

GRADE: Probing Knowledge Gaps in LLMs through Gradient Subspace Dynamics

Yujing Wang^{1,3*} Yuanbang Liang² Yu-Kun Lai² Hainan Zhang¹ Hanqi Yan³

¹Beihang University ²Cardiff University ³King’s College London
eugeniawyj@outlook.com {LiangY32, LaiY4}@cardiff.ac.uk hanqi.yan@kcl.ac.uk

Abstract

Detecting whether a model’s internal knowledge is sufficient to correctly answer a given question is a fundamental challenge in deploying responsible LLMs. In addition to verbalising the confidence by LLM self-report, more recent methods explore the model internals, such as the hidden states of the response tokens, to capture how much knowledge is activated. We argue that such activated knowledge may not align with what the query requires, e.g., capturing the stylistic and length-related features that are uninformative for answering the query. To fill the gap, we propose **GRADE** (**GR**Adient **D**ynamics for knowl**E**dge gap detection), which quantifies the knowledge gap via the cross-layer rank ratio of the gradient to that of the corresponding hidden state subspace. This is motivated by the property of gradients as estimators of the required knowledge updates for a given target. We validate **GRADE** on six benchmarks, demonstrating its effectiveness and robustness to input perturbations. In addition, we present a case study demonstrating how the gradient chain can generate interpretable explanations of knowledge gaps in long-form answers. Code is available at <https://github.com/yjEugenia/llm-aware>.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities across diverse reasoning tasks by eliciting internal knowledge in response to various queries (Wei et al., 2022; Ahn et al., 2024). However, before taking the generated response for granted, we need to check whether the model is capable of solving the problem (Kadavath et al., 2022; Li et al., 2025a), which is different from uncertainty quantification, as a model can be highly confident yet still be wrong. Detecting such knowledge gaps is essential to mitigate failure modes such as hallucination (Ji et al., 2023; Huang et al., 2025). Here, a knowledge gap refers to the disparity between a model knowing and not knowing how to answer a query, which empirically manifests as the intrinsic difference between generating correct and incorrect answers. While LLM verbalization (Lin et al., 2022; Tian et al., 2023; Ren et al., 2025) offers the most direct route to eliciting self-reported knowledge gaps, like “Yes, I’m able to”, recent work has shifted toward exploiting intrinsic model signals. Among these, hidden states of generated answer tokens (Ni et al., 2025; Chen et al., 2026) have emerged as a particularly informative source for measuring a model’s internal knowledge.

Inspired by recent advances in mechanistic interpretability (Burns et al., 2023; Azaria & Mitchell, 2023b; Su et al., 2024), hidden states have been used to classify correctly and incorrectly answered reasoning samples. These representations — ranging from the hidden state of the final response token (Burns et al., 2023; Azaria & Mitchell, 2023b) to a geometric Chain-of-Embedding (CoE) (Wang et al., 2025) over reasoning chains — consistently outperform predominant metrics such as perplexity and entropy, with further gains achieved by supervised variants (Ji et al., 2024; Ni et al., 2025). The success of leveraging the encoded internal representations extends beyond knowledge gap detection to applications

*Yujing is an incoming PhD student at King’s College London, supervised by Hanqi Yan. Correspondence to Hanqi Yan (hanqi.yan@kcl.ac.uk).

such as fact verification (Xue et al., 2025), underscoring the powerful expressivity of LLM representations more broadly.

However, hidden states capture activated knowledge broadly, including those irrelevant to the query, such as input lengths and style variations, which do not principally affect the knowledge required to answer correctly. In other words, existing hidden-states based methods can be violated by unexpected trivial perturbations in the input.

We illustrate this in Figure 1 using a knowledge-intensive task, where external documents can be retrieved as additional context. We prompt the LLM to generate answers both with and without retrieved documents, then project the last-layer’s hidden states into 2D space via t-SNE. Red and green denote correctly and incorrectly answered responses, respectively, while squares and triangles denote responses generated with and without retrieved documents. We observe that hidden states are clearly separated by the presence or absence of retrieved documents, but *not* by whether the answer is correct. It demonstrates that adding additional information, even when it doesn’t help close the knowledge gap, can greatly affect the hidden states distribution.

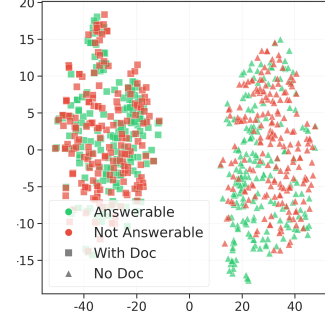


Figure 1: t-SNE visualizations of hidden states for answerable and unanswerable queries.

Building on these insights, we propose **GRADE**, a gradient-based dynamic analysis that measures the effective dimension of the parameter updates for the given query. Inspired by pioneering works showing that gradients localize knowledge-sensitive parameters via gradient-based Fisher Information (Kirkpatrick et al., 2017; Cha et al., 2025), we use the rank of the gradient matrix to distinguish the knowledge structure between correctly and incorrectly answered questions. Based on the observation that gradients lie within the subspace spanned by hidden states, we project the gradient into this shared subspace and apply a rank ratio to measure the effective dimension of query-specific update pressure represented within the activated parameter subspace. To alleviate the hyperparameter reliance, we introduce a spectrum-aware measurement that adaptively selects the most informative components for stable rank analysis, and train a probe to capture cross-layer rank ratio dynamics.

Our main contributions can be summarized below:

- We identify a systematic problem of existing hidden state methods in knowledge gap detection, i.e., their vulnerability to irrelevant input perturbations.
- We propose a gradient-based dynamic analysis method **GRADE** by measuring the stable rank ratio of the projected knowledge updates, for LLM knowledge gap detection. This method also enables the interpretation generation for long-form answers.
- Empirically, we demonstrate that **GRADE** consistently outperforms baselines across six datasets, including question answering, multiple-choice, and math reasoning in predicting question answerability and cross-dataset transferability.

2 Related Work

Knowledge gaps detection can be achieved by either merely evaluating the model’s outputs (black-box) or accessing the model’s internal states.

Black-box. *Verbalization* (Tian et al., 2023; Amayuelas et al., 2024; Ren et al., 2025) directly prompts LLMs to self-report their confidence or acknowledge their inability to answer a given question in the generated text. Despite the further enhancements, such as self-reflection capabilities of RLHF-tuned models, Zhou et al. (2024) found these methods are still largely hindered by overconfidence and miscalibration. *Sampling-based methods* (Kuhn et al., 2023; Lin et al., 2024; Gao et al., 2024; Xiong et al., 2024) evaluate the semantic consistency or agreement across multiple sampled responses to the same query. For instance, semantic uncertainty (Kuhn et al., 2023) estimates confidence by sampling multiple responses, clustering them by semantic equivalence, and computing entropy over the re-

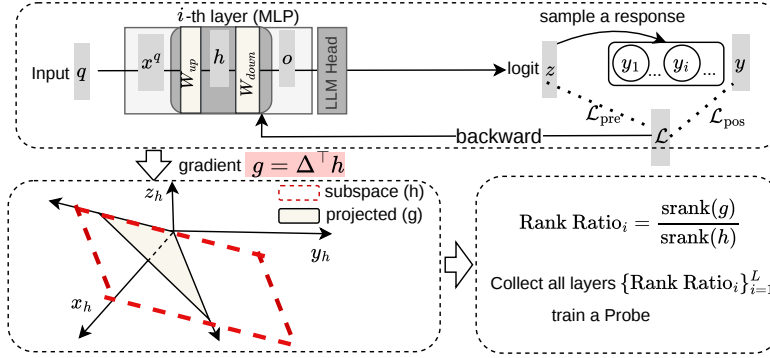


Figure 2: **GRADE** for knowledge gap probing. Given an input q , (i) *Forward pass*: compute hidden states h, o in the MLP block and loss $\mathcal{L}$, either before response generation or after; *Backward pass*: derive gradient g ; (ii) *Rank ratio calculation*: project gradients onto the subspace spanned by h and compute rank ratios based on stable rank (srank); (iii) *Probe training*: aggregate rank ratios across L layers to predict the knowledge gap.

sulting clusters. However, such methods are often computationally expensive, requiring multiple sampling passes and external models for semantic consistency evaluation.

White-box methods allow access to model internals, such as generated *token probability* and *hidden states* (activations). Entropy (Kadavath et al., 2022) or perplexity (Ren et al., 2023) from the model’s logits are the most-widely adopted baseline metrics for both knowledge gap (He et al., 2025) or uncertainty (Nikitin et al., 2024). More recent work also extends the token-level measurement to the atomic fact-level of the predictions (Fadeeva et al., 2024) and query-level (Chen et al., 2026), by computing a margin over sequence token probabilities (Wang et al., 2023). Moving deeper into the model’s architecture, i.e., internal activations, such as MLP outputs (Azaria & Mitchell, 2023b; Ni et al., 2025) and attention (Li et al., 2025b). For example, Li et al. (2025b) identify semantically crucial tokens by back-tracking through an attention chain to explain answer derivation. Azaria & Mitchell (2023b) and Ni et al. (2025) train a knowledge gap detector with the hidden as inputs and factual accuracy as output label. Our **GRADE** also leverages the hidden states, but with emphasis on leveraging gradient information to alleviate the reliance on spurious features in the inputs.

