

STEP: Soft Trajectory Estimation of Prediction Correctness via Per-Step Probability Scoring

Chenyang Yu^{1*} Zifan Zeng^{1,2} Qunli Zhang² Feng Liu²

¹Technical University of Munich

²RAMS Lab, Huawei Heisenberg Research Center

chenyang036.yu@tum.de, zifan.zeng@huawei.com, zhangqunli1@huawei.com, feng.liu1@huawei.com

Abstract

Estimating the correctness of large language model (LLM) predictions without access to ground truth is critical for reliable deployment in high-stakes settings. Existing approaches, such as entropy and perplexity, operate solely on final output statistics, discarding the rich information encoded in intermediate reasoning steps. To address this limitation, we propose *STEP* (*Soft Trajectory Estimation of Prediction correctness*), a training-free framework that scores each reasoning step via lightweight forward passes to obtain a soft probability distribution q_i over candidate answers, from which we derive a set of *soft trajectory features* capturing per-step confidence and its progression toward the final answer. Extensive experiments on various benchmarks demonstrate consistent and substantial improvements, particularly in AUROC, over output-based baselines and hard-indicator trajectory methods. Notably, *Temporally-Weighted Consistency*, which assigns higher importance to later reasoning steps where answer distributions have stabilized, contributes the largest gains, highlighting the importance of temporal structure in reasoning trajectories. These results further validate the advantage of preserving continuous probability signals over binary discretization.

1 Introduction

Large language models (LLMs) have demonstrated remarkable capabilities across a wide range of reasoning tasks, from mathematical problem solving to multi-step commonsense inference. Despite their impressive performance, these models remain prone to generating incorrect answers with high apparent confidence, making correctness estimation at inference time a critical requirement for

reliable deployment. It is essential, particularly in high-stakes settings such as medical question answering, legal reasoning, and automated decision support, where acting on a wrong prediction can carry serious consequences.

Existing methods typically rely on final output statistics derived from the model’s token-level distributions: entropy over the output (Malinin and Gales, 2021; Kuhn et al., 2023), token-level perplexity (Guo et al., 2017), or the probability assigned to the most likely answer (Hendrycks and Gimpel, 2017). Although these signals are simply accessible from the model’s logits, they operate at the level of individual tokens and capture only local fluency and output sharpness (Guo et al., 2017). They do not reflect whether the model has engaged in coherent, logically grounded reasoning at a semantic level (Kuhn et al., 2023). A model can assign high probability to a wrong answer simply because that answer is superficially plausible (Kadavath et al., 2022); conversely, a correct answer may receive modest token-level probability when the model remains genuinely uncertain yet ultimately reaches the right conclusion. Without examining the reasoning process itself, such methods lack a principled connection to semantic correctness.

Another approach to correctness estimation is to dive into the latent space of the model. A line of work has shown that the internal representations of LLMs carry signals correlated with factual correctness, and lightweight probes trained on hidden states can reliably predict output correctness (Azaria and Mitchell, 2023; Kossen et al., 2024). However, even lightweight probes require labeled training data and are sensitive to distribution shift, limiting their applicability when the task domain or model changes. Our approach requires no such training, and is thus applicable out of the box across tasks and models.

* Work done during an internship at the RAMS Lab, Huawei Heisenberg Research Center.

Another line of work exploits the internal geometry of reasoning trajectories. Wang et al. (2025) and Bi et al. (2025) derive correctness signals from the trajectory of hidden states across layers, showing that correct and incorrect responses exhibit measurably different geometric patterns in embedding space. A related but distinct approach is taken by Zhang et al. (2025), who operate in the output probability space rather than the latent space: correct trajectories exhibit high consistency, where intermediate steps steadily converge toward the final answer, while incorrect trajectories deviate erratically before settling on a wrong conclusion. However, this framework has two limitations that prevent direct application to inference-time correctness estimation. First, it requires sampling N independent trajectories per prompt, which scales inference cost proportionally. Second, correctness is measured via hard binary indicators that discard the full spectrum of confidence information, treating near-chance and near-certain probabilities identically.

In this work, we propose a training-free framework for correctness estimation of LLM output in closed-set tasks, which represent a broad and practically important class of real-world applications: from standardized tests and professional certification exams to clinical decision support and automated document classification, many high-stakes settings inherently present a finite, predefined set of options. Inspired by the trajectory-based insight of Zhang et al. (2025), we address both limitations by re-scoring the already-generated reasoning trajectory: truncating it at each intermediate step i and performing a lightweight forward pass to obtain a soft probability distribution q_i over all candidate answers, requiring no additional generation and no resampling.

This per-step distribution is a strictly richer signal than a binary correct/incorrect label: it captures not just which answer the model favors at each step, but how confidently and consistently it does so across the full trajectory. From these distributions, we derive a family of **soft trajectory features** reflecting distinct aspects of reasoning dynamics (Section 3).

We conduct extensive experiments across 4 datasets and 2 models, evaluating our features against output-based methods, training-free trajectory-based methods, and latent-trajectory approaches. Our soft features achieve consistent and substantial improvements, particularly in AUROC,

across models and datasets. Our contributions are as follows:

1. We propose **STEP**, a single-trajectory, training-free framework for correctness estimation in closed-set tasks, grounded in per-step soft probability distributions over the candidate answer set.
2. In STEP, we introduce three soft trajectory metrics that preserve continuous confidence signals discarded by prior binary-indicator methods.
3. We provide systematic empirical evidence across four benchmarks and two open-source models that soft trajectory features achieve strong and consistent performance relative to existing baselines, and find that later reasoning steps carry disproportionately more predictive signal.

2 Related Work

Output-statistics-based uncertainty estimation.

