition in different scenarios. For further discussion see Appendix A.3.

Setting	δ	**Formula**	**Examples**
Aleatoric-Epistemic Decomposition	model w	$\mathbb{H}[y \mid x, \mathcal{D}] = \mathbb{H}[y \mid x, w]$ $+I[y; w \mid x, \mathcal{D}]$	(Hüllermeier & Waegeman, 2021)
In-Context Examples or Input Clarification	context or clarification c	$\mathbb{H}[y \mid x, w] = \mathbb{H}[y \mid c, x, w]$ $+I[y; c \mid x, w]$	(Ling et al., 2024; Hou et al., 2024)
Prompt Rephrasing or Input Augmentation	prompts q	$\mathbb{H}[y \mid x, w] = \mathbb{H}[y \mid q, x, w]$ $+I[y; q \mid x, w]$	(Jiang et al., 2023; Yadkori et al., 2024)
VLM Attributions Binding (Ours)	attribution a, rephrasing r	$\mathbb{H}[y \mid x, w] = \mathbb{H}[y \mid r, a, x, w]$ $+I[y; r \mid a, x, w] + I[y; a \mid x, w]$	Sec. 3.1
LLM Math Reasoning (Ours)	entity name c, prompts q	$\mathbb{H}[y \mid x, w] = \mathbb{H}[y \mid q, c, x, w]$ $+I[y; q \mid c, x, w] + I[y; c \mid x, w]$	Sec. 3.2

As we mentioned earlier, this framework is capable of decomposing various types of uncertainty, without imposing any prior assumptions on the intermediate variable δ. Table 1 provides examples of relevant use cases from the literature.

Previous works like Bootstrapped DQN (Lakshminarayanan et al., 2017b) and random network distillation Burda et al. (2018), in our view, can be summarized as follows: in these learning algorithms, epistemic uncertainty serves as an effective calibrator. Thus, the effectiveness of learning is reflected in the uncertainty of w, i.e., the epistemic term, which supports the efficacy of ensemble learning (Dong et al., 2020; Osband et al., 2016; Ghasemipour et al., 2022) and pessimistic learning, eliminating aleatoric components inherent in the environment like the "noisy TV" problem (Burda et al., 2018).. In recent work on LLMs, different parts have been decoupled according to the settings, each focusing on a single intermediate variable and drawing analogies with the aleatoric-epistemic decomposition. For example, Hou et al. (2024) focused on input ambiguity, introducing an additional clarification step c. Ling et al. (2024) concentrated on sampling in-context examples, treating sampling the context examples as an intermediate variable. Additionally, Yadkori et al. (2024) implicitly

used the prompt prefix or suffix as an intermediate variable, while Jiang et al. (2023) integrated various prompting design methods.

3 Applications and Experiments

In this section, we present empirical use cases for our decomposition framework and the insights it provides. We apply the framework to two applications (hallucination detection, self-training) in two tasks: VLM Attribution Binding (Sec. 3.1) and LLM math reasoning (Sec. 3.2). In each task, our setting involves two subsets: (1) Dev Set: a small subset with ground truth answers, is used to identify effective and ineffective calibrators. (2) Test Set: the subset without ground truth answers, where we test applications (e.g. hallucination detection and self-training) with identified calibrators.

3.1 Application 1: VLM Attribution Binding Task

When querying a vision language model (VLM) about an image containing a green circle and a blue square (Fig. 1), such as asking which one is bigger, we need an *attribution* for composing the query. This attribution may refer to the object name (e.g. comparing the circle and the square) or to the object color (e.g. comparing the green one with the blue one). The accuracy often differs between these two question forms (Rahmanzadehgervi et al., 2024; Kamath et al., 2023; Zeng et al., 2024). We view this setting as an instance of the binding problem (Greff et al., 2020) or a problem on compositionality (Han et al., 2024) and refer to it as *attribution binding* (details in Appendix A.3). Because VLMs are expected to correctly identify all visible attributes of objects (e.g. names and colors) and link these attributes to the correct objects, we treat variations in attributions as different instantiations of the same question, "`which object is bigger`". In other words, the intermediate variables a (attribution) and r (rephrasing) influence the model's output without changing the ground truth that is determined by the original input x (comparing the objects in the image). We explore the binding issue by separating the uncertainty related to attribution a from that related to prompt rephrasing r.

Implementations. We need to quantify the uncertainty on a and r in a controlled setting. Since there is no good benchmark available yet, we use the following synthetic data approach. For this task, we create a dataset consisting of images each depicting simple scenes of two objects with visible different occupancy, detailed in Appendix C.1. Then the pipeline follows Algorithm 1. Along with the images, we pose all images with the same question x: "`Which object is bigger?`". Then we generate specific question instances incorporate two dimensions of variances: specific attributions a to reference the objects and the rephrasing r. Specifically, we prompt `GPT-4o` using `Langfun`[1] framework to generate both dimensions of the questions (Achiam et al., 2023; Peng, 2023). We set the number of candidates for both $\delta_1 = a$ and $\delta_2 = r$ as 6. Under each query instance, we sample 10 predictions from the tested model (`InternVL2-4B` and `Llava-1.6-7B`) (Chen et al., 2023; 2024) with temp $= 1.0$. Then we use sampled predictions to estimate the model prediction distribution $p(y \mid r, a, x, w)$, in which the y is defined on semantic space (Farquhar et al., 2024): which means, in this task, since y has only two options (two objects) while it can be expressed in various wording ways with different attributions like "`red object`", or the "`object on the right side`", we view those generations that referred to the same object as the same y. We use `GPT-4o` prompted with ground truth information to cluster the various predictions into two y options, judge their correctness, and calculate the average hallucination rate for every x: Error(x). Detailed implementation procedures and additional examples are provided in Appendix C.1.