3 GRADE for Knowledge Gap Probing

The goal of knowledge gap detection is to determine whether the model $\mathcal{M}$ possesses sufficient latent knowledge to answer a given question q . We formalize this task as Knowledge Boundary (KB) Detection, where the mapping function F learns to map the model’s internal states to a continuous score s representing the knowledge gap:

$$F : \mathcal{I}_{\mathcal{M}}(q) \mapsto [0, 1], \quad (1)$$

where $\mathcal{I}_{\mathcal{M}}(q)$ represents the updated model’s internal states after feeding with q . Here, a knowledge gap refers to the disparity between a model knowing and not knowing how to answer a query, which empirically manifests as the intrinsic difference between generating correct and incorrect answers.

An effective detector should capture how much it contains knowledge specifically required to answer q . Many hidden state-based methods (Azaria & Mitchell, 2023b) directly use the output representations from the Multi-Layer Perceptron (MLP) blocks as the detector input, i.e., $\mathcal{I}_{\mathcal{M}} \leftarrow \text{MLP}(q)$. However, hidden states could encode information irrelevant to the required knowledge (see Figure 1), making them susceptible to spurious input variations. **GRADE** extracts features from $\mathcal{I}_{\mathcal{M}}(q)$ that capture the internal computation relevant to determining knowledge sufficiency.

Method overview. We use the gradient matrix rank to distinguish the knowledge structures of correctly and incorrectly answered questions. The overview of the method is shown in Figure 2. Given an input query q , a forward pass derives hidden states h in the MLP blocks and loss $\mathcal{L}$, conditioned on whether a sampled response y is available ($\mathcal{L}_{\text{pos}}$) or not ($\mathcal{L}_{\text{pre}}$). A backward pass then yields the gradients (§3.1). To measure the proportion of generation-induced update pressure represented within the activated parameter subspace, we project the gradients onto the subspace spanned by h and compute stable rank ratios (§3.2). Finally, rank ratios across all L layers are aggregated as input features to train a knowledge gap detection probe (§3.3).

3.1 Bi-Directional Information Flow

MLP feedforward. Given a Large Language Model $\mathcal{M}$ processing a sequence of length n , we focus on the MLP blocks, as they are widely recognized as the primary storage for factual information (Geva et al., 2023). Let $x^q \in \mathbb{R}^{n \times d_{\text{model}}}$ denote the MLP input hidden states, where d_{model} is the model dimension. The information flow through a gated MLP block in the LLM can be formalized as:

$$h = \sigma(x^q W_{\text{gate}}^\top) \odot (x^q W_{\text{up}}^\top), \quad o = h W_{\text{down}}^\top. \quad (2)$$

During the MLP feedforward process, we first obtain the *intermediate hidden states* $h \in \mathbb{R}^{n \times d_{\text{ff}}}$ via the two weight matrices W_{gate} and W_{up} , where σ is the non-linear activation function, $\odot$ denotes the element-wise product, and d_{ff} is the intermediate feed-forward dimension. The MLP output hidden states $o \in \mathbb{R}^{n \times d_{\text{model}}}$ are then computed via the projection matrix W_{down} (hereafter denoted simply as W).

Generation and loss function. From the LLM generation head, we obtain the logit $z \in \mathbb{R}^V$, after the projection from model dimension d_{model} to vocabulary V , from which tokens are sampled autoregressively to form the response $y = [y_1, y_2, \dots, y_{\text{out}}]$. We then define two loss objectives $\mathcal{L}$ depending on whether the response y has been generated, formally defined in Eq (3):

$$\mathcal{L}_{\text{pre}} = - \sum_{j=1}^V p(z_j) \log p(z_j), \quad \mathcal{L}_{\text{pos}} = - \sum_{t=1}^{|y_{\text{out}}|} \log p(y_t \mid y_{<t}, q), \quad (3)$$

where z_j is the logit for the j -th vocabulary token, and $p(\cdot)$ denotes the softmax function.

- For $\mathcal{L}_{\text{pre}}$, we calculate the entropy of z to measure how spread out the distribution is over the vocabulary. A low entropy implies that the model knows well what to generate next given the query.
- For $\mathcal{L}_{\text{pos}}$, we take the model’s own generated response y as a pseudo-label for supervision, and then use the standard cross-entropy loss to measure how far the model’s parameter updates can reach the final generation. Specifically, in long-form reasoning scenarios, the response is segmented into discrete steps based on punctuation markers. A step-wise cross-entropy loss is then applied, restricting the loss calculation to the tokens within each step. We thus generate the interpretation for the detected gap based on this mechanism, and the results are shown in §5.

Backward pass. We now apply the chain rule to derive the gradient $g \in \mathbb{R}^{d_{\text{model}} \times d_{\text{ff}}}$ of the loss with respect to W . The gradient derivation can be expanded as:

$$g = \frac{\partial \mathcal{L}}{\partial W} = \left(\frac{\partial \mathcal{L}}{\partial o} \right)^\top \frac{\partial o}{\partial W} = \Delta^\top h, \quad (4)$$

where $\Delta = \frac{\partial \mathcal{L}}{\partial o} \in \mathbb{R}^{n \times d_{\text{model}}}$ is the intermediate signal at the pre-activation layer. Consequently, the i -th row of g , $g_i^\top \in \mathbb{R}^{d_{\text{ff}}}$, can be written as:

$$g_i^\top = \sum_{t=1}^n \Delta_{t,i} h_t^\top, \quad (5)$$

a weighted sum of the input hidden states $\{h_t\}_{t=1}^n$. Since every row of g is a linear combination of these vectors, g lies entirely within the subspace spanned by h .

Gradient subspace illustration. Based on Eq. (5), the gradient lies within the subspace spanned by the hidden states, as illustrated in Figure 2. Specifically, the gradient (white triangle) resides within the subspace (bounded by the red dashed boundary) of the 3D space $(x_h - y_h - z_h)$ spanned by the hidden states. This motivates projecting both the gradient and hidden states onto their shared subspace, where the relative volume of the gradient subspace to the hidden state subspace serves as a natural proxy for the proportion of required knowledge updates within the activated knowledge. In what follows, we describe how the stable rank ratio operationalizes this proxy.

3.2 Gradient Projection and Rank Ratio

To compute the Rank Ratio as a proxy for the proportion of parameter query-specific update pressure represented within the activated parameter subspace, we first project both gradients and hidden states into a shared subspace (the region within the red dashed boundary in Figure 2), followed by the rank ratio calculation, which also serves as an implicit normalization against input length.

3.2.1 Projected Rank Calculation

To facilitate a direct comparison of ranks, we first project the parameter gradient into a unified representation space shared with the hidden states. Then, we employ the stable rank to robustly compute the effective dimensionality of required parameter updates.

Projection to a shared representation space. To measure how much gradient-introduced information occupies the total capacity of the hidden states, we take the subspace spanned by the hidden states h as the base space and project the gradient g into it. Formally, using the orthogonal projection matrix $P_h = h^\top (hh^\top)^\dagger h$ onto the row space of h (Penrose, 1955), the projected gradient can be expressed as $gP_h = Ah$, where $A = gh^\top (hh^\top)^\dagger$ represents the coordinate representation of the projected gradient under the hidden-state basis h .

We compute the covariance matrix of these projected gradient coordinates as $C_g = AA^\top \in \mathbb{R}^{n \times n}$, which simplifies to:

$$C_g = C_h^\dagger (hg^\top gh^\top) C_h^\dagger, \quad (6)$$

where $C_h = hh^\top$ is the Gram matrix of the hidden states, and $C_h^\dagger$ denotes its Moore-Penrose pseudoinverse. C_g reflects the covariance of the gradient dynamics projected within the hidden-state space.

To quantify the effective dimensionality of the projected gradient space, we compute the rank of C_g using Singular Value Decomposition (SVD). The singular value distributions are long-tailed, and recent studies suggest that the core semantics only lie in a low-dimensional manifold, so we retain only singular values above 10^{-6} for rank calculation. Figure 9 demonstrates the effectiveness of gradient rank as a discriminative signal for answerable and unanswerable questions.

Stable rank for robust rank estimation. Despite the effectiveness of naive rank, the hard threshold makes the naive rank estimate sensitive to numerical noise near the cutoff. As illustrated in Figure 10, truncating at 10^{-4} yields a rank of 269, whereas a naive threshold at 10^{-5} results in a rank of 299, leading to a substantial difference in rank calculation. Therefore, we employ the stable rank (Sanyal et al., 2020; Ipsen & Saibaba, 2025) to ensure a robust and stable calculation of effective dimensionality, as it offers a smooth, spectrum-aware measure that accounts for all singular values. Formally, let $\lambda_1^g \geq \lambda_2^g \geq \dots \geq \lambda_n^g \geq 0$ be the sorted singular values of C_g . The stable rank applied to the two objectives:

$$sr_{pre}(C_g) = \sum_{i=1}^n \frac{\lambda_i^g}{\lambda_1^g}, \quad sr_{pos}(C_g) = \sum_{i=1}^n \frac{(\lambda_i^g)^2}{(\lambda_1^g)^2}. \quad (7)$$

The stable rank provides a continuous measure that naturally concentrates on high-energy singular values while discounting the long tail, as illustrated in Figure 10.