A foundational line of work estimates output reliability from token-level statistics derived from a single forward pass. Predictive entropy over the output distribution (Malinin and Gales, 2021), length-normalized log-probability, and the maximum softmax probability (Hendrycks and Gimpel, 2017) are canonical examples. These signals are cheap to compute but operate at the level of individual tokens: they capture local fluency and output sharpness rather than semantic correctness. A model can assign high token-level probability to a wrong answer that is superficially fluent, and low probability to a correct but uncertain answer (Guo et al., 2017).

Semantic and sampling-based methods.

To capture meaning-level uncertainty, Kuhn et al. (2023) introduce *semantic entropy*, which clusters multiple sampled responses by semantic equivalence and computes entropy over the resulting distribution, substantially improving over token-level baselines. Farquhar et al. (2024) extend this to a practical hallucination detection method, demonstrating its effectiveness at scale. However, both approaches require sampling N responses per prompt, which is a significant inference overhead that limits deployment in latency-sensitive settings.

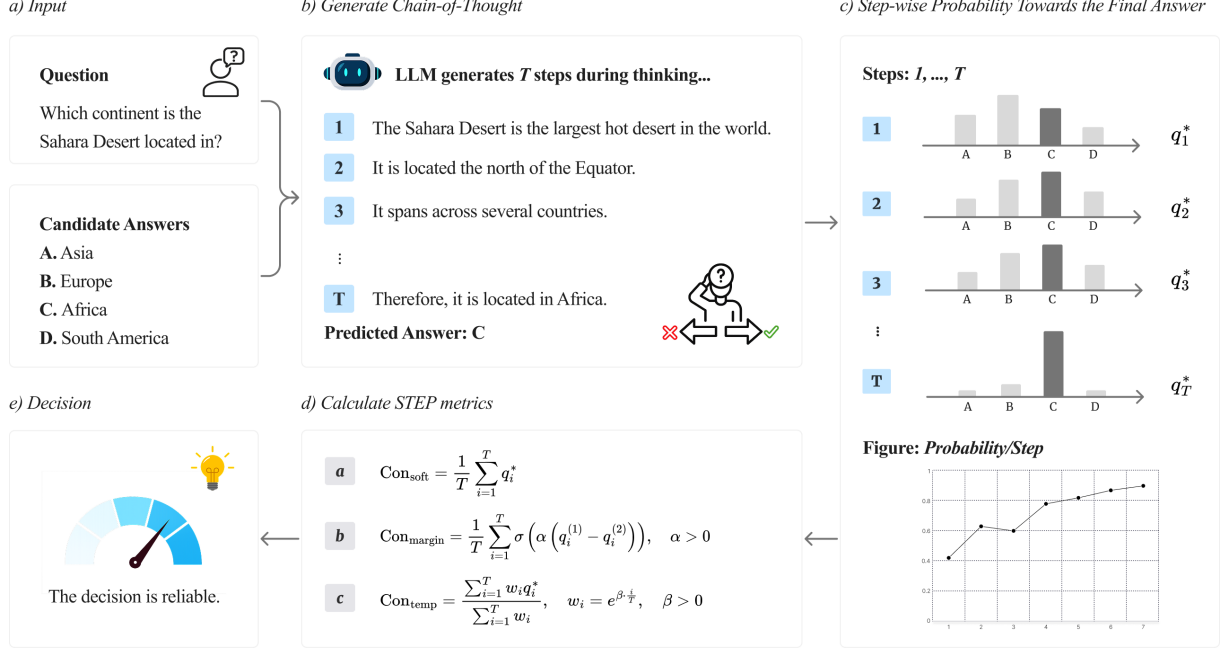


Figure 1: Overview of the STEP framework. (a) Given a question and a predefined candidate answer set, (b) the LLM generates a T -step chain-of-thought followed by a predicted answer. (c) For each prefix c_i , a lightweight forward pass yields a soft probability distribution over candidate answers, from which q_i^* is extracted. The resulting trajectory $\{q_1^*, \dots, q_T^*\}$ reflects how confidently the model commits to its predicted answer across reasoning steps. (d) Three soft trajectory features are computed from this trajectory. (e) The resulting correctness score is used to assess the reliability of the model’s prediction.

Probing-based methods. A complementary line of work shows that LLM hidden states encode signals correlated with truthfulness and correctness. Burns et al. (2023) and Marks and Tegmark (2024) show that truthfulness is linearly represented in activation space and can be recovered without supervision, by finding directions that separate true from false statements. While these unsupervised methods require no correctness labels, they still require fitting to the target distribution and have been shown to generalize poorly to structurally different inputs. Supervised probes face the same limitation more acutely: Azaria and Mitchell (2023) train a classifier on hidden layer activations to predict statement truthfulness, and Kossen et al. (2024) train linear probes to approximate semantic entropy from a single forward pass. In both the supervised and unsupervised settings, the reliance on distribution-specific fitting limits applicability when the task domain or model changes, which is a limitation our training-free approach avoids entirely.

Trajectory-based correctness estimation. Several works exploit the structure of the reasoning process rather than the final output alone. Wang et al. (2025) show that the trajectory of hidden

states across layers exhibits measurable differences between correct and incorrect responses: correct trajectories tend to travel greater distances in embedding space and maintain more consistent direction, and their CoE-R and CoE-C scores aggregate these geometric signals as training-free correctness estimates. Bi et al. (2025) model the dynamics of hidden state transitions across reasoning steps from a theoretical perspective, deriving a scalar score that reflects the soundness of the reasoning trajectory. These methods operate entirely in the latent space and require white-box access to specific layer activations.