Correlation Test. Following Algorithm 1, given the pre-selected w, we have the prediction distribution $p(y \mid a, r, x, w)$ for all a, r, x. Using these, we calculate $p(y \mid a, x, w) = \mathbb{E}_r[p(y \mid r, a, x, w)]$ and $p(y \mid x, w) = \mathbb{E}_a[p(y \mid a, x, w)]$ Then we quantify all uncertainty terms with $U_o(x) = \mathbb{E}_{a,r}\big[\mathbb{H}[p(y \mid r, a, x, w)]\big]$; $U_r(x) = \mathbb{E}_{a,r}\big[D_{\mathrm{KL}}[p(y \mid r, a, x, w) \parallel p(y \mid a, x, w)]\big]$; $U_a(x) = \mathbb{E}_a\big[D_{\mathrm{KL}}[p(y \mid a, x, w) \parallel p(y \mid x, w)]\big]$, and $U_t(x) = U_o(x) + U_r(x) + U_a(x)$. As introduced in Sec 2.3, we examine the correlation between different uncertainty terms and hallucination rates Error(x) on the dev set, results listed the statistics in Table 2. We also present the scatter plot between uncertainty components and hallucination rate on the dev set in Fig 9, which shows that for

[1] `https://github.com/google/langfun`

Table 2: Calibration test of different calibrators in the VLM attribution binding task. Using this table we can determine whether it is an effective or ineffective calibrator in our framework: here we interpret that U_a is an effective calibrator while the other three: U_r, U_o are negatively correlated and U_t is at chance level (or random).

Calibrator	InternVL-2-4B		Llava-1.6-7B		Description
	Corre. Coeff.	p-value	Corre. Coeff.	p-value	
U_o	0.0125	0.9016	0.0052	0.8341	random
U_r	-0.3729	0.0001	-0.3104	0.0023	negative correlated
U_a	0.5701	0.0011	0.5903	0.0009	positive correlated
U_t	0.0704	0.4867	0.0453	0.5120	random

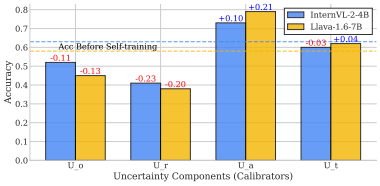
an ineffective calibrator like total uncertainty, there are samples in the low uncertainty region that exhibits both low hallucination rates, showing the model's overconfident behavior.

Hallucination Detection. We start with evaluating the performance of different calibrators in hallucination detection, as in previous work (Farquhar et al., 2024; Hou et al., 2024; Ling et al., 2024). We calculate the AUROC and AURAC metrics between different uncertainty components and hallucination rates (Table 2a) to validate the hypothesis that effective calibrators work well in uncertainty-based hallucination detection while others work poorly.

Self-Training. Then we inspect to possibility of self-training with the help of an effective calibrator. This is quite similar to rejection sampling in LLM settings, however here we use the uncertainty metrics to select those maintained for fine-tuning. In this task, we use certain predictions of the model's own as pseudo-labels to finetune itself, just as rejection sampling which trains the model itself with its highest-scored generations. Specifically, we used the samples that fall into the lower 30% partition of each uncertainty term as training labels and fine-tuned the model, results in Fig. 2b.

	InternVL-2-4B		Llava-1.6-7B	
	AUROC	AURAC	AUROC	AURAC
U_o	0.486	0.614	0.476	0.553
U_r	0.253	0.476	0.364	0.495
U_a	0.788	0.837	0.803	0.858
U_t	0.587	0.692	0.609	0.689

(a) AUROC and AURAC values in hallucination detection for different uncertainty components (calibrators).



(b) Accuracy under different Self-Training guided by different uncertainty components (calibrators).

Analysis. The intuition behind why a turns out to be an effective calibrator is that given a VLM model's performance varies across different attributes a compared with changing prompt wording by r; for example, it may achieve a relatively high accuracy when questioned about color, but perform poorly in understanding spatial relationships. These results suggest that the ability to bind attributes to objects is crucial for reliable VLM performance and that improvements in attribution binding will have a more substantial impact than merely rephrasing prompts. It also hints that by properly using the binding structure, we can gain semi-supervised information gain for free, just like the self-training experiment.

3.2 Application 2: LLM Math Reasoning Task

In this section, we apply uncertainty decomposition to a math reasoning task using LLMs. The setting is that an LLM is queried with math problems, while the questions presented in the form of application problems often contain multiple entity names (such as human names), even though the underlying question remains unchanged. Additionally, the style of prompting, particularly which elicits chain-of-thought (CoT) (Wei et al., 2023), can also influence the output. We will examine these two variables: $\delta_1 = c$ represents entity names, $\delta_2 = q$ represents prompts design.

Implementation. We choose the `SVAMP` benchmark (Patel et al., 2021) and model `Gemma-2-9B-it` (Team et al., 2024) for the task. The implementation is quite similar to that of the previous task. We utilize `GPT-4o` with `Langfun` framework to rephrase the entity names in a single math question, resulting in the same query in 6 different forms alternating the entity names, see Table 6. For example, we rephrase a question where "`Paige raised 7 goldfish and 12 catfish in the pond, but stray cats loved eating them. Now she has 15 left`" with alternative entity names, such as "`Tom raised 7 rabbits and 12 hamsters in the yard, but wild foxes loved chasing them. Now he has 15 left`", resulting in equivalent queries like "`How many fishes disappeared?`" and "`How many pets vanished?`" respectively. We then query them with 6 different designed CoT prompts templates respectively. For more details, see the Appendix C.2. Applying our framework, we can naturally derive the decomposition: $\mathbb{H}[y \mid x, w] = \mathbb{H}[y \mid x, c, q, w] + I[y; q \mid x, c, w] + I[y; c \mid x, w]$ and quantify U_o, U_q (CoT prompt design), U_c (entity-name-related), U_t following Algorithm 1.

Results and Analysis. The experimental procedure is the same as in the previous Sec. 3.1. Here, we report the results of the calibration test and hallucination detection. Results for the calibration test are presented in Table 3, indicating that U_q is an effective calibrator with a significant positive correlation, while U_o, U_c, and U_t show either weak or negative correlations with error rates. For hallucination detection, we report the AUROC and AURAC metrics for different calibrators, as shown in Table 2a. The result shows U_q outperforms other uncertainty components in detecting hallucinations, suggesting that prompt design variability is more informative than entity names in questions for signaling the possibility of making errors.