3.2.2 Rank Ratio Calculation

Rank ratio as implicit normalization. To quantify the knowledge gap, we introduce the Gradient Subspace Rank Ratio, which evaluates the proportion of effective required updates within the activated parameters. The gradient subspace rank ratio is computed as $srank(C_g)/srank(C_h)$, where $srank(C_h)$ is computed analogously to Eq. (7). More importantly, since longer inputs simultaneously expand both the gradient and hidden state subspaces, their ratio remains stable, acting as a natural normalization against input length variability. To verify the justification, we calculate the correlation between metrics and the different input sequence lengths (results in Figure 3). Specifically, we select the queries from the HotpotQA (Yang et al., 2018) dataset that can’t be correctly answered even when the model is fed with retrieved external documents (longer inputs shown in the x-axis). We observe that the rank of hidden states (a) is sensitive to the changes of the input length, with 0.59 Pearson correlation. The rank ratio (c) is the most robust (0.18), with a significant improvement over the stable rank (0.36) of the gradient (b).

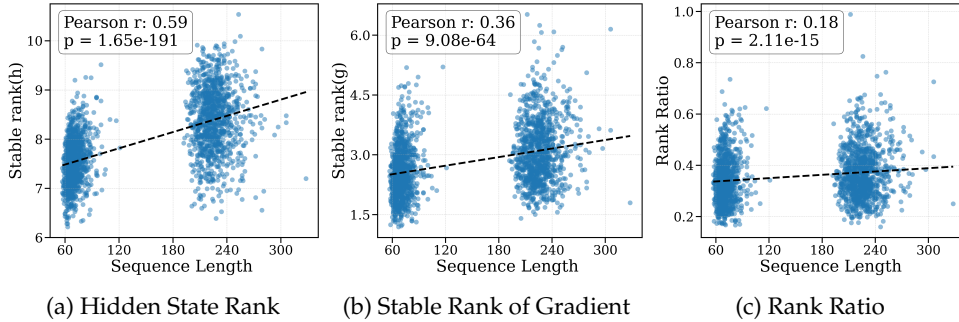


Figure 3: Pearson correlation between the input sequence length and the values of different rank-based metrics. The internals are extracted from Llama3.1-8B-Instruct model with HotpotQA as input. The rank ratio is the most robust metric to input length variation.

3.3 Supervised Probe based on Rank Ratio

To capture the dynamics of gradients across different layers, we gather an input feature consisting of $[RankRatio_1, RankRatio_2, \dots, RankRatio_L]$. This feature vector characterizes how ratio of required to activated knowledge evolves across layers, providing a richer propagation trajectory than scalar value-based thresholding. Empirical evidence in Table 6 validates that layer-wise features are more indicative of knowledge sufficiency than such threshold-based alternatives.

To train the supervised probe, we construct a dataset of paired inputs consisting of layer-wise feature vectors and binary labels, where each label indicates whether the model can correctly answer the given query. In practice, for each query, we sample multiple responses and assign a positive label (answerable) if the accuracy across samples exceeds an upper threshold, and a negative label (unanswerable) otherwise. Further details are in § A.1.4.

4 Experiments

4.1 Experimental Settings

Models and dataset. We use three backbone *models*, i.e., Llama3.1-8b (Grattafiori et al., 2024), Qwen2.5-7b (QwenTeam, 2025a) and Gemma2-9b (Team et al., 2024). Evaluation are conducted across six datasets, including math (GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021b)), MMLU (Hendrycks et al., 2021a), and free-form QA: NQ (Kwiatkowski et al., 2019), TQA (Joshi et al., 2017) and HotpotQA (Yang et al., 2018).

Baselines. We compare with both verbalization, i.e., Judge (Ren et al., 2025) and white-box, including token-probability and hidden states. P-Entropy (Kadavath et al., 2022) calculates

the cumulative token-level entropy over the generated sequence. Internal Confidence (IC) (Chen et al., 2026) is unsupervised, mapping the last layer hidden state of the last input token to a confidence score $P(\text{YES})$, and Align-P (Ni et al., 2025) is a supervised method by utilizing the intermediate hidden states before response generation.

Metrics. Following Wang et al. (2023) and Chen et al. (2026); Kuhn et al. (2023), we utilize Accuracy (Acc) to calculate the proportion of samples where the predicted answerability aligns with the actual task performance; Area Under the Receiver Operating Characteristic (AUROC) (Hanley & McNeil, 1982) as a robust evaluation insensitive to the class imbalance.¹ We focus on correctness-discrimination metrics rather than calibration metrics, as our objective is to identify knowledge boundaries instead of calibrating confidence. Further discussion and empirical comparisons with UQ calibration are provided in Appendix A.4.1.

4.2 Evaluation results of knowledge gap detection

Results for original queries. The knowledge gap detection for original queries in the evaluation benchmarks is shown in Table 1. Overall, our proposed $\text{GRADE}_{\text{pre}}$ and $\text{GRADE}_{\text{pos}}$ consistently outperform all baselines. Notably, $\text{GRADE}_{\text{pre}}$, which requires no response generation, achieves the best performance in 19 out of 36 settings, underscoring the strength of gradient-based signals as a lightweight yet effective indicator of knowledge sufficiency. Among the baselines, Align-P achieves the most competitive accuracy as a supervised probe, but suffers from severe AUROC degradation on complex datasets such as MMLU and HQA. Notably, the effectiveness of our two methods aligns with task complexity. For single-hop QA tasks, i.e., NQ and TQA, $\text{GRADE}_{\text{pre}}$ achieves the best with only the query-related information. For HQA and difficult reasoning datasets (MMLU, GSM8K and MATH), those requiring Chain-of-thought reasoning capabilities, $\text{GRADE}_{\text{pos}}$ can leverage its token-level information in the generated response to capture the procedure knowledge confidence. To further demonstrate the advantage of GRADE over other recent representation-probing and reasoning-chain baselines (such as CoE (Wang et al., 2025) and SAPLMA (Azaria & Mitchell, 2023a)), we provide an extended baseline comparison on our primary model backbone in Appendix A.4.2. The results show that GRADE consistently achieves superior accuracy and AUROC.

	Method	NQ		TQA		HQA		MMLU		GSM8K		MATH	
		Acc	AUROC	Acc	AUROC	Acc	AUROC	Acc	AUROC	Acc	AUROC	Acc	AUROC
Llama3.1-8b	Judge	0.523	-	0.652	-	0.536	-	0.519	-	0.639	-	0.501	-
	P-Entropy	0.616	0.643	0.645	0.691	0.596	0.607	0.701	0.728	0.643	0.496	0.661	0.704
	IC	0.640	0.669	0.661	0.715	0.747	0.651	0.504	0.651	0.518	0.639	0.575	0.627
	Align-P	0.771	0.793	0.784	0.855	0.737	0.806	0.759	0.506	0.692	0.766	0.753	0.607
	$\text{GRADE}_{\text{pos}}$	0.794	0.841	0.713	0.774	0.778	0.857	0.727	0.778	0.836	0.893	0.891	0.962
	$\text{GRADE}_{\text{pre}}$	0.830	0.888	0.813	0.895	0.754	0.810	0.703	0.751	0.714	0.767	0.770	0.841
Qwen2.5-7b	Judge	0.671	-	0.601	-	0.536	-	0.490	-	0.471	-	0.662	-
	P-Entropy	0.596	0.616	0.611	0.649	0.543	0.541	0.713	0.678	0.627	0.541	0.702	0.542
	IC	0.625	0.670	0.607	0.693	0.612	0.602	0.502	0.536	0.603	0.637	0.676	0.722
	Align-P	0.680	0.704	0.705	0.833	0.752	0.745	0.755	0.476	0.645	0.669	0.747	0.797
	$\text{GRADE}_{\text{pos}}$	0.745	0.802	0.719	0.776	0.786	0.866	0.734	0.758	0.719	0.772	0.879	0.921
	$\text{GRADE}_{\text{pre}}$	0.781	0.814	0.795	0.862	0.715	0.778	0.735	0.688	0.699	0.828	0.804	0.879
Gemma2-9b	Judge	0.574	-	0.692	-	0.573	-	0.472	-	0.513	-	0.400	-
	P-Entropy	0.588	0.607	0.631	0.655	0.547	0.538	0.702	0.765	0.739	0.794	0.691	0.751
	IC	0.588	0.650	0.649	0.698	0.562	0.655	0.505	0.567	0.507	0.541	0.715	0.757
	Align-P	0.734	0.730	0.725	0.799	0.764	0.855	0.782	0.534	0.727	0.776	0.613	0.636
	$\text{GRADE}_{\text{pos}}$	0.759	0.823	0.781	0.818	0.805	0.882	0.768	0.838	0.776	0.821	0.871	0.934
	$\text{GRADE}_{\text{pre}}$	0.814	0.907	0.832	0.932	0.815	0.868	0.750	0.809	0.745	0.783	0.794	0.813

Table 1: Main results on various datasets.