A related but distinct approach is taken by Zhang et al. (2025), who operate in the output probability space rather than the latent space. In their *CoVo* framework, correct trajectories exhibit high *consistency* (intermediate steps consistently favor the final answer) and low *volatility* (the last deviating step occurs early in the chain). Our method builds on this insight but addresses two key limitations: *CoVo* requires sampling N independent trajectories per prompt, and reduces per-step confidence to hard binary indicators, discarding fine-grained confidence information. We exploit the closed-set property of the task to operate on a single trajectory,

and replace binary indicators with soft probability distributions that preserve the full confidence signal.

Chain-of-thought reasoning and faithfulness. Wei et al. (2022) demonstrate that prompting LLMs to generate intermediate reasoning steps substantially improves accuracy on multi-step tasks. A key assumption underlying trajectory-based correctness estimation, including our own, is that these intermediate steps are meaningful reflections of the model’s reasoning process. Our method exploits this structure by truncating the chain-of-thought at each step and probing the model’s implicit answer preference, treating the reasoning trajectory as a window into the model’s evolving commitment to candidate answers. This is complementary to process reward models (Lightman et al., 2024), which also exploit step-level structure but require annotated training data; our approach is training-free and derives correctness signals directly from the model’s output distribution at each step.

3 Method

3.1 Problem Formulation

We focus on **closed-set reasoning tasks**, in which a model is given an input question x and a predefined set of M candidate answers $\mathcal{A} = \{a_1, a_2, \dots, a_M\}$, and generates a chain-of-thought (CoT) followed by a final answer $\hat{a} \in \mathcal{A}$.

Given the question x , candidate set $\mathcal{A}$, and the generated chain-of-thought $\mathcal{C}$, our goal is to compute a scalar correctness score $\phi(x, \mathcal{C}) \in \mathbb{R}$ that correlates with whether $\hat{a}$ matches the ground-truth answer, without access to labels. We evaluate methods using AUROC, AUPR-C and AUPR-E (see Section 4.1 and Appendix C for formal definitions).

3.2 Per-Step Answer Distribution

Step Segmentation. The model is prompted to generate reasoning in a structured format, with each logical step on a separate line. The resulting chain-of-thought $\mathcal{C}$ is segmented into T discrete steps $\{s_1, s_2, \dots, s_T\}$. The prefix context at step i is defined as

$$c_i = [x, s_1, s_2, \dots, s_i], \quad i = 1, \dots, T.$$

Each prefix c_i represents the model’s reasoning state after i steps: it captures everything the model has considered up to that point, without revealing how the reasoning will continue. By probing the

model at each such prefix, we obtain a snapshot of its implicit answer preference at every stage of the reasoning process, allowing us to track how confidence evolves across the full trajectory rather than relying solely on the final output. Notably, this segmentation requires no additional generation: the steps $\{s_1, \dots, s_T\}$ are extracted directly from the already-generated chain-of-thought, and each prefix c_i is constructed by simple truncation.

Soft Probability Distribution. To probe the model’s implicit preference for each candidate answer at step i , we append a short answer-elicitation prompt to c_i and perform a single forward pass to obtain next-token logits (see Appendix A). For each candidate answer token, the probability is computed as

$$q_i(a_j) = \frac{\exp(z_i^{(j)})}{\sum_{k=1}^M \exp(z_i^{(k)})}, \quad j = 1, \dots, M, \quad (1)$$

where $z_i^{(j)}$ is the logit corresponding to a_j at step i . Our method requires only that each candidate answer can be disambiguated by its first token: in standard multiple-choice settings this is naturally satisfied by option labels (e.g., A, B, C, D), but the approach generalizes to any closed-set task where candidate answers have distinct first tokens, without requiring answers to be single tokens. We denote $q_i^* = q_i(\hat{a})$ as the probability assigned to the final predicted answer at step i .

The sequence $\{q_1^*, q_2^*, \dots, q_T^*\}$ constitutes the **reasoning trajectory** in probability space. This training-free re-scoring of the already-generated chain-of-thought requires only T lightweight forward passes, one per reasoning step, without additional text generation.

3.3 STEP: Soft Trajectory Estimation of Prediction Correctness

We define a family of step-level features that summarize different aspects of the trajectory’s dynamics, each building on the previous by incorporating additional structure. All features aggregate information from the per-step distributions q_i across steps and are normalized by the number of steps T , making them comparable across chains of different lengths.

Soft Consistency. The simplest summary of the trajectory is the mean per-step confidence, which

averages q_i^* uniformly across all T reasoning steps:

$$\text{Con}_{\text{soft}} = \frac{1}{T} \sum_{i=1}^T q_i^*.$$

This feature captures the overall level of commitment to the predicted answer throughout the reasoning process. Unlike hard consistency, which only records whether the predicted answer is the argmax at each step, Con_{soft} preserves the full magnitude of q_i^* , providing a strictly richer signal.

Margin-Based Consistency. Mean confidence alone does not capture how decisively the model distinguishes between competing candidate answers. To capture this decisiveness directly, we measure the gap between the top two probabilities at each step:

$$\text{Con}_{\text{margin}} = \frac{1}{T} \sum_{i=1}^T \sigma \left(\alpha \left(q_i^{(1)} - q_i^{(2)} \right) \right),$$

where $q_i^{(1)} \geq q_i^{(2)}$ are the two highest probabilities at step i , σ is the sigmoid function, and $\alpha > 0$ controls strictness. A large margin indicates the model decisively favors one answer over all others, while a small margin signals genuine uncertainty between competing options. In the following experiment section, we adopt $\alpha = 5.0$.