Table 3: Quantitative comparison of different calibrators in the uncertainty calibration task. We can interpret U_q as an effective calibrator, while U_c, U_o, and U_t are not.

Calibrator	Corre. Coeff.	p-value	Description
U_o	-0.525	2.109e-08	negative correlated
U_q	0.460	1.495e-06	positive correlated
U_c	0.089	0.3767	random
U_t	-0.024	0.8103	random

Table 4: AUROC and AURAC values in hallucination detection task for different uncertainty parts.

U_x	AUROC	AURAC
U_o	0.291	0.316
U_q	0.819	0.742
U_c	0.530	0.540
U_t	0.592	0.544

The results show that prompt design q serves as an effective calibrator for LLM math reasoning, as its structure significantly influences reasoning accuracy. It reveals that, for reasoning tasks, refining prompt structure is more impactful for reducing uncertainty than altering surface things like entity names in questions. The results here are actually supported by other works (Jiang et al., 2023), where techniques like prompting have been shown to help calibrate the model.

4 Conclusion

In conclusion, we introduce a unified uncertainty decomposition framework that extends traditional concepts of uncertainty decomposition to encompass multiple intermediate variables, which is more suitable and useful in the foundation model era. By systematically quantifying different uncertainty components, we can diagnose the sources of model uncertainty and their impact on performance. Our framework reveals that not all sources of uncertainty are equally informative; specifically, it distinguishes between effective calibrators, which correlate positively with error rates, and ineffective calibrators. This nuanced understanding better fits the practice reality where models exhibit high overconfidence and provide a pathway to better uncertainty estimation. We demonstrate the practical use of our framework through two settings—the VLM attribution binding task and the LLM math reasoning task, and applications of hallucination detection and self-training. Our framework is highly extensible and grounded in information theory. It opens new avenues for future research into uncertainty for foundation models, paving the way for trustworthy and interpretable AI systems.

REFERENCES

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

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pp. 610–623, 2021.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation, 2018. URL https://arxiv.org/abs/1810.12894.

Bertrand Charpentier, Ransalu Senanayake, Mykel Kochenderfer, and Stephan Günnemann. Disentangling epistemic and aleatoric uncertainty in reinforcement learning. *arXiv preprint arXiv:2206.01558*, 2022.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*, 2023.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*, 2024.

Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter? *Structural safety*, 31(2):105–112, 2009.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. A survey on in-context learning, 2022.

Xibin Dong, Zhiwen Yu, Wenming Cao, Yifan Shi, and Qianli Ma. A survey on ensemble learning. *Frontiers of Computer Science*, 14:241–258, 2020.

Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jiang, Bill Yuchen Lin, Sean Welleck, Peter West, Chandra Bhagavatula, Ronan Le Bras, et al. Faith and fate: Limits of transformers on compositionality. *Advances in Neural Information Processing Systems*, 36, 2024.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, June 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07421-0. URL https://doi.org/10.1038/s41586-024-07421-0.

Mudasir A Ganaie, Minghui Hu, Ashwani Kumar Malik, Muhammad Tanveer, and Ponnuthurai N Suganthan. Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence*, 115:105151, 2022.

Seyed Kamyar Seyed Ghasemipour, Shixiang Shane Gu, and Ofir Nachum. Why so pessimistic? estimating uncertainties for offline rl through ensembles, and why their independence matters, 2022. URL https://arxiv.org/abs/2205.13703.

Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. On the binding problem in artificial neural networks, 2020. URL https://arxiv.org/abs/2012.05208.

Tobias Groot and Matias Valdenegro-Toro. Overconfidence is key: Verbalized uncertainty evaluation in large language and vision-language models. *arXiv preprint arXiv:2405.02917*, 2024.

Sebastian Gruber and Florian Buettner. Better uncertainty calibration via proper scores for classification and beyond. *Advances in Neural Information Processing Systems*, 35:8618–8632, 2022.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. *arXiv preprint arXiv:1706.04599*, 2017.

Xu Han, Linghao Jin, Xiaofeng Liu, and Paul Pu Liang. Progressive compositionality in text-to-image generative models. *arXiv preprint arXiv:2410.16719*, 2024.

Stephen C Hora. Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management. *Reliability Engineering & System Safety*, 54(2-3):217–223, 1996.

Bairu Hou, Yujian Liu, Kaizhi Qian, Jacob Andreas, Shiyu Chang, and Yang Zhang. Decomposing uncertainty for large language models through input clarification ensembling, 2024. URL https://arxiv.org/abs/2311.08718.

Jiaxin Huang, Shixiang Shane Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. Large language models can self-improve. *arXiv preprint arXiv:2210.11610*, 2022.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions, 2023. URL https://arxiv.org/abs/2311.05232.

Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Machine Learning*, 110(3):457–506, March 2021. ISSN 1573-0565. doi: 10.1007/s10994-021-05946-3. URL http://dx.doi.org/10.1007/s10994-021-05946-3.

Alan Jeffares, Tennison Liu, Jonathan Crabbé, and Mihaela van der Schaar. Joint training of deep ensembles fails due to learner collusion, 2023. URL https://arxiv.org/abs/2301.11323.

Mingjian Jiang, Yangjun Ruan, Sicong Huang, Saifei Liao, Silviu Pitis, Roger Baker Grosse, and Jimmy Ba. Calibrating language models via augmented prompt ensembles. In *International conference on machine learning*. PMLR, 2023.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know, 2022. URL https://arxiv.org/abs/2207.05221.

Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's "up" with vision-language models? investigating their struggle with spatial reasoning, 2023. URL https://arxiv.org/abs/2310.19785.