Robustness to input perturbation. To evaluate the robustness of the methods against linguistic variation without confounding changes in task difficulty, we use Qwen3-30b-instruct to rephrase the original queries while preserving their core semantics. Crucially, to avoid selection bias and isolate pure linguistic robustness, we retain only query pairs where the LLM’s answer correctness remains unchanged, thereby ensuring the ground-truth knowledge gap label is held fixed. A desirable knowledge gap detector should yield consistent

¹Implementation details are shown in Appendix A.1.

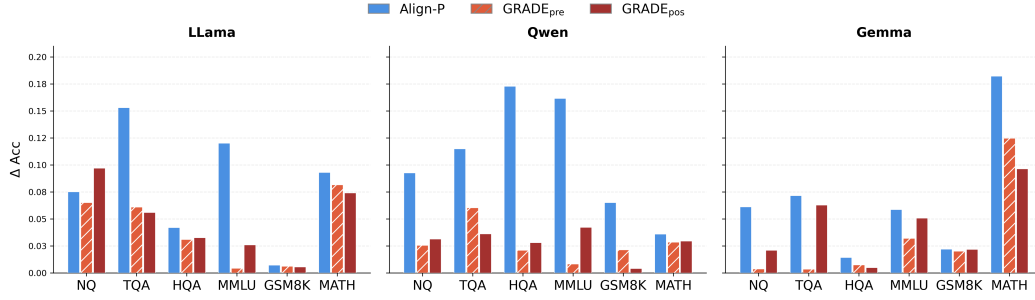


Figure 4: Relative change in detection accuracy (ΔAcc) before and after input paraphrase. Smaller changes imply that the method is more robust to the perturbation.

predictions on these rephrased inputs. We quantify detector robustness by measuring the relative performance change, defined as $|\text{Acc}_{\text{rephrased}} - \text{Acc}_{\text{original}}| / \text{Acc}_{\text{original}}$.

The evaluation results on the perturbed inputs are presented in Figure 4. The results clearly demonstrate that our proposed metrics, **GRADE_{pos}** and **GRADE_{pre}**, are more robust to input perturbations compared to baselines. In contrast, **Align-P** exhibits notable performance fluctuations when input phrasing changes. This confirms that **GRADE** ignores superficial linguistic variations and reliably isolates intrinsic representational signals associated with actual knowledge gaps. Detailed relative change breakdowns for both Accuracy and AUROC, including IC (Internal Confidence), are provided in Appendix A.4.3.

Robustness to length and context variations.

To isolate the impact of sequence length and external context, we conduct cross-condition transfer experiments by training and evaluating probes on symmetric subsets: (i) *long vs. short* sequences, and (ii) *with-document vs. without-document* settings. As presented in Table 2, **GRADE** consistently outperforms the baseline *Align-P* across most cross-condition transfer directions on NQ, TQA, and HQA. Notably, when transferring from short to long sequences or across context settings (e.g., $w/o \rightarrow w$ on HQA, where *Align-P*’s AUROC drops to 0.673), **GRADE** maintains high classification accuracy and AUROC (0.842). These results confirm that **GRADE** effectively isolates intrinsic knowledge gaps and is robust to changes in sequence length and style.

Direction	Method	NQ	TQA	HQA
<i>(a) Length Transfer: Long vs. Short</i>				
Long $\rightarrow$ Short	Align-P	0.687 / 0.733	0.747 / 0.826	0.654 / 0.726
	GRADE	0.782 / 0.828	0.775 / 0.840	0.727 / 0.801
Short $\rightarrow$ Long	Align-P	0.731 / 0.806	0.769 / 0.829	0.684 / 0.790
	GRADE	0.785 / 0.841	0.772 / 0.801	0.738 / 0.827
<i>(b) Context Transfer: With-Doc (w) vs. W/o-Doc (w/o)</i>				
$w \rightarrow w/o$	Align-P	0.716 / 0.799	0.682 / 0.780	0.751 / 0.821
	GRADE	0.784 / 0.861	0.720 / 0.788	0.809 / 0.862
$w/o \rightarrow w$	Align-P	0.684 / 0.785	0.725 / 0.817	0.680 / 0.673
	GRADE	0.750 / 0.843	0.752 / 0.809	0.762 / 0.842

Table 2: Cross-condition transfer performance (Acc / AUROC) across length and context variations.

4.3 Cross-Dataset Transferability

To assess the transferability, we evaluated **GRADE** by training on dataset A and testing on dataset B (see Figure 5). By looking at the results at non-diagonal areas, we observe that **GRADE_{pos}** demonstrates the most robust generalization, followed by **GRADE_{pre}** and **Align-P**. Moreover, **GRADE_{pos}** also demonstrates the best transfer patterns across different complexity datasets. Specifically, the transfer between similar complex datasets (TQA and NQ are single-hop) is shown within the red frame, and **GRADE_{pre}** and **Align-P** show a significant degradation within this area. Noted that both **GRADE_{pre}** and **Align-P** are methods using the generation logit before response sampling. In contrast, the superiority of **GRADE_{pos}** suggests that sequence-level gradient dynamics along the generated response capture how required knowledge accumulates and interacts across tokens, making it more robust to distribution shifts between tasks of different complexities.

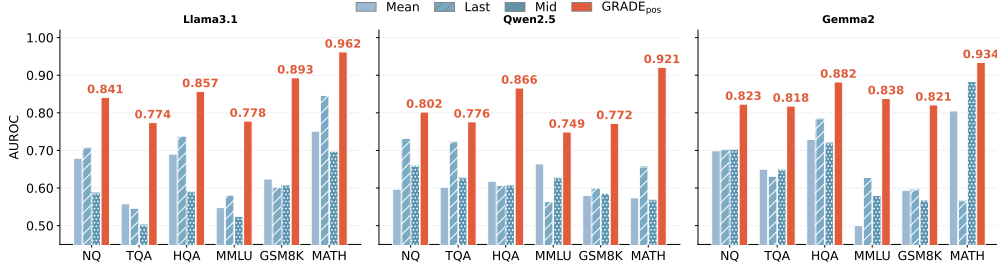


Figure 6: Comparison AUROC among detection thresholds (the mean, last-layer, and middle-layer rank ratio, respectively) versus **GRADE** (rank ratio patterns cross different layers).

5 Further Analysis

We further analyze the effects of key designs of the proposed probes, such as modeling the gradient dynamics across different layers and the layer selections as probe inputs. We also show the generated fine-grained interpretation for the long-form response.

Effects of modeling the cross-layer dynamics.

Some existing work suggests that layer-wise signals (Ni et al., 2025; Park et al., 2025) are reliable indicators of model behaviour, motivating a simple thresholding approach: take the rank ratio from a specific layer, determine an answerable/unanswerable threshold on the training set, and apply it to the

test set. We compare several such variants in Figure 6, where Mean, Last, and Mid denote using the average, last-layer, and middle-layer rank ratio as the thresholding signal, respectively. In contrast, **GRADE_{pos}** takes the full layer-wise rank² ratio vector as probe input, enabling it to learn cross-layer propagation patterns rather than relying on any single layer’s signal. The results confirm the effectiveness of this design, with a clear and consistent performance gap over all thresholding baselines (Acc results are shown in Figure 13). We also show the rank ratios across different layers for both answerable and unanswerable queries in Figure 11: it shows that their absolute values for the two categories are not clearly distinguishable, but the dynamics across different layers differ.

Ablation on Rank-Ratio vs. Vanilla Gradient Rank. To isolate the contribution of the rank-ratio normalization, we evaluate a baseline that directly uses the vanilla gradient rank vector as the probing feature without dividing by hidden-state ranks. As shown in Table 7, the proposed rank-ratio vector outperforms raw gradient rank across most settings on both LLaMA and Qwen backbones. This confirms that normalizing the gradient dynamics against hidden-state capacity effectively reduces sequence-length interference. Nevertheless, the vanilla gradient rank itself achieves strong accuracy and AUROC, demonstrating that parameter-level gradient dynamics inherently provides a powerful discriminative signal for the detection of knowledge boundaries.

Interpretation. As **GRADE_{pos}** considers the information across all the generated tokens, we can thus leverage it to generate token-level interpretation about the knowledge suf-

²We ablate the layer selections and the results in Table 12 showing that taking all layer-wise rank ratio is the best-performing setup.

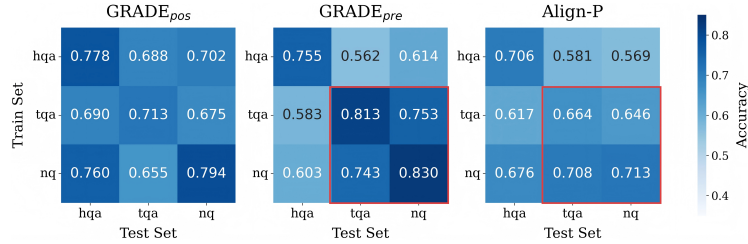


Figure 5: Cross-dataset generalization accuracy heatmaps. The results within the red box are transferred among similar complex questions (single-hop), i.e., between TQA and NQ.

iciency. Specifically, we use the t -th row-sum of $\mathcal{C}_g$ to identify the knowledge gap of token t . A high score indicates an epistemic shock, signifying a knowledge gap where the model requires large updates; a low score suggests that the token resides within the model’s knowledge boundary, requiring minimal adjustment. To ensure a fair and consistent comparison, the scores are globally normalized across all samples.