Temporally-Weighted Consistency. As reasoning progresses, the model gradually eliminates competing options and its answer distribution concentrates toward the final prediction, making later steps more discriminative between correct and incorrect trajectories. We capture this by introducing an exponential weighting scheme that assigns higher weights to later steps:

$$\text{Con}_{\text{temp}} = \frac{\sum_{i=1}^T w_i q_i^*}{\sum_{i=1}^T w_i}, \quad w_i = e^{\beta \cdot \frac{i}{T}}, \quad \beta > 0,$$

where β controls the degree to which later steps are emphasized. As $\beta \rightarrow 0$, Con_{temp} recovers Con_{soft} , making β a natural generalization parameter.

3.4 Relationship to Hard Indicators

The soft features defined above are strictly more informative than the hard binary indicators used in prior trajectory-based methods, as they preserve the full continuous confidence signal at each step rather than reducing it to a binary correct/incorrect label.

The hard consistency score

$$\text{Con}_{\text{hard}} = \frac{1}{T} \sum_{i=1}^T \mathbb{1} \left[\hat{a} = \arg \max_{a_j} q_i(a_j) \right]$$

retains only rank information and ignores the magnitude of the model’s commitment at each step. Its soft counterpart, Con_{soft} , preserves the full probability value q_i^* : two steps with the same argmax but vastly different confidence levels are treated identically by Con_{hard} , yet carry very different information about reasoning reliability. $\text{Con}_{\text{margin}}$ and Con_{temp} further enrich this signal by capturing decisiveness between competing options and temporal structure respectively, aspects that binary discretization fundamentally cannot encode.

Moreover, the closed-set structure of the task is what makes this single-trajectory probing feasible in the first place: because $\mathcal{A}$ is predefined, we can directly extract q_i over a fixed candidate token set without resampling, whereas prior work (Zhang et al., 2025) requires N sampled trajectories to construct a cross-trajectory distance matrix before any binary indicator can be computed. Our framework thus corresponds to a closed-set, single-trajectory simplification of the original CoVo consistency; the full derivation of this relationship is given in Appendix E.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate on four datasets spanning multiple reasoning domains: **MMLU** (Hendrycks et al., 2021), a massive multitask benchmark covering 57 academic subjects; **CommonsenseQA** (Talmor et al., 2019), a commonsense reasoning benchmark with commonsense-related multiple-choice questions; **Belebele** (Bandarkar et al., 2024), a comprehensive reading comprehension benchmark; and **ARC-Challenge** (Clark et al., 2018), a set of grade-school level science questions requiring complex reasoning. Dataset statistics are provided in Appendix D.

Models. We select **Qwen2.5-7B-Instruct** (Yang et al., 2024) and **Meta-Llama-3-8B-Instruct** (Grattafiori et al., 2024), two representative open-source instruction-tuned models at the 7–8B parameter scale. We evaluate on models of this scale as they offer a practical balance between reasoning capability and computational efficiency. All models generate chain-of-thought

Method	MMLU			CommonsenseQA			Belebele			ARC-Challenge		
	AUROC	AUPR-C	AUPR-E	AUROC	AUPR-C	AUPR-E	AUROC	AUPR-C	AUPR-E	AUROC	AUPR-C	AUPR-E
QWEN2.5-7B-INSTRUCT												
<i>Final-output statistics</i>												
Entropy	54.00	72.44	32.43	56.96	83.95	23.42	55.39	91.04	12.04	58.81	90.05	16.64
Perplexity	53.39	72.19	31.91	56.09	83.41	22.74	54.94	90.06	11.80	58.44	90.20	15.84
Max Prob.	53.33	72.11	31.93	56.07	83.35	22.92	54.69	90.88	11.76	58.62	90.27	16.03
<i>Latent-trajectory</i>												
CoE-R	64.37	80.58	41.43	55.50	82.37	23.89	55.27	90.87	14.76	57.32	89.05	16.20
CoE-C	64.87	80.52	41.73	56.07	82.00	26.05	55.83	90.81	14.29	62.17	90.59	20.74
CoT-Kinetics	59.99	75.79	37.44	53.49	81.59	22.68	54.71	90.33	12.21	60.14	90.34	17.22
<i>Hard token-trajectory</i>												
CoVo-Con	70.58	86.49	53.85	73.01	92.25	45.43	78.72	97.23	49.76	64.84	94.91	34.55
CoVo-Vol	70.45	86.44	54.00	73.00	92.23	46.83	77.95	97.15	44.31	64.92	94.92	<u>36.23</u>
CoVo-Linear	70.73	86.52	54.32	73.27	<u>92.29</u>	46.10	78.85	97.21	<u>48.45</u>	64.98	94.93	35.66
<i>STEP (ours)</i>												
CON _{soft}	<u>75.23</u>	<u>86.91</u>	<u>55.68</u>	<u>76.06</u>	92.25	46.44	<u>83.60</u>	<u>97.40</u>	46.94	<u>77.97</u>	95.56	35.48
CON _{margin}	74.22	86.48	54.90	75.56	91.98	45.95	82.87	97.30	46.81	77.66	<u>95.53</u>	35.46
CON _{temp}	75.92	87.24	56.82	76.86	92.50	<u>46.76</u>	84.16	97.45	46.54	78.38	95.49	37.01
META-LLAMA-3-8B-INSTRUCT												
<i>Final-output statistics</i>												
Entropy	49.84	59.63	40.16	48.90	72.53	25.82	53.61	90.93	11.23	48.51	78.53	20.40
Perplexity	49.49	59.40	39.74	49.12	72.78	26.08	53.65	90.78	11.18	48.67	78.35	20.25
Max Prob.	49.43	59.34	39.70	48.94	72.68	26.07	53.76	90.74	11.16	48.28	78.27	19.99
<i>Latent-trajectory</i>												
CoE-R	59.21	68.68	48.05	54.21	75.15	29.38	59.69	91.81	14.32	56.52	82.91	26.43
CoE-C	53.56	64.01	42.17	51.19	74.18	26.69	52.93	89.93	11.92	56.05	82.76	25.38
CoT-Kinetics	57.35	67.32	44.92	52.85	74.86	28.26	54.25	89.76	12.52	57.76	83.73	26.24
<i>Hard token-trajectory</i>												
CoVo-Con	77.54	<u>86.34</u>	74.12	77.29	91.55	66.29	87.33	98.39	68.95	84.52	95.65	72.95
CoVo-Vol	76.91	86.12	73.74	75.86	91.11	64.91	86.86	98.34	67.20	84.29	95.61	73.26
CoVo-Linear	77.52	86.32	74.11	77.27	<u>91.54</u>	66.00	87.29	98.38	<u>68.80</u>	84.60	95.67	73.32
<i>STEP (ours)</i>												
CON _{soft}	<u>82.11</u>	86.28	<u>74.73</u>	<u>81.24</u>	90.93	66.82	<u>91.78</u>	<u>98.62</u>	67.76	<u>88.83</u>	<u>96.15</u>	<u>74.94</u>
CON _{margin}	81.75	86.07	73.96	81.03	90.93	<u>67.01</u>	91.56	98.60	64.04	88.51	96.06	73.47
CON _{temp}	82.29	86.47	74.84	81.70	91.20	67.38	91.89	98.64	67.28	89.24	96.24	76.09