Amita Kamath, Cheng-Yu Hsieh, Kai-Wei Chang, and Ranjay Krishna. The hard positive truth about vision-language compositionality, 2024. URL https://arxiv.org/abs/2409.17958.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation, 2023. URL https://arxiv.org/abs/2302.09664.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017a.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles, 2017b. URL https://arxiv.org/abs/1612.01474.

Paul Pu Liang, Yun Cheng, Xiang Fan, Chun Kai Ling, Suzanne Nie, Richard Chen, Zihao Deng, Nicholas Allen, Randy Auerbach, Faisal Mahmood, Ruslan Salakhutdinov, and Louis-Philippe Morency. Quantifying and modeling multimodal interactions: An information decomposition framework, 2023. URL https://arxiv.org/abs/2302.12247.

Chen Ling, Xujiang Zhao, Xuchao Zhang, Wei Cheng, Yanchi Liu, Yiyou Sun, Mika Oishi, Takao Osaki, Katsushi Matsuda, Jie Ji, Guangji Bai, Liang Zhao, and Haifeng Chen. Uncertainty quantification for in-context learning of large language models, 2024. URL https://arxiv.org/abs/2402.10189.

Jeremiah Liu, John Paisley, Marianthi-Anna Kioumourtzoglou, and Brent Coull. Accurate uncertainty estimation and decomposition in ensemble learning. *Advances in neural information processing systems*, 32, 2019a.

Ling Liu, Wenqi Wei, Ka-Ho Chow, Margaret Loper, Emre Gursoy, Stacey Truex, and Yanzhao Wu. Deep neural network ensembles against deception: Ensemble diversity, accuracy and robustness. In *2019 IEEE 16th international conference on mobile ad hoc and sensor systems (MASS)*, pp. 274–282. IEEE, 2019b.

W. McGill. Multivariate information transmission. *Transactions of the IRE Professional Group on Information Theory*, 4(4):93–111, 1954. doi: 10.1109/TIT.1954.1057469.

Subhabrata Mukherjee and Ahmed Awadallah. Uncertainty-aware self-training for few-shot text classification. *Advances in Neural Information Processing Systems*, 33:21199–21212, 2020.

Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn, 2016. URL https://arxiv.org/abs/1602.04621.

Ian Osband, Zheng Wen, Seyed Mohammad Asghari, Vikranth Dwaracherla, Morteza Ibrahimi, Xiuyuan Lu, and Benjamin Van Roy. Epistemic neural networks. *Advances in Neural Information Processing Systems*, 36:2795–2823, 2023.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are NLP models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2080–2094, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.168. URL https://aclanthology.org/2021.naacl-main.168.

Daiyi Peng. Langfun. https://github.com/google/langfun (Version 0.0.1), September 2023.

Pooyan Rahmanzadehgervi, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. Vision language models are blind, 2024. URL https://arxiv.org/abs/2407.06581.

Lior Rokach. Ensemble-based classifiers. *Artificial intelligence review*, 33:1–39, 2010.

Kajetan Schweighofer, Lukas Aichberger, Mykyta Ielanskyi, Günter Klambauer, and Sepp Hochreiter. Quantification of uncertainty with adversarial models, 2023. URL https://arxiv.org/abs/2307.03217.

Yifan Song, Da Yin, Xiang Yue, Jie Huang, Sujian Li, and Bill Yuchen Lin. Trial and error: Exploration-based trajectory optimization for llm agents, 2024. URL https://arxiv.org/abs/2403.02502.

Stone Tao, Fanbo Xiang, Arth Shukla, Yuzhe Qin, Xander Hinrichsen, Xiaodi Yuan, Chen Bao, Xinsong Lin, Yulin Liu, Tse kai Chan, Yuan Gao, Xuanlin Li, Tongzhou Mu, Nan Xiao, Arnav Gurha, Zhiao Huang, Roberto Calandra, Rui Chen, Shan Luo, and Hao Su. Maniskill3: Gpu parallelized robotics simulation and rendering for generalizable embodied ai. *arXiv preprint arXiv:2410.00425*, 2024.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Maria Mihaela Trusca, Wolf Nuyts, Jonathan Thomm, Robert Honig, Thomas Hofmann, Tinne Tuytelaars, and Marie-Francine Moens. Object-attribute binding in text-to-image generation: Evaluation and control, 2024. URL https://arxiv.org/abs/2404.13766.

Roman Vashurin, Ekaterina Fadeeva, Artem Vazhentsev, Lyudmila Rvanova, Akim Tsvigun, Daniil Vasilev, Rui Xing, Abdelrahman Boda Sadallah, Kirill Grishchenkov, Sergey Petrakov, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, Maxim Panov, and Artem Shelmanov. Benchmarking uncertainty quantification methods for large language models with lm-polygraph, 2025. URL https://arxiv.org/abs/2406.15627.

Cheng Wang. Calibration in deep learning: A survey of the state-of-the-art. *arXiv preprint arXiv:2308.01222*, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023. URL https://arxiv.org/abs/2201.11903.

Junda Wu, Zhehao Zhang, Yu Xia, Xintong Li, Zhaoyang Xia, Aaron Chang, Tong Yu, Sungchul Kim, Ryan A. Rossi, Ruiyi Zhang, Subrata Mitra, Dimitris N. Metaxas, Lina Yao, Jingbo Shang, and Julian McAuley. Visual prompting in multimodal large language models: A survey, 2024. URL https://arxiv.org/abs/2409.15310.

Yunlong Xiong, Jinhyuk Lee, Chandan Joshi, Jesse Finnie-Ansley, and Dragomir Radev. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*, 2023.

Yasin Abbasi Yadkori, Ilja Kuzborskij, András György, and Csaba Szepesvári. To believe or not to believe your llm, 2024. URL https://arxiv.org/abs/2406.02543.

Haoyan Yang, Yixuan Wang, Xingyin Xu, Hanyuan Zhang, and Yirong Bian. Can we trust llms? mitigate overconfidence bias in llms through knowledge transfer. *arXiv preprint arXiv:2405.16856*, 2024.