We show the generated uncertainty maps for both correctly and incorrectly answered queries from the GSM8k dataset in Figure 8a. We observe that the high scores are predominantly concentrated on critical logical anchors and mathematical operators, such as ‘multiply’ and the subsequent numeric result. This suggests that the model experiences momentary cognitive pressure only at pivotal decision points or computational junctions. Once these key steps are resolved, the scores rapidly decay, indicating that the rest of the reasoning chain resides safely within the internal knowledge. For the incorrect reasoning, we observe a rather diffuse pattern across the whole inputs, such as a high score is observed during the unit conversion process (e.g., ‘convert’, ‘ounces’), which persists throughout the following tokens. This pattern reflects a global epistemic collapse.

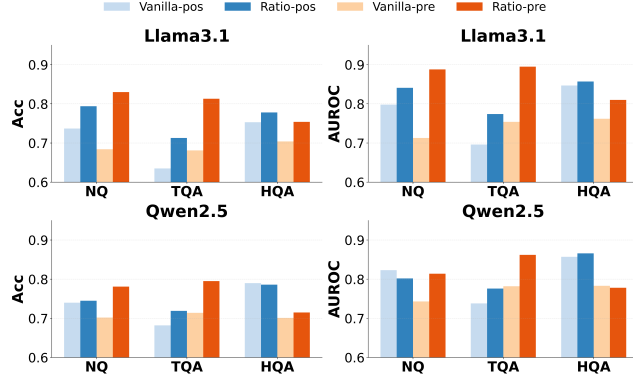


Figure 7: Ablation on the effectiveness of the rank ratio. We compare the proposed rank ratio with the vanilla gradient rank on Llama3.1 and Qwen2.5 across NQ, TQA, and HQA.

To find the total number of balls Julia bought, we need to multiply the number of packs of each color by the number of balls in each pack, and then add those amounts together. 3 packs of red balls * 19 balls per pack = 57 balls. 10 packs of yellow balls * 19 balls per pack = 190 balls, and 8 packs of green balls * 19 balls per pack = 152 balls. Adding these amounts together gives us 57 + 190 + 152 = 399 balls.

(a) Correct reasoning outcome.

To solve this problem, we need to convert the weight of the collection from pounds to ounces and then divide by the weight of a single bottle cap. Since there are 16 ounces in a pound, Josh's collection weighs 18 x 16 = 288 ounces. Then, we divide the total weight in ounces by the weight of a single bottle cap (7 ounces) to find the number of bottle caps: 288 / 7 = 41.14. Since we can't have a fraction of a bottle cap, we round down to the nearest whole number, 41.

(b) Incorrect reasoning sample.

Figure 8: Token-level metric across correctly and incorrectly answered examples.

6 Conclusion

We propose **GRADE** (**GR**Adient **DY**namics for knowl**E**dge gap detection), a gradient-based approach that addresses the fundamental challenge of detecting knowledge gaps for a given query. Unlike hidden-state methods that capture activated knowledge richness, **GRADE** captures the query-specific knowledge structure reflected in parameter updates. Specifically, **GRADE** quantifies the rank ratio between the gradient and hidden state subspaces, and uses the cross-layer ratio tendency as input to train a gap detector. Validation across six benchmarks demonstrates superior robustness and accuracy over verbalised, probabilistic, and hidden-state baselines, and a case study further illustrates how the gradient chain yields interpretable explanations of knowledge gaps in long-form answers.

Acknowledgment

This work was supported by the Start-up funding from King’s College London, and the Engineering and Physical Sciences Research Council (No. EP/Y028805/1).

References

- Janice Ahn, Rishu Verma, Renze Lou, Di Liu, Rui Zhang, and Wenpeng Yin. Large language models for mathematical reasoning: Progresses and challenges. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pp. 225–237, 2024.
- Alfonso Amayuelas, Kyle Wong, Liangming Pan, Wenhua Chen, and William Yang Wang. Knowledge of knowledge: Exploring known-unknowns uncertainty with large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 6416–6432, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.383. URL <https://aclanthology.org/2024.findings-acl.383/>.
- Amos Azaria and Tom Mitchell. The internal state of an llm knows when it’s lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 967–976, 2023a.
- Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it’s lying. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023b. URL <https://openreview.net/forum?id=y2V6YgLaW7>.
- Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=ETKGuby0hcs>.
- Sungmin Cha, Sungjun Cho, Dasol Hwang, and Moontae Lee. Towards robust and parameter-efficient knowledge unlearning for LLMs. In *The Thirteenth International Conference on Learning Representations*, 2025. URL <https://openreview.net/forum?id=1ExfUpmIW4>.
- Lihu Chen, Gerard de Melo, Fabian M. Suchanek, and Gaël Varoquaux. Query-level uncertainty in large language models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL <https://openreview.net/forum?id=11QZITAMU0>.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.
- Ekaterina Fadeeva, Aleksandr Rubashevskii, Artem Shelmanov, Sergey Petrakov, Haonan Li, Hamdy Mubarak, Evgenii Tsymbalov, Gleb Kuzmin, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, and Maxim Panov. Fact-checking the output of large language models via token-level uncertainty quantification. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 9367–9385, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.558. URL <https://aclanthology.org/2024.findings-acl.558/>.
- Xiang Gao, Jiaxin Zhang, Lalla Mouatadid, and Kamalika Das. SPUQ: Perturbation-based uncertainty quantification for large language models. In Yvette Graham and Matthew Purver (eds.), *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2336–2346, St. Julian’s, Malta, March 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.eacl-long.143. URL <https://aclanthology.org/2024.eacl-long.143/>.

- Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12216–12235, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.751. URL <https://aclanthology.org/2023.emnlp-main.751/>.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.
- James A Hanley and Barbara J McNeil. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.
- Zhitao He, Sandeep Polisetty, Zhiyuan Fan, Yuchen Huang, Shujin Wu, and Yi R Fung. Mmboundary: Advancing mllm knowledge boundary awareness through reasoning step confidence calibration. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16427–16444, 2025.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021a. URL <https://openreview.net/forum?id=d7KBjmI3GmQ>.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021b. URL <https://openreview.net/forum?id=7Bywt2mQsCe>.
- Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, 2025.
- Ilse CF Ipsen and Arvind K Saibaba. Stable rank and intrinsic dimension of real and complex matrices. *SIAM Journal on Matrix Analysis and Applications*, 46(3):1988–2007, 2025.
- Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL <https://openreview.net/forum?id=jKN1pXi7b0>.
- Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12):1–38, 2023.
- Ziwei Ji, Delong Chen, Etsuko Ishii, Samuel Cahyawijaya, Yejin Bang, Bryan Wilie, and Pascale Fung. LLM internal states reveal hallucination risk faced with a query. In Yonatan Belinkov, Najoung Kim, Jaap Jumelet, Hosein Mohebbi, Aaron Mueller, and Hanjie Chen (eds.), *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 88–104, Miami, Florida, US, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.blackboxnlp-1.6. URL <https://aclanthology.org/2024.blackboxnlp-1.6/>.
- Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In Regina Barzilay and Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1147. URL <https://aclanthology.org/P17-1147/>.

- Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.
- James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.
- Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl.a.00276. URL <https://aclanthology.org/Q19-1026/>.
- Moxin Li, Yong Zhao, Wenxuan Zhang, Shuaiyi Li, Wenya Xie, See Kiong Ng, Tat-Seng Chua, and Yang Deng. Knowledge boundary of large language models: A survey. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5131–5157, 2025a.
- Yinghao Li, Rushi Qiang, Lama Moukheiber, and Chao Zhang. Language model uncertainty quantification with attention chain. In *Second Conference on Language Modeling*, 2025b. URL <https://openreview.net/forum?id=QTrW2HWNXe>.
- Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL <https://openreview.net/forum?id=8s8K2UZGTZ>.
- Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Generating with confidence: Uncertainty quantification for black-box large language models. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL <https://openreview.net/forum?id=DWkJCSxKU5>.
- Potsawee Manakul, Adian Liusie, and Mark Gales. 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*, pp. 9004–9017, 2023.
- Shiyu Ni, Keping Bi, Jiafeng Guo, Lulu Yu, Baolong Bi, and Xueqi Cheng. Towards fully exploiting LLM internal states to enhance knowledge boundary perception. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 24315–24329, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1184. URL <https://aclanthology.org/2025.acl-long.1184/>.
- Alexander Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. Kernel language entropy: Fine-grained uncertainty quantification for llms from semantic similarities. *Advances in Neural Information Processing Systems*, 37:8901–8929, 2024.
- Seongheon Park, Xuefeng Du, Min-Hsuan Yeh, Haobo Wang, and Yixuan Li. Steer LLM latents for hallucination detection. In *Forty-second International Conference on Machine Learning*, 2025. URL <https://openreview.net/forum?id=UMqNQEPNT3>.
- R. Penrose. A generalized inverse for matrices. *Mathematical Proceedings of the Cambridge Philosophical Society*, 51(3):406–413, 1955. doi: 10.1017/S0305004100030401.
- QwenTeam. Qwen2.5 technical report, 2025a. URL <https://arxiv.org/abs/2412.15115>.
- QwenTeam. Qwen3 technical report, 2025b. URL <https://arxiv.org/abs/2505.09388>.