Table 1: AUROC, AUPR-C, and AUPR-E on LLM experiments across four benchmarks and two models. **Bold**: best result per column within each model block. Underline: second best result per column within each model block.

reasoning by prompting them to explicitly articulate intermediate reasoning steps before producing a final answer, following the prompt template described in Appendix B. For each query we generate *one* fixed chain-of-thought rollout per model and reuse it for every method compared, so differences across methods reflect only the scoring procedure, not generation variance.

Task Accuracy. Table 2 reports task accuracy for the same generations used throughout. Accuracy ranges 61.13–89.89%, confirming the task is meaningfully imbalanced (Section 4.1) rather than near-ceiling or near-chance. At most 0.85% of gen-

Dataset	Qwen2.5-7B Acc. (%)	Llama-3-8B Acc. (%)
MMLU	70.60	61.13
CommonsenseQA	80.67	74.42
Belebele	89.89	89.42
ARC-Challenge	87.63	79.20

Table 2: Task accuracy per benchmark and model.

erations lack a parsable answer and are excluded.

Computational Cost. STEP’s only overhead versus the free final-output-statistics baselines is T lightweight forward passes per query, no new to-

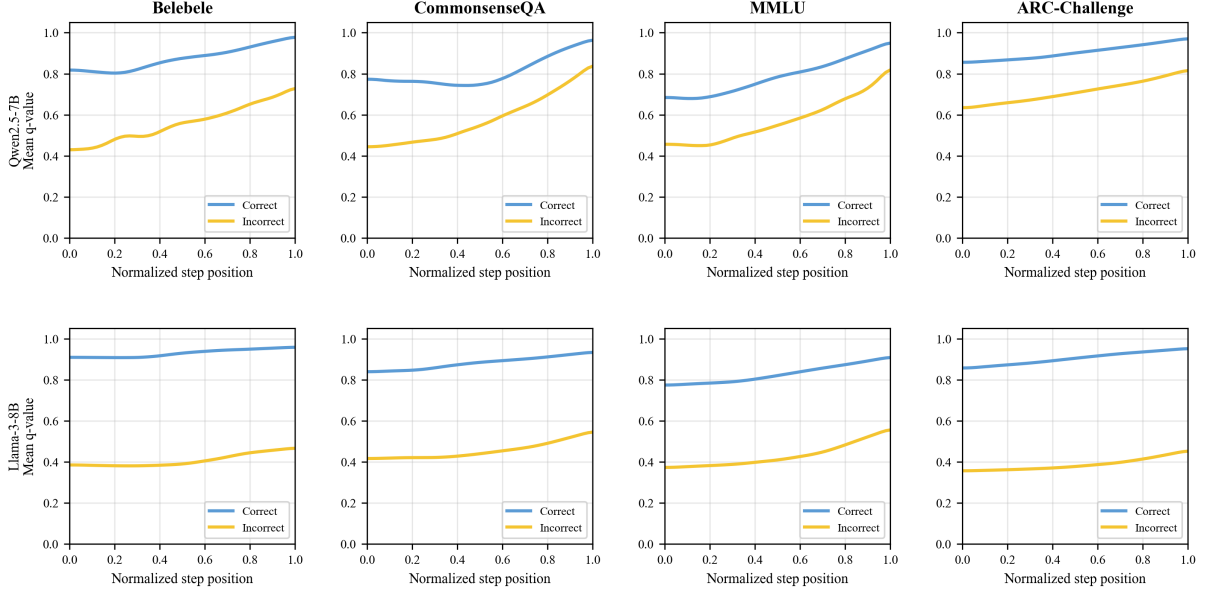


Figure 2: Mean q^* trajectory across normalized reasoning steps for correct (blue) and incorrect (orange) predictions, across four datasets and two models. Correct trajectories consistently maintain higher confidence and converge toward near-certainty by the final step, while incorrect trajectories plateau at a lower level. The widening gap toward later steps motivates the temporal weighting in Con_{temp} .

kens generated (Appendix A); $\bar{T}$ remains small across all datasets (typically in the single digits). This matches the hard token-trajectory baselines (CoVo-Con/Vol/Linear), which use the same q_i , so the soft-vs-hard gain (Section 4.2) is free. STEP’s real advantage is over methods needing N sampled full CoT generations, e.g. original CoVo (Zhang et al., 2025) (Appendix E) and semantic uncertainty methods (Kuhn et al., 2023; Farquhar et al., 2024): $O(N\bar{T})$ generated tokens vs. STEP’s $O(T)$ forward passes with none. More information is in Appendix F.