Yue Yu, Lingkai Kong, Jieyu Zhang, Rongzhi Zhang, and Chao Zhang. Actune: Uncertainty-based active self-training for active fine-tuning of pretrained language models. In *Proceedings of the 2022 conference of the North American chapter of the association for computational linguistics: human language technologies*, pp. 1422–1436, 2022.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it?, 2023. URL https://arxiv.org/abs/2210.01936.

Y. Zeng, Y. Huang, J. Zhang, Z. Jie, Z. Chai, and L. Wang. Investigating compositional challenges in vision-language models for visual grounding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14141–14151, Los Alamitos, CA, USA, jun 2024. IEEE Computer Society. doi: 10.1109/CVPR52733.2024.01341. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52733.2024.01341.

Jize Zhang, Bhavya Kailkhura, and T. Yong-Jin Han. Mix-n-match: Ensemble and compositional methods for uncertainty calibration in deep learning, 2020. URL https://arxiv.org/abs/2003.07329.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. Siren's song in the ai ocean: A survey on hallucination in large language models, 2023. URL https://arxiv.org/abs/2309.01219.

Ruochen Zhao, Xingxuan Li, Shafiq Joty, Chengwei Qin, and Lidong Bing. Verify-and-edit: A knowledge-enhanced chain-of-thought framework, 2023. URL https://arxiv.org/abs/2305.03268.

A Background and related work

A.1 Preliminary of Information Theory Used

Some basic rules:

$$\mathrm{CE}[P, Q] = \mathbb{H}[P] + D_{\mathrm{KL}}(P \parallel Q) \tag{7}$$

$$I(X; Y) = \mathbb{E}_Y \left[D_{\mathrm{KL}} \left(p_{X|Y} \parallel p_X \right) \right] \tag{8}$$

The uncertainty contributed to a specific variable δ is quantified by mutual information and by the equation $I(X;Y) = \mathbb{E}_Y \left[D_{\mathrm{KL}} \left(p_{X|Y} \parallel p_X \right) \right]$ are related to KL, which matches with the natural on KL-divergence. The KL calculates the averaged difference of surprisal, thus the decrease of uncertainty, note that $D_{\mathrm{KL}}(P \parallel Q) = \mathbb{E}_p[(-\log q) - (-\log p)]$ in which $-\log p$ is the surprisal.

A.2 Traditional Uncertainty Decomposition

In classical formulations (Hüllermeier & Waegeman, 2021; Schweighofer et al., 2023), the Bayesian framework offers a principled way to treat the uncertainty about the model weights through the posterior over hypothesis space $p(w \mid \mathcal{D}) \propto p(\mathcal{D} \mid w)p(w)$ for a given dataset $\mathcal{D}$. The Bayesian model average (BMA) predictive distribution is given by

$$p(y \mid x, \mathcal{D}) = \int_{\mathcal{W}} p(y \mid x, w) p(w \mid \mathcal{D}) \, dw \tag{9}$$

.

And the uncertainty of the BMA predictive distribution is commonly measured by the entropy $\mathbb{H}[p(y \mid x, \mathcal{D})]$. It refers to the total uncertainty, which can be decomposed into an aleatoric and an epistemic part. The BMA predictive entropy is equal to the posterior expectation of the cross-entropy between the predictive distribution of candidate models and the BMA, using Eq. 7.

Expected uncertainty when selecting a model w:

$$\mathrm{CE}[p(y \mid x, w), p(y \mid x, \mathcal{D})] = \mathbb{H}[p(y \mid x, w)] + D_{\mathrm{KL}}(p(y \mid x, w) \parallel p(y \mid x, \mathcal{D})) \tag{10}$$

Taking an expectation of w on Eq. 10 results in uncertainty formulation:

$$\begin{aligned} \underbrace{\mathbb{H}[p(y \mid x, \mathcal{D})]}_{\text{total uncertainty}} &= \mathbb{E}_{p(w|\mathcal{D})} \left[\mathrm{CE}[p(y \mid x, w), p(y \mid x, \mathcal{D})] \right] \\ &= \underbrace{\mathbb{E}_{p(w|\mathcal{D})} \left[\mathbb{H}(p(y \mid x, w)) \right]}_{\text{aleatoric uncertainty}} + \underbrace{\mathbb{E}_{p(w|\mathcal{D})} \left[D_{\mathrm{KL}}(p(y \mid x, w) \parallel p(y \mid x, \mathcal{D})) \right]}_{\text{epistemic uncertainty}} \end{aligned} \tag{11}$$

Or, writing in the form of mutual information and conditional entropy as:

$$\mathbb{H}[y \mid x, \mathcal{D}] = \mathbb{H}[y \mid x, w] + I[y; w \mid x, \mathcal{D}] \tag{12}$$

A.3 Application Background: VLM Binding (Compositionality) Problem

The running example in our paper, which we refer to as VLM attribution binding problem, has profound background. We provide a brief review here. We abstract this compositionality issue as the problem of binding objects with their multiple attributes.

Recent research has revealed that large-scale pretrained VLMs struggle with understanding compositionality in images (Zeng et al., 2024; Kamath et al., 2024). They exhibit limitations in integrating objects with their attributes and understanding spatial relationships (Rahmanzadehgervi et al., 2024; Kamath et al., 2023). We abstract this compositionality issue as the binding problem of objects and their multiple dimensions of attributions. When a model has binding issues between attributes and objects, it can lead to severe hallucinations, such as failing to distinguish between "`the grass is eating the horse`" and "`the horse is eating the grass`" (binding of the object and predicate), which appears absurd to humans. Even state-of-the-art VLMs can easily make

errors in determining which object is on the left and which is on the right (binding of position). Although the model can correctly identify two objects in an image, it often confuses them when referring to attributes like color or shape. Some works have analyzed this issue from the perspectives of flaws in pretraining data and model priors (Yuksekgonul et al., 2023; Trusca et al., 2024), but a systematic quantitative explanation is lacking.