- Jie Ren, Jiaming Luo, Yao Zhao, Kundan Krishna, Mohammad Saleh, Balaji Lakshminarayanan, and Peter J Liu. Out-of-distribution detection and selective generation for conditional language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=kJUS5nD0vPB>.
- Ruiyang Ren, Yuhao Wang, Yingqi Qu, Wayne Xin Zhao, Jing Liu, Hua Wu, Ji-Rong Wen, and Haifeng Wang. Investigating the factual knowledge boundary of large language models with retrieval augmentation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 3697–3715, 2025.
- Amartya Sanyal, Philip H. Torr, and Puneet K. Dokania. Stable rank normalization for improved generalization in neural networks and gans. In *International Conference on Learning Representations*, 2020. URL <https://openreview.net/forum?id=H1enKkrFDB>.
- Weihang Su, Changyue Wang, Qingyao Ai, Yiran Hu, Zhijing Wu, Yujia Zhou, and Yiqun Liu. Unsupervised real-time hallucination detection based on the internal states of large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 14379–14391, 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.
- Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5433–5442, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.330. URL <https://aclanthology.org/2023.emnlp-main.330/>.
- Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=1PL1NIMMrw>.
- Yiming Wang, Pei Zhang, Baosong Yang, Derek F. Wong, and Rui Wang. Latent space chain-of-embedding enables output-free LLM self-evaluation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL <https://openreview.net/forum?id=jxo70B9fQo>.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.
- Miao Xiong, Zhiyuan Hu, Xinyang Lu, YIFEI LI, Jie Fu, Junxian He, and Bryan Hooi. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=gjeQKFxFpZ>.
- Boyang Xue, Fei Mi, Qi Zhu, Hongru Wang, Rui Wang, Sheng Wang, Erxin Yu, Xuming Hu, and Kam-Fai Wong. Ualign: Leveraging uncertainty estimations for factuality alignment on large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6002–6024, 2025.
- Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun’ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380, Brussels, Belgium, October–November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1259. URL <https://aclanthology.org/D18-1259/>.

Kaitlyn Zhou, Jena D. Hwang, Xiang Ren, and Maarten Sap. Relying on the unreliable: The impact of language models' reluctance to express uncertainty. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3623–3643, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.198. URL <https://aclanthology.org/2024.acl-long.198/>.

A Appendix

A.1 Implementation details

A.1.1 Data Setup

For open-domain QA datasets (NQ, TriviaQA, HotpotQA), we evaluate both no-context and with-doc settings. In the with-doc setting, we employ the Contriever (Izacard et al., 2022) dense retriever to retrieve relevant evidence passages. For each query, the top-1 retrieved document is selected and prepended to the question within the prompt. This document is inserted as an additional context block before the question to provide external evidence for answer generation.

A.1.2 Knowledge Gap Labeling

To determine answerability, we categorize queries based on their empirical accuracy across 10 samples for QA and multiple-choice tasks, and 5 samples for CoT reasoning. For the GSM8K dataset, we define the knowledge boundary using thresholds of 0.6 (answerable) and 0.4 (unanswerable), while other datasets utilize 0.8 and 0.2, respectively. As detailed in Table 3, the vast majority of samples (76.9%–92.4%) are preserved across all datasets, confirming that the filtering process maintains dataset integrity without discarding excessive data. Evaluation is conducted using Exact Match (EM) for QA and multiple-choice tasks, whereas CoT tasks are evaluated based on the correctness of the final result extracted from the reasoning chain.

Dataset	NQ	TQA	HQA	MMLU	GSM8K	MATH
Retained Ratio	91.5%	87.2%	92.4%	76.9%	87.8%	82.0%

Table 3: Retained sample ratio across different benchmark datasets after empirical accuracy filtering ($\text{Acc} \geq 0.8$ for answerable and $\text{Acc} \leq 0.2$ for unanswerable).

A.1.3 GRADE_{pos} for Long-form Generation

To evaluate GRADE_{pos} in extended reasoning scenarios, we apply our gradient-based analysis to segmented sequences. Following the generation of a complete reasoning chain, we segment the response into discrete steps $\{S_1, S_2, \dots, S_K\}$ based on punctuation markers (e.g., ", ."). For a given reasoning step S_k , let $n_k = n_q + \sum_{j=1}^k |S_j|$ be the cumulative length of the sequence up to the end of this step, where n_q denotes the length of the input prompt. The loss function of the current step is:

$$\mathcal{L}_{pos}^{(k)} = - \sum_{t \in S_k} \log p(y_t | \mathbf{x}_{<t}), \quad (8)$$

where only the tokens within the current step S_k contribute. Let $\mathbf{h}^{(k)} \in \mathbb{R}^{n_k \times d_{\text{ff}}}$ denote the intermediate hidden states, with $\mathbf{g}^{(k)}$ representing the corresponding step-specific gradient. Finally, to derive the metric GRADE_c for the complete sequence, we aggregate the step-specific stable rank ratios by averaging them across all K steps:

$$\text{GRADE}_{pos} = \frac{1}{K} \sum_{k=1}^K \frac{\text{srnk}(\mathbf{g}^{(k)})}{\text{srnk}(\mathbf{h}^{(k)})}. \quad (9)$$

While r_G captures the continuous dimensionality of the knowledge flux, it suffers from sensitivity to sequence length, since longer contexts inherently possess higher rank ceilings and introduce more baseline degrees of freedom. Let $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_s > 0$ be the sorted eigenvalues of the hidden Gram matrix $HH^\top$. We define the effective rank of the

hidden-state r_H similarly:

$$r_H = \sum_{i=1}^s \frac{\sigma_i}{\sigma_1}$$

Finally, we propose the Gradient Subspace Rank Ratio (GSRR) to quantify the relative geometric alignment between the parameter update and the representation subspace:

$$\text{GSRR} = \frac{r_G}{r_H}.$$

A.1.4 Probe Architecture

The alignment probe is implemented as a 5-layer feed-forward neural network designed to progressively compress input features into a scalar probability ($\text{num_layers} \rightarrow 256 \rightarrow 128 \rightarrow 64 \rightarrow 32 \rightarrow 1$). Each hidden block consists of a linear transformation followed by Batch Normalization, a Leaky ReLU activation (negative slope = 0.01), and a Dropout layer ($p = 0.5$) to enhance generalization and prevent overfitting. We employ Kaiming uniform initialization for all linear weights and initialize batch normalization parameters to unit scale and zero bias. The final layer utilizes a Sigmoid activation to map the output to $[0, 1]$.

A.1.5 Probe Training

We implement the training pipeline for the diagnostic probe using PyTorch. To ensure strict reproducibility across all experiments, the global random seed is fixed to 42. The probe is trained for a maximum of 100 epochs with a batch size of 64. For optimization, we initialize the learning rate at 5×10^{-3} and apply a weight decay of 1×10^{-5} to regularize the network and mitigate overfitting. To facilitate stable convergence and avoid local minima, we incorporate a learning rate scheduler that dynamically reduces the learning rate by a factor of 0.75 if no improvement is observed over a patience window of 5 epochs. Finally, during the evaluation phase, we adopt a standard decision threshold of 0.5 to binarize the probe’s continuous output into a definitive prediction.

A.1.6 Paraphrase Generation

For each query in datasets, we generate a semantically equivalent paraphrase using Qwen3-30B-A3B-Instruct-2507 (QwenTeam, 2025b). To ensure the perturbations only affect the linguistic form without altering the core factual or logical requirements, we employ a zero-shot prompting strategy. The system is provided with the following instruction:

System Prompt: “You are a helpful assistant. Please paraphrase the following question accurately without changing its original meaning or entity names. Output ONLY the paraphrased question.”

A.1.7 Calculation of AUROC

The Area Under the Receiver Operating Characteristic curve (AUROC) is a threshold-independent metric utilized to evaluate the discriminative capability of our gradient-based probing classifier. It is particularly advantageous in our setting as it remains robust against class imbalance—a common scenario when evaluating the proportion of answerable versus unanswerable queries in LLMs.

Let $\mathcal{D}^+$ represent the set of positive samples (i.e., queries the model answers correctly) and $\mathcal{D}^-$ represent the set of negative samples (i.e., queries exposing a knowledge gap). Let $s(x) \in [0, 1]$ denote the predicted confidence score output by our probe for a given query x . For any decision threshold $\tau \in [0, 1]$, the True Positive Rate (TPR) and False Positive Rate (FPR) are formally defined as:

$$\begin{aligned} \text{TPR}(\tau) &= \frac{|\{x \in \mathcal{D}^+ \mid s(x) \geq \tau\}|}{|\mathcal{D}^+|} \\ \text{FPR}(\tau) &= \frac{|\{x \in \mathcal{D}^- \mid s(x) \geq \tau\}|}{|\mathcal{D}^-|}. \end{aligned}$$

The ROC curve is constructed by plotting $\text{TPR}(\tau)$ against $\text{FPR}(\tau)$ as the threshold τ varies from 1 down to 0. The AUROC is geometrically defined as the integral of this curve:

$$\text{AUROC} = \int_0^1 \text{TPR}(\tau) d(\text{FPR}(\tau)).$$

A.2 Dataset Descriptions

GSM8K (Cobbe et al., 2021) is a collection of 8,500 high-quality grade-school math word problems designed to evaluate models’ abilities in multi-step arithmetic reasoning. The dataset focuses on interpretable numerical reasoning rather than symbolic manipulation, making it a standard benchmark for assessing mathematical problem-solving in natural language.