Baselines. We compare against three categories of baselines. (1) *Final-output statistics*: **Entropy**, token-level perplexity (**Perplexity**), and maximum softmax probability (**MaxProb**), which aggregate token-level statistics over the entire generated sequence and discard the reasoning structure entirely. (2) *Latent-trajectory methods*: **CoE-R** and **CoE-C** (Wang et al., 2025), which derive correctness scores from the geometric trajectory of hidden states across layers, and **CoT-Kinetics** (Bi et al., 2025), which models the dynamics of hidden state transitions across reasoning steps. (3) *Hard token-trajectory methods*: **CoVo-Con**, **CoVo-Vol**, and **CoVo-Linear**, which are adaptations of the consistency and volatility concepts introduced by Zhang et al. (2025) to our single-trajectory setting. These variants derive confidence from per-step binary cor-

rectness indicators rather than soft probabilities, discarding fine-grained confidence information.

Evaluation Metrics. To better evaluate the quality of different scoring methods, we report AUROC (Fawcett, 2006), AUPR-C and AUPR-E (Boyd et al., 2013) as evaluation metrics, following established protocols for misclassification detection (Hendrycks and Gimpel, 2017). AUROC (Area Under the Receiver Operating Characteristic curve) measures the overall discriminative ability between correct and incorrect predictions. AUPR-C (Area Under the Precision-Recall curve for the *Correct* class) and AUPR-E (Area Under the Precision-Recall curve for the *Error* class) treat correct and incorrect predictions as the positive class, respectively. Since LLM accuracy on our benchmarks ranges from 60% to 90%, the class distribution is heavily skewed toward correct predictions, making AUROC alone insufficient (Corbière et al., 2019). AUPR-C reflects the ability to confirm high-confidence correct predictions, while AUPR-E captures the more practically important ability to identify errors among the minority class. Formal definitions are provided in Appendix C.

4.2 Main Results

Table 1 presents AUROC, AUPR-C, and AUPR-E of all methods on four benchmarks and two models.

STEP consistently outperforms all baselines.

Across all datasets and both models, soft trajectory features achieve the best overall performance, surpassing output-statistics-based, latent-trajectory, and hard token-trajectory baselines in the vast majority of settings. Although individual metrics occasionally favor a specific baseline on a particular dataset, the improvements are consistent across models and tasks, validating that preserving continuous probability signals across reasoning steps provides richer information for correctness estimation than either final-output statistics or hard trajectory indicators.

Weighting favoring later steps further boosts performance.

Among the three variants of STEP, *Temporally-Weighted Consistency* Con_{temp} achieves the best evaluation metrics in nearly all settings, outperforming the simpler *Soft Consistency* Con_{soft} and *Margin Consistency* $\text{Con}_{\text{margin}}$; we recommend it as the single deployed estimator, reporting the other two as ablations rather than alternatives to combine at inference time. This suggests that not all reasoning steps contribute equally to correctness estimation: later steps, where the model’s answer distribution has begun to stabilize, carry more predictive signal. Con_{soft} and $\text{Con}_{\text{margin}}$ still substantially outperform hard token-trajectory methods despite treating all steps uniformly, confirming the gain from soft signals is orthogonal to the aggregation strategy.

4.3 Trajectory Analysis

Figure 2 visualizes the mean q^* trajectory across reasoning steps for correct and incorrect predictions, averaged over all samples in each dataset. Correct trajectories maintain substantially higher q^* values throughout the reasoning process, confirming that the model’s commitment to its final answer is a reliable signal of correctness. This pattern is consistent across all four datasets and both model families, suggesting it reflects a general property of LLM reasoning rather than a dataset- or model-specific artifact. The base variant of STEP, *Soft Consistency*, which simply averages q^* throughout the chain-of-thought, is therefore shown to be effective.

4.4 Ablation Study

Figure 3 illustrates how the temporal weighting parameter β affects AUROC across datasets on model Meta-Llama-3-8B-Instruct. Recall that $\beta >$

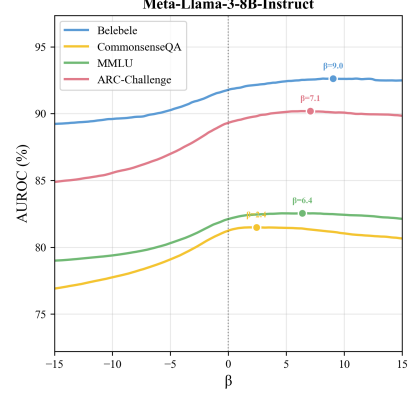


Figure 3: AUROC as a function of the temporal weighting parameter β across datasets on Meta-Llama-3-8B-Instruct. $\beta > 0$ up-weights later reasoning steps; $\beta = 0$ recovers Con_{soft} .

0 assigns higher weights to later reasoning steps, $\beta < 0$ to earlier steps, and $\beta = 0$ recovers Con_{soft} .

Across all datasets, performance increases monotonically as β grows from negative to positive values. We highlight three observations:

Later steps are more informative. Performance improves consistently as β increases from zero, confirming that reasoning steps closer to the final answer carry stronger correctness signal. This aligns with the intuition that early steps often consist of problem restatement or initial exploration, where the model has not yet committed to any reasoning direction. As reasoning progresses, the model gradually eliminates competing options and its answer distribution concentrates toward the final prediction, producing the largest divergence between correct and incorrect trajectories. Temporal weighting respects this process by giving the model room to explore early on while placing greater emphasis on the steps where meaningful commitment emerges.