A.3.1 Uncertainty Quantification in LLMs

Uncertainty quantification in large language models focuses on measuring uncertainty within the semantic space of model outputs. Logit-based estimation calculates token-level probabilities or entropy (Guo et al., 2017). Confidence elicitation via verbalization asks models to provide numerical uncertainty scores, with chain-of-thought prompting shown to improve these estimates (Xiong et al., 2023). Consistency-based assessment detects conflicting responses as indicators of uncertainty or hallucination (Zhao et al., 2023). These approaches can offer insights into the internal reasoning and reliability of LLMs Zhang et al. (2023). Another direction focus on the semantic invariance of natural language and build methods based on the concept of semantic uncertainty, which does some clustering on semantic level and then adapt entropy-based formulation (Farquhar et al., 2024).

B Discussion On the Framework

In this section we discussion about details when applying our framework, using the setting in the running example presented in Sec. 2.2.

B.1 Why some calibrator (uncertainty components) ill-calibrated?

Based on our unified decomposition framework, especially the ensembling perspective outlined in Eq. 6 that each uncertainty part is quantified as KL divergence between individual prediction distributions and an aggregated average distribution, we can discuss why some uncertainty components are badly calibrated-negatively correlated with error rates. Ensemble-based methods are traditionally valued for enhancing model performance by reducing uncorrelated errors and increasing robustness through the diversity of ensemble members, thereby better capturing the $x \to y$ mapping (Rokach, 2010; Ganaie et al., 2022; Lakshminarayanan et al., 2017a). However, this benefit does not consistently extend to inference-time ensembling techniques, such as prompt augmentation or output bootstrapping (Jiang et al., 2023). In these cases, ensembling may fail to produce a more accurate mapping, meaning that higher uncertainty does not necessarily indicate a higher likelihood of error. This discrepancy arises primarily for two reasons. First, when the true distribution is approximated by model parameters rather than the data $\mathcal{D}$, ensembling can reinforce the model's inherent biases, conflicting with the actual data distribution and nullifying beneficial explorations from sampling stochasticity (Song et al., 2024). Second, joint training can lead to ensembling collapse, where ensemble members become overly similar, reducing their diversity and effectiveness (Jeffares et al., 2023; Liu et al., 2019b). This lack of diversity can result in spurious structures and ineffective ensembling during inference. Our unified decomposition framework quantifies uncertainty in these scenarios and provides diagnostic methods to identify sources of model uncertainty, thereby highlighting situations where ensembling might not enhance—and could even degrade—model performance.

B.2 Intermediate Variables Designs

When first examining our uncertainty decomposition framework (e.g., Fig. 1), it may seem that intermediate variables, such as rephrasings (r) and attributions referred to (a), are arbitrarily added and inflate the total uncertainty. However, these variables are intrinsic to model operation and crucial for capturing prediction variability and uncertainty.

Ideally, the model's predictions should remain consistent regardless of variations in q and c. However, models often show sensitivity to these variations, causing output fluctuations. Including these intermediate variables in our decomposition explicitly captures the uncertainty resulting from this sensitivity.

These variables are not artificial constructs but integral to the model interaction. Every model query involves specific prompt rephrings r and context attributions a, naturally treated as random variables from underlying distributions rather than fixed inputs. This treatment reflects real-world usage and allows us to model the variability introduced by differing prompt formulations and contexts.

Our approach aligns with prompt optimization and ensembling techniques. Traditional prompt optimization seeks the best prompt instance for model performance, while we generalize by treating r as a distribution, quantifying uncertainty across prompt variations. Similarly, ensembling aggregates predictions across configurations to enhance robustness; our framework systematically accounts for uncertainty at this level by considering distributions over intermediate variables.

Explicitly modeling intermediate variables within uncertainty decomposition deepens our understanding of factors affecting predictions. It reveals model sensitivity to prompt or context shifts, essential for improving reliability. This method also highlights areas needing further training or refinement to enhance robustness.

B.3 Discussion of Decomposition Practice

What about multivariate mutual information? The extension of mutual information between 3 or more variables is an open question (Liang et al., 2023; McGill, 1954), so we not bother on introducing more complex decomposition which involves multivariate mutual information as in such scenarios the physical meaning of the decomposed terms remains unclear. However, we acknowledge this direction as worthy exploration.

Decomposition Order. In applying the chain rule of conditional entropy for our uncertainty decomposition such as in Eq. 4, a natural question arises regarding the order in which we perform the decomposition: should we first condition on the context c or the prompt rephrasing q? The answer to this question is critical because the decomposition order affects the interpretation of the uncertainty components and must reflect the underlying conditional dependencies among the variables.

The appropriate decomposition order is determined by the conditional independence relationships among the variables. Conditional independence dictates how variables influence each other and, consequently, how uncertainty propagates through the model. When variables are conditionally independent given certain conditions, the order of decomposition should respect these relationships to ensure that each term accurately represents its contribution to the total uncertainty.

C Implementation Details and Examples

We present the details of experiments here.

C.1 VLM Attribution Binding Task

The Dataset Synthesis. In this dataset generation process, we use the `ManiSkill` simulator[2] (Tao et al., 2024), which is primarily designed for robot arm manipulation tasks. To adapt it to our needs, we remove the robot arm and transform it into a tabletop manipulation environment. This setup enables us to generate synthetic images that meet our controlled requirements for quantifying uncertainty in the variables a (attribution) and r (rephrasing) with minimal bias.

The dataset consists of simple scenes with two objects that differ in size and other attributes. To introduce controlled variation, we adjust several variables in each scene, including color, shape, camera position, background, and object size. Each variable has a set of options, such as nine color choices (rainbow colors plus black and white). Using nested for-loops, we exhaustively combine all options across these variables, resulting in over 1,000 unique data points.

Prompts We present the prompts used to generate intermediate variables a and r, along with the prompts used to evaluate the answers. The format of prompts is followed the `Langfun` documents.