MATH (Hendrycks et al., 2021b) contains more than 12,000 challenging competition-level mathematics problems covering algebra, geometry, number theory, combinatorics, and calculus. Each problem includes step-by-step solutions, enabling rigorous evaluation of advanced mathematical reasoning and multi-step derivation abilities.

The Massive Multitask Language Understanding (MMLU) benchmark (Hendrycks et al., 2021a) consists of 57 multiple-choice subjects spanning STEM, humanities, social sciences, and professional domains. It assesses models’ world knowledge and problem-solving skills across a broad range of disciplines, making it one of the most comprehensive evaluations of general knowledge reasoning.

Natural Questions (NQ) (Kwiatkowski et al., 2019) comprises real anonymized queries issued to the Google search engine, paired with evidence passages from Wikipedia. It evaluates open-domain question answering, requiring systems to locate and extract relevant information from long documents under realistic information-seeking scenarios.

TriviaQA (TQA) (Joshi et al., 2017) is a large-scale open-domain question answering dataset consisting of over 95,000 question-answer pairs authored by trivia enthusiasts. Each question is paired with evidence documents from the web or Wikipedia, enabling evaluation of retrieval-based QA, reasoning over long contexts, and robust answer extraction.

HotpotQA (HQA) (Yang et al., 2018) is an open-domain QA dataset featuring multi-hop reasoning questions that require retrieving and integrating information from multiple Wikipedia articles. It supports both extractive and abstractive answers and provides supporting facts, enabling evaluation of interpretable multi-step reasoning.

A.3 Baselines Implementation

Judge: The Prior Judge baseline evaluates epistemic uncertainty by directly prompting the Large Language Model (LLM) to explicitly state whether it possesses the knowledge to answer a question.

(1) In the closed-book setting, we use the prompt: "Are you sure to accurately answer the following question based on your internal knowledge? If yes, you should give a short answer with one or a few words; if no, you should answer 'Unknown'. Question: question Answer: "".

(2) For the Retrieval-Augmented Generation (RAG) setting, the prompt is adapted to: "Given the following information: context Can you answer the following question based on the given information or your internal knowledge, if yes, you should give a short answer with one or few words, if no, you should answer 'Unknown'. Question: question Answer: "".

During inference, we enforce greedy decoding (temperature 0.0) and restrict the maximum generation length to 16 tokens to ensure deterministic and concise self-assessment. The generated text is then mapped to a binary confidence indicator $\hat{c} \in \{0, 1\}$. If the response contains the keyword "Unknown" (case-insensitive), we assign a confidence score of $\hat{c} = 0$, indicating a lack of knowledge. Conversely, any other generated short answer yields $\hat{c} = 1$.

P-entropy: The Predictive Entropy baseline quantifies epistemic uncertainty by aggregating the token-level Shannon entropy across the entire generated response. For a given instruction of length L_{instr} and a generated response of length L_{resp} , we extract the probability distribution $P_M(x_t|x_{<t})$ over the model’s full vocabulary V at each generation step t . The total predictive entropy H is then calculated as the sum of the step-wise entropies:

$$H = - \sum_{t=L_{instr}+1}^{L_{instr}+L_{resp}} \sum_{x_t \in V} P_M(x_t|x_{<t}) \log P_M(x_t|x_{<t}).$$

IC (Internal Confidence): We follow the training-free implementation of Internal Confidence. Rather than performing any data-driven search for an optimal center, we use a fixed decision center positioned at the last layer and the last token position. The confidence score is computed by aggregating the log-probability of $P(\text{YES})$ from this fixed center alongside its immediate spatial (neighboring tokens) and depth (neighboring layers) neighborhoods.

For a prompt of length N and a model with L layers, let the hidden state at layer l and token position n be $h_n^{(l)}$. Instead of relying solely on the final output token, these intermediate representations are extracted. Each hidden state is individually passed through the model’s language modeling head to isolate the probability of predicting the “Yes” target token, denoted as $P(\text{YES}|h_n^{(l)})$. To compute the final continuous confidence score, these intermediate probabilities are aggregated using a weighted average. This aggregation is built around an optimal “decision center”—the specific layer and token position where the model most effectively separates answerable from non-answerable queries. The Internal Confidence (IC) is formally defined as:

$$IC(h) = \sum_{n=1}^N \sum_{l=1}^L w_n^{(l)} P(\text{YES}|h_n^{(l)})$$

where $w_n^{(l)}$ denotes the weight assigned to the hidden representation $h_n^{(l)}$. These weights are determined by a positional attention mechanism that applies a distance-based decay originating from the decision center, placing the highest emphasis on the final k tokens and the deepest layers of the model.

A.4 Additional experiment results

A.4.1 Uncertainty Quantification vs. Knowledge Boundary Detection

While both Uncertainty Quantification (UQ) and Knowledge Boundary Detection (KBD) utilize binary label correctness for evaluation, they differ in core assumptions and target objectives:

- **Underlying Assumptions and Failure Modes:** Standard UQ methods assume that a model’s internal confidence (or entropy/consistency) aligns with prediction correctness. However, LLMs frequently suffer from severe overconfidence—generating consistent yet incorrect answers for queries outside their knowledge base, leading traditional UQ metrics to misinterpret certainty for correctness.
- **Evaluation Focus (Calibration vs. Discrimination):** Standard UQ focuses on *calibration* (e.g., Expected Calibration Error), measuring how well confidence scores align with accuracy. In contrast, Knowledge Boundary Detection focuses explicitly on *correctness discrimination*—identifying whether required latent knowledge is present in the model, regardless of its output confidence.

Empirical Evaluation on “Confidently Wrong” Cases. To compare these approaches when confidence signals fail, we construct a hard evaluation set of **500 “confidently wrong” samples** where the LLM repeatedly generates identical incorrect answers across multiple independent sampling passes. A robust knowledge boundary detector should identify these queries as *unanswerable* despite high confidence.

As shown in Table 4, unsupervised UQ baselines perform poorly: *Self-consistency* (Manakul et al., 2023) achieves 0.0% accuracy due to identical generation paths, and *P-entropy* (Kadavath et al., 2022) achieves only 32.8%. Supervised *Align-P* (Ni et al., 2025) reaches 67.5%. In contrast, **GRADE** achieves the highest accuracy of **72.6%**, demonstrating that rank ratio analysis on projected gradient dynamics effectively detects knowledge gaps under overconfidence.

Category	Method	Accuracy (%)
Unsupervised UQ	P-entropy (Kadavath et al., 2022)	32.8
	Self-consistency (Manakul et al., 2023)	0.0
Supervised Baseline	Align-P (Ni et al., 2025)	67.5
Ours	GRADE	72.6

Table 4: Detection accuracy on 500 “confidently wrong” queries where the LLM produces consistently incorrect answers. Higher accuracy indicates superior capability in identifying knowledge gaps despite high output confidence.

A.4.2 Extended Baseline Evaluations

To further evaluate $\text{GRADE}_{\text{pre}}$ against a broader range of internal-representation probing and self-evaluation approaches, we expand our empirical comparisons by including three additional baselines on LLaMA-3-8B across all six benchmark datasets: (1) **CoE** (Wang et al., 2025), which models progressive layer-wise embeddings as a latent reasoning trajectory; (2) **SAPLMA** (Azaria & Mitchell, 2023a), a supervised probing method targeting final-token hidden activations from generated responses; and (3) **Self-Consistency** (Manakul et al., 2023), a widely used non-probing self-evaluation baseline measuring generation stability across multiple sample paths.

As shown in Table 5, $\text{GRADE}_{\text{pre}}$ consistently achieves superior accuracy and AUROC compared to these baselines on almost all tasks. While CoE and SAPLMA show competitive accuracy on individual benchmarks (e.g., MMLU), their AUROC performance is significantly lower than $\text{GRADE}_{\text{pre}}$, indicating weaker overall ranking capability for knowledge boundary detection. These results confirm that gradient-subspace rank features offer a more reliable discriminative signal than standard hidden-state probing methods or output repetition counts.

Method	NQ		TQA		HQA		MMLU		GSM8K		MATH	
	Acc	AUROC	Acc	AUROC	Acc	AUROC	Acc	AUROC	Acc	AUROC	Acc	AUROC
CoE	0.790	0.626	0.605	0.634	0.643	0.511	0.724	0.554	0.617	0.666	0.693	0.537
SAPLMA	0.716	0.788	0.712	0.784	0.729	0.796	0.664	0.523	0.619	0.602	0.716	0.790
self-consistency	0.572	0.830	0.698	0.791	0.535	0.825	0.583	0.602	0.372	0.423	0.491	0.500
$\text{GRADE}_{\text{pre}}$	0.830	0.888	0.813	0.895	0.754	0.810	0.703	0.751	0.714	0.767	0.770	0.841

Table 5: Performance comparison with updated baselines.