Performance saturates at large β . Gains of AUROC plateau around $\beta \approx 5$ to 10 and remain stable thereafter. We attribute this to a ceiling effect at the very final steps: regardless of whether the prediction is correct or incorrect, the model tends to assign near-certain probability to its predicted answer as it self-consistently converges toward its own conclusion, reducing the discriminative value of the last steps.

Negative β consistently hurts performance. To further verify that later steps are more informative than earlier ones, we also explore negative values of β , which up-weight early steps instead. Performance degrades consistently across all datasets,

confirming our hypothesis: early steps, where the model has not yet begun to commit to any candidate and the answer distribution remains noisy, carry weaker signal than the later steps where meaningful reasoning commitment emerges.

5 Conclusion

We presented STEP, a training-free framework for correctness estimation in closed-set reasoning tasks. By performing lightweight forward passes on an already-generated chain-of-thought, STEP derives soft trajectory features from per-step probability distributions over the candidate answer set. Across multiple benchmarks and two models, all three instantiations of STEP consistently outperform existing baselines in AUROC while achieving competitive or improved AUPR performance. Among them, *Temporally-Weighted Consistency* achieves the strongest performance, confirming that later reasoning steps carry disproportionately more predictive signal. These results validate the core premise of STEP: preserving continuous probability signals across reasoning steps provides richer information than hard binary indicators, at no additional generation cost. Future work may explore extending soft trajectory features to open-ended settings by constructing candidate sets dynamically from the model’s top- k predictions at each step.

Limitations

Scope of applicability. STEP is designed for closed-set tasks with a predefined candidate answer set, and is not directly applicable to open-ended generation. It further requires that candidate answers be disambiguable by their first token; answers sharing a common prefix would require additional aggregation strategies.

Dependence on reasoning structure. STEP relies on structured chain-of-thought reasoning to construct per-step prefixes. Models producing unstructured or highly variable reasoning, as well as formal symbolic tasks where correctness depends on exact computation rather than progressive semantic narrowing, may not benefit consistently.

Access requirements. STEP requires access to next-token logits, available for open-source models but not for black-box APIs.

References

- Amos Azaria and Tom Mitchell. 2023. The internal state of an LLM knows when it’s lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976.
- Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2024. The Belebele benchmark: a parallel reading comprehension dataset in 122 language variants. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775, Bangkok, Thailand. Association for Computational Linguistics.
- Jinhe Bi, Danqi Yan, Yifan Wang, Wenke Huang, Haokun Chen, Guancheng Wan, Mang Ye, Xun Xiao, Hinrich Schütze, Volker Tresp, and Yunpu Ma. 2025. CoT-Kinetics: A theoretical modeling assessing LRM reasoning process. *arXiv preprint arXiv:2505.13408*.
- Kendrick Boyd, Kevin H. Eng, and C. David Page. 2013. Area under the precision-recall curve: Point estimates and confidence intervals. In *Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, pages 451–466. Springer.
- Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2023. Discovering latent knowledge in language models without supervision. In *International Conference on Learning Representations*.
- Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? Try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.
- Charles Corbière, Nicolas Thome, Avner Bar-Hen, Matthieu Cord, and Patrick Pérez. 2019. Addressing failure prediction by learning model confidence. In *Advances in Neural Information Processing Systems*, volume 32, pages 2902–2913.
- Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.
- Tom Fawcett. 2006. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, and 1 others. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *International Conference on Machine Learning*.

- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.
- Dan Hendrycks and Kevin Gimpel. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*.
- Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.
- Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. 2024. Semantic entropy probes: Robust and cheap hallucination detection in LLMs. *arXiv preprint arXiv:2406.15927*.
- Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *International Conference on Learning Representations*.
- Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let’s verify step by step. In *International Conference on Learning Representations*.
- Andrey Malinin and Mark Gales. 2021. Uncertainty estimation in autoregressive structured prediction. In *International Conference on Learning Representations*.
- Samuel Marks and Max Tegmark. 2024. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. In *Conference on Language Modeling*.
- Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.
- Yiming Wang, Pei Zhang, Baosong Yang, Derek F. Wong, Zhuosheng Zhang, and Rui Wang. 2025. Latent space chain-of-embedding enables output-free LLM self-evaluation. In *International Conference on Learning Representations*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837.
- An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.
- Kongcheng Zhang, Qi Yao, Shunyu Liu, Yingjie Wang, Baisheng Lai, Jieping Ye, Mingli Song, and Dacheng Tao. 2025. Consistent paths lead to truth: Self-rewarding reinforcement learning for LLM reasoning. *arXiv preprint arXiv:2506.08745*.

A Answer-Elicitation Prompt

To compute the per-step answer distribution q_i (Equation 1), we append the following prompt to the truncated context c_i before performing the forward pass:

Final Answer:

This prompt is designed to elicit the model’s implicit preference over candidate answers at the current reasoning state, without generating any additional reasoning content. By appending it immediately after the truncated chain-of-thought, we place the model in a position where its next-token distribution directly reflects its answer preference given everything it has reasoned so far. The logits over the candidate answer tokens are extracted at this position and normalized via softmax over the candidate set to obtain q_i , as described in Equation 1. No further tokens are generated beyond this single forward pass.

B Chain-of-Thought Generation Prompt

All models are prompted with the following template to elicit structured chain-of-thought reasoning. The placeholder `{input_data}` is replaced with the dataset-specific question and answer choices. The candidate set is $\{A, B, C, D, E\}$ for CommonsenseQA and $\{A, B, C, D\}$ for all other datasets.

```

Answer the following multiple choice question.

Explain your reasoning in a few statements.
Each statement should represent one small
logical improvement over the previous
statements.
Write each statement on its own line.
Do not number or label the statements.

The final line must be:
Answer: X

Where X MUST BE one of A, B, C, D [, or E].
Do not write anything after the answer.

{input_data}

```

The prompt deliberately avoids numbered lists or structured headers to produce free-form reasoning steps that can be cleanly segmented by newline splitting. Each line of the response (excluding the final Answer: line) constitutes one reasoning step s_k , yielding a trajectory of length T for confidence scoring.