[2] `https://github.com/haosulab/ManiSkill`

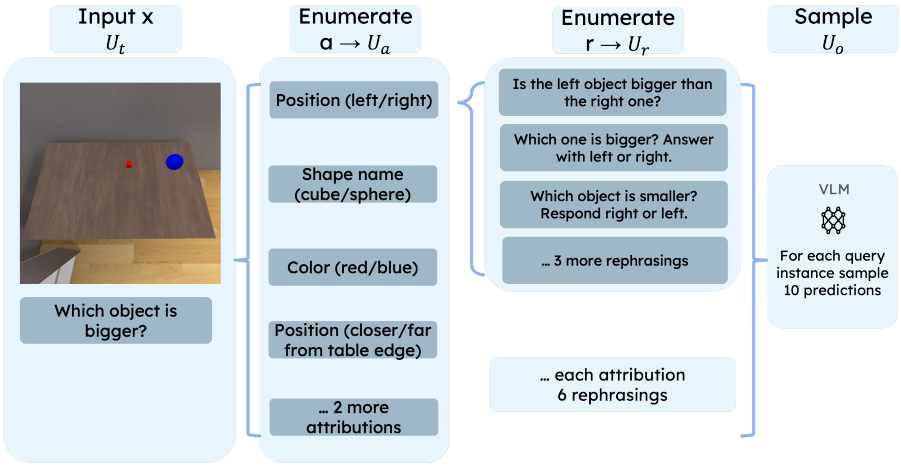


Figure 3: Decomstration of example implementation of variations on a, r in the VLM attribution binding task. We illustrate the pipeline that for a single query x, enumerates different a and r then sample predictions which then clustered into a distribution over y.

The question is asking about a fact in the image.
Please identify the fact it asks, and rephrase the question while keeping the fact the same but refer to different attribution in the image.
First reason about what are the attributions of the nouns in the question, and then rephrase the question.
For example, you can replace the nouns in the question using their color, shape, position, state, or alternate names, thus creating different questions.

Image: {{image}}

Question: {{question}}

Fact:

Rephrased questions:
1.
2.
3.
4.
5.

Figure 4: VLM attribution prompt. It is used to generate different candidates of attribution a for any specific question sample x. The "fact" is just to improve Langfun's performance.

Finetuning For the self-training experiment, we use the official repo for `InternVL`[3] (Chen et al., 2024) for fine-tuning `InternVL-2-4B`. The hyperparameters are listed in Table 5.

C.2 LLM Reasoning Task

We also detail the prompts utilized for the LLM reasoning tasks, as illustrated in Fig. 7 and 8. Additionally, Table 6 provides example bodies and their corresponding questions, offering a clear overview of the task setup and the types of problems addressed in our experiments.

[3] `https://github.com/OpenGVLab/InternVL`

Given the following question, please rephrase it in 5 different ways:
The rephrase should be asking the same thing as the question.
When asking about the same thing, you can rephrase the question by changing the nouns, adjectives, or verbs.
And you can also change the asking type from choice-of-one to yes-or-no.

Question: {{question}}

Rephrased questions:
1.
2.
3.
4.
5.

Figure 5: VLM rephrasing prompt. It is used to generate different candidates of rephrasings r given attributions a.

Your task is to determine if the model response is correct given the question and groundtruth response.
Ensure to interpret the model response in accordance to the the question.

If the question asks about the comparison of occupancy of two objects, if the groundtruth is A is bigger than B. Then the desired answer is either: "A is bigger than B" or "B is smaller than A". Both answers are correct.

If the question asks about a detail of an element that is not present in the image, A prediction of "yes", "no" or "nothing" should be considered incorrect because it inaccurately suggests that the element is presented in the image.
The correct prediction in such cases should acknowledge the absence of the element in question by stating the element is not present.
If prediction says that it can not assist or cannot provide an answer, then the prediction is incorrect.
If the question is about counting, then the prediction is correct only it matches the groundtruth counts exactly.

question={{question}},
model_response={{model_response}}
groundtruth_response={{groundtruth_response}},

Figure 6: VLM Attribution labeling prompt

C.3 More Analysis on the Calibration Test

In this subsection, we provide a comprehensive analysis of the calibration tests conducted across two distinct tasks, as illustrated in Fig. 9 and 10. These scatter plots demonstrate that when total uncertainty is employed, the model exhibits a pronounced overconfidence phenomenon, with numerous samples simultaneously showing high error rates and low uncertainty. Specifically, Fig 9 pertains to the VLM attribution binding task, while Fig 10 relates to the LLM math reasoning task. The consistent observation of high errors paired with low uncertainty across both tasks underscores the significant limitations of using total uncertainty alone. This finding highlights the critical importance and necessity of uncertainty decomposition, which enables a more nuanced and accurate calibration of model confidence, thereby enhancing the reliability and robustness of predictive performance.

Table 5: Key Training Hyperparameters

Hyperparameter	**Value**
Total Batch Size	32
Number of Epochs	2
Learning Rate	6×10^{-6}
Weight Decay	0.05
Warmup Ratio	0.03
Learning Rate Scheduler	Cosine
Model Name or Path	`OpenGVLab/InternVL2-4B`
Image Size	448
Max Sequence Length	4096
Mixed Precision (bf16)	Yes
Gradient Checkpointing	Yes
Zero Optimization Stage	zero_stage1
Freeze (Vision) Backbone	True
Vision Select Layer	-1

Given the following math problem consisting of a Body and a Question, please rephrase it by replacing entity names which are irrelevant to the underlying mathematical reasoning. Keep all numbers and the core mathematical structure intact.

Original Body: {{body}}
Original Question: {{question}}

Please provide a rephrased version with different entity names but the same mathematical structure:

Rephrased Body:
Rephrased Question:

Figure 7: LLM Reasoning entity replacing prompt; The "body" and "question" are those corresponding keys in the SVAMP dataset.

Your task is to determine if the model response is correct given the question and groundtruth response. Ensure to interpret the model response in accordance to the the question.

If the question asks about the comparison of occupancy of two objects, if the groundtruth is A is bigger than B. Then the desired answer is either: "A is bigger than B" or "B is smaller than A". Both answers are correct.