A.4.3 Detailed Perturbation Analysis Including AUROC and IC Baseline

To provide a comprehensive evaluation of detector robustness under input perturbation, we extend our comparison to include the IC baseline as well as the relative change in AUROC performance. Table 6 presents the relative performance fluctuations (%) for both Accuracy and AUROC before and after query paraphrasing.

Analysis. Across the vast majority of benchmarks, **GRADE** models exhibit consistently smaller performance fluctuations in both Accuracy and AUROC compared to representation- and logit-based baselines like IC and Align-P. Specifically, while Align-P suffers severe AUROC degradation on complex reasoning datasets (e.g., 27.6% drop on MMLU and 42.0% on MATH), **GRADE** maintains exceptional stability (less than 2% shift on MMLU), further proving its resilience against superficial syntactic changes.

Method	NQ	TQA	HQA	MMLU	GSM8K	MATH
<i>(a) Accuracy Relative Change (%)</i>						
IC	21.7	3.9	3.8	3.3	16.2	21.7
Align-P	7.5	15.8	4.2	11.9	0.7	9.3
GRADE _{pos}	9.6	5.2	3.2	2.6	0.5	7.4
GRADE _{pre}	6.5	6.1	3.1	0.4	0.6	8.1
<i>(b) AUROC Relative Change (%)</i>						
IC	4.0	2.1	0.4	2.0	1.8	20.5
Align-P	3.6	17.3	5.9	27.6	2.3	42.0
GRADE _{pos}	3.7	4.7	3.3	1.7	1.2	3.5
GRADE _{pre}	3.0	10.1	8.3	1.8	1.8	6.3

Table 6: Relative performance fluctuations (%) under input perturbation for Accuracy (a) and AUROC (b). Smaller values indicate higher robustness to linguistic variations.

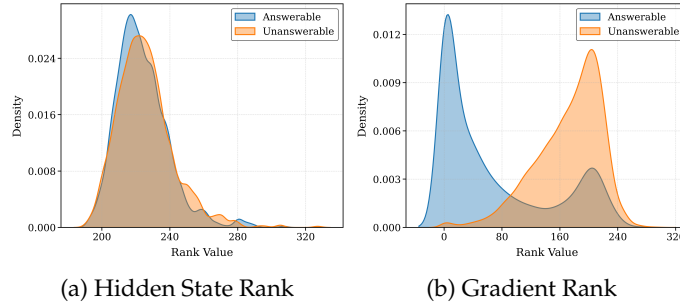


Figure 9: Density distributions of rank(Hidden States) in (a) and rank (Gradient) in (b) for Answerable and Unanswerable queries from the HotpotQA dataset.

A.4.4 Rank Calculation is sensitive to threshold selection.

As illustrated in Figure 10, truncating at $1e-4$ yields a rank of 269, whereas a naive threshold at $1e-5$ results in a rank of 299, leading to a substantial difference in rank calculation. Both including substantial long-tail components. Therefore, we introduce the stable rank as a consistent and robust measure of effective dimensionality.

A.4.5 Effects of layer selections.

(a) Dynamics of the layer-wise rank ratio. To elucidate why a holistic, cross-layer dynamic modelling is necessary, we show the trends for both answerable and unanswerable examples in Figure 11. The distinguishability between answerable and unanswerable queries varies inconsistently across layers — no single layer dominates, and the absolute differences remain marginal throughout. This motivates a cross-layer approach that leverages all layer-wise representations to capture the nuanced dynamics underlying answerability.

(b) Ablation of using rank ratios from different layers as probe input. We conduct an ablation study on the choice of Transformer layers used as input features for the probe. Figure 12 illustrates the classification performance when utilizing metrics from different layer subsets. The results demonstrate that aggregating signals from all layers yields the optimal performance, while utilizing later layers generally outperforms relying on earlier ones.

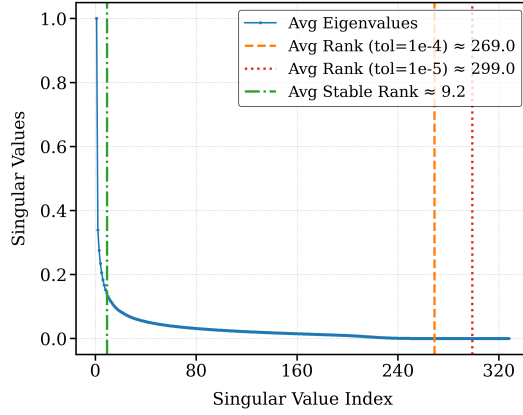


Figure 10: Average eigenvalue spectrum on HotpotQA using Llama-3-8B-Instruct model. It shows that a trivial change in the threshold leads to larger rank fluctuation, motivating the proposal of the stable rank.

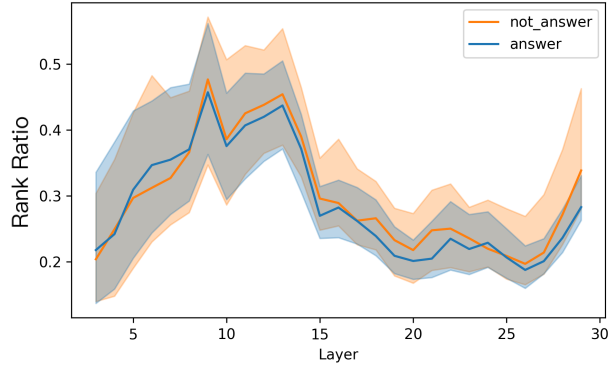


Figure 11: Rank ratio across different layers for correctly and incorrectly answered samples.

A.4.6 Acc results of comparison between GRADE and the threshold based methods.

In addition to the AUROC results in Figure 6, we show the Acc in Figure 13. The trends are consistent that our **GRADE** performs better across all the datasets and models compared to the layer-wise threshold baselines (mean, last and mid shown here).

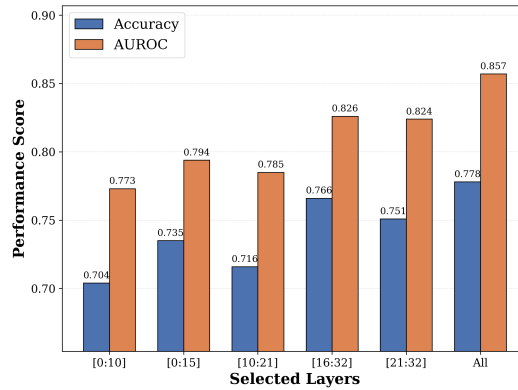


Figure 12: Performance comparison across different selected layers.

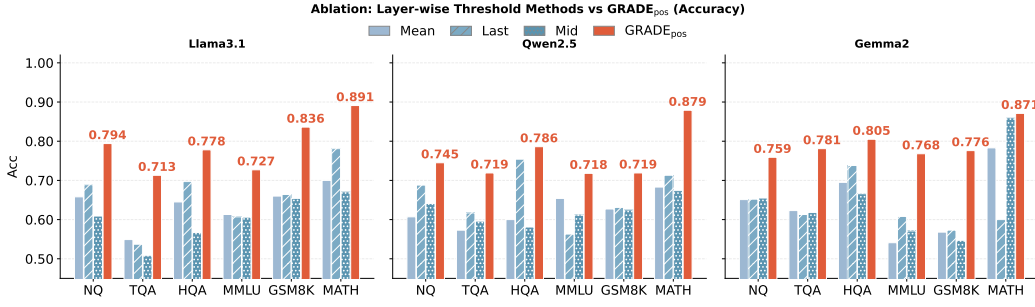


Figure 13: Comparison results (Acc) among detection threshold (the mean, last-layer, and middle-layer rank ratio, respectively) versus **GRADE** (rank ratio patterns cross different layers).

A.4.7 Computational Complexity and Resource Overhead

GRADE introduces minimal computational overhead, requiring only a single backward pass per reasoning step without parameter updates or iterative optimization. Table 7 presents the additional computational time and memory deployment cost on GSM8K, along with their relative increase compared to the standard forward pass.

Backbone	Extra Time	Extra Memory
Llama-8B	0.063 s (2.2%)	2857.11 MB (17.1%)
Qwen-7B	0.049 s (1.9%)	3266.42 MB (20.9%)
Gemma-9B	0.116 s (2.6%)	2527.18 MB (12.7%)

Table 7: Extra computation time and memory deployment cost introduced by the backward pass per reasoning step on GSM8K, along with relative increase percentages compared to the forward pass.

Across all evaluated backbones, the backward pass introduces at most 0.116 seconds (a maximum relative time increase of only 2.6%) per reasoning step. While computing gradients requires additional memory deployment for activations, the extra footprint remains bounded within 20.9%, demonstrating that **GRADE** is efficient and highly scalable for practical deployment.