C Evaluation Metrics

AUROC. The Area Under the Receiver Operating Characteristic Curve measures the probability that a randomly chosen correct prediction receives a higher correctness score than a randomly chosen incorrect prediction:

$$\text{AUROC} = \mathbb{P}(\phi(x^+) > \phi(x^-)) \quad (2)$$

where x^+ and x^- are randomly drawn correct and incorrect samples respectively, and $\phi(\cdot)$ denotes the correctness score assigned by the method. A value of 1.0 indicates perfect separation; 0.5 corresponds to random performance.

AUPR-C. The Area Under the Precision-Recall Curve with correct predictions as the positive class summarizes the trade-off between precision and recall across all decision thresholds:

$$\text{AUPR-C} = \int_0^1 P_C(r) dr \quad (3)$$

where $P_C(r)$ is the precision at recall level r when treating correct predictions as positives. AUPR-C reflects the ability to confirm high-confidence correct predictions, and serves as a natural complement to AUROC under class imbalance.

AUPR-E. The Area Under the Precision-Recall Curve with incorrect predictions as the positive class measures the ability to detect errors among the minority class:

$$\text{AUPR-E} = \int_0^1 P_E(r) dr \quad (4)$$

where $P_E(r)$ is the precision at recall level r when treating incorrect predictions as positives. Since LLM accuracy on our benchmarks ranges from 60% to 90%, incorrect predictions constitute a small minority, making AUPR-E particularly informative for evaluating error detection capability in practical deployment scenarios.

D Dataset Statistics

We evaluate on four multiple-choice question answering datasets spanning diverse reasoning domains. MMLU (Hendrycks et al., 2021) covers 57 academic subjects ranging from STEM to humanities. CommonsenseQA (Talmor et al., 2019) requires everyday commonsense knowledge to disambiguate between plausible answer choices. Belebele (Bandarkar et al., 2024) tests reading comprehension across typologically diverse languages; we use the English subset. ARC-Challenge (Clark et al., 2018) consists of science questions specifically selected to be resistant to simple retrieval-based and word co-occurrence methods. Table 3 summarizes the key statistics.

Dataset	#Samples	#Choices
MMLU	1,531	4
CommonsenseQA	2,000	5
Belebele	900	4
ARC-Challenge	1,172	4

Table 3: Dataset statistics used in our evaluation.

E Relationship to CoVo Consistency

The original CoVo consistency (Zhang et al., 2025) is defined over N sampled trajectories with K distinct answers via a distance matrix $\mathbf{D} \in \mathbb{R}^{T \times K}$, where

$$\begin{aligned} \mathbf{D}[i, k] &= d(s_i, y_k) \\ &= -\frac{1}{|y_k|} \sum_{j=1}^{|y_k|} \log \pi_\theta(y_k[j] \mid s_i, y_k[:j]) \end{aligned}$$

is the negative log-likelihood of answer y_k at reasoning state s_i . Consistency is then defined as the fraction of intermediate states whose own final answer is closer than any alternative:

$$\text{Con}_{\text{CoVo}}(\tau) = \frac{1}{T} \sum_{i=0}^{T-1} \mathbb{1} \left[\mathbf{D}[i, 0] = \min_{0 \leq k < K} \mathbf{D}[i, k] \right] \quad (5)$$

This formulation requires N sampled trajectories to populate K distinct candidate answers, since the alternative answers $\{y_1, \dots, y_{K-1}\}$ are obtained from other sampled trajectories rather than from a predefined set.

In the closed-set setting, $\mathcal{A} = \{a_1, \dots, a_M\}$ is predefined and fixed, which admits two simplifications. First, the candidate set is known without re-sampling, so the distance matrix can be replaced by

the per-step softmax distribution q_i defined in Equation 1, where $q_i(a_j) \propto \exp(z_i^{(j)})$ directly reflects the model’s relative preference over all candidates at step i . Second, since q_i is already normalized over $\mathcal{A}$, the distance-based argmax comparison reduces to:

$$\begin{aligned} \mathbf{D}[i, 0] &= \min_{0 \leq k < K} \mathbf{D}[i, k] \\ \iff \hat{a} &= \arg \max_{a_j} q_i(a_j) \end{aligned}$$

Applying these substitutions to Con_{CoVo} and dropping the group-level summation over G trajectories (as only a single trajectory is available) recovers exactly Con_{hard} as defined in Section 3:

$$\text{Con}_{\text{hard}} = \frac{1}{T} \sum_{i=1}^T \mathbb{1} \left[\hat{a} = \arg \max_{a_j} q_i(a_j) \right]$$

where we reindex intermediate states from $i = 1$ to T to match the notation used throughout this paper; the original formulation uses $i = 0$ to $T - 1$. Con_{hard} thus inherits the binary structure of Con_{CoVo} while requiring only a single forward pass per reasoning step. The soft features introduced in Section 3 replace this binary indicator with the continuous value q_i^* , preserving the full confidence signal that the hard indicator discards.

F Runtime and Hardware

All experiments were run on a single NVIDIA A100 GPU, with no model training involved. Total inference time per dataset, including the initial chain-of-thought generation and all downstream scoring (final-output statistics, latent-trajectory, hard token-trajectory, and STEP), ranges from 20 minutes to approximately 3–4 hours depending on dataset size and model.

Each of STEP’s additional forward passes (Section 4.1) processes a fixed truncated prefix and produces no additional output tokens. Unlike sampling-based methods, STEP therefore avoids autoregressive generation of additional full reasoning trajectories.