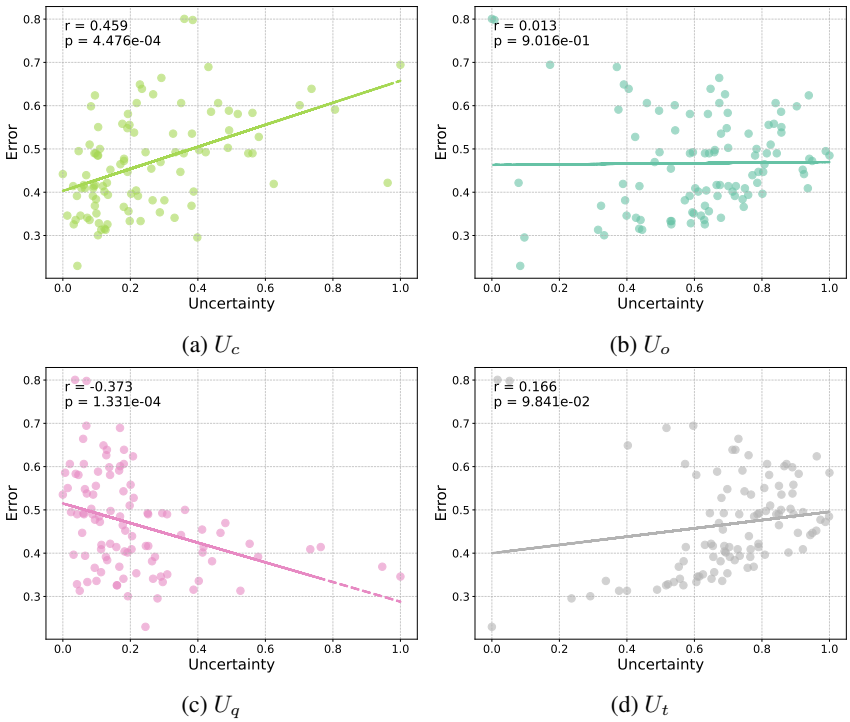
If the question asks about a detail of an element that is not present in the image, A prediction of "yes", "no" or "nothing" should be considered incorrect because it inaccurately suggests that the element is presented in the image.
The correct prediction in such cases should acknowledge the absence of the element in question by stating the element is not present.
If prediction says that it can not assist or cannot provide an answer, then the prediction is incorrect.
If the question is about counting, then the prediction is correct only it matches the groundtruth counts exactly.

question={{question}},
model_response={{model_response}}
groundtruth_response={{groundtruth_response}},

Figure 8: LLM Reasoning labeling prompt

Bodies	**Questions**
Paige raised 7 goldfish and 12 catfish in the pond but stray cats loved eating them. Now she has 15 left.	How many fishes disappeared?
Tom raised 7 rabbits and 12 hamsters in the yard but wild foxes loved chasing them. Now he has 15 left.	How many pets vanished?
Lisa raised 7 puppies and 12 kittens in the shelter but stray dogs loved bothering them. Now she has 15 left.	How many animals went missing?
Mark raised 7 ducks and 12 geese in the pond but hungry raccoons loved stealing them. Now he has 15 left.	How many birds were lost?
Emily raised 7 turtles and 12 frogs in the aquarium but curious cats loved disturbing them. Now she has 15 left.	How many creatures disappeared?
Jake raised 7 turtles and 12 slugs in the garden but wandering snails loved tasting them. Now he has 15 left.	How many critters went away?

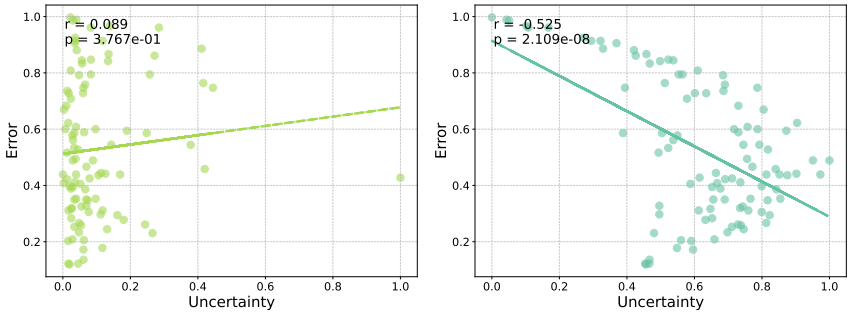
Table 6: Example Bodies and Corresponding Questions



(a) U_c (b) U_o

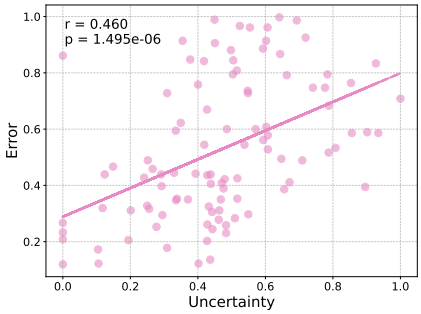
(c) U_q (d) U_t

Figure 9: Scatter plots between error rate and uncertainty parts U_c, U_o, U_q, and U_t in the VLM attribution binding task. We calculate the Pearson r and p-value to show the correlation relationships. U_c is a effective calibrator while others behave poorly.

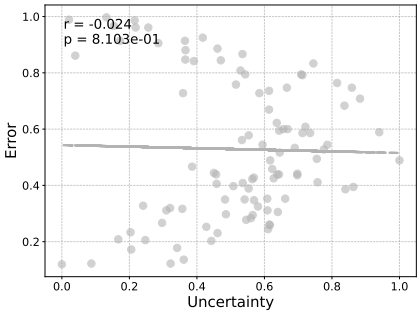


(a) U_c

(b) U_o



(c) U_q



(d) U_t

Figure 10: Scatter plots between error rate and uncertainty terms U_c, U_o, U_q, and U_t in the LLM math reasoning task. We calculate the Pearson r and p-value to show the correlation relationships. U_q is a effective calibrator while others behave poorly.