Benchmarking Uncertainty Metrics for LLM Target-Aware Search

Pei-Fu Guo and **Yun-Da Tsai** and **Shou-De Lin**
National Taiwan University
r12922217@csie.ntu.edu.tw
f08946007@csie.ntu.edu.tw
sdlin@csie.ntu.edu.tw

Abstract

LLM search methods, such as Chain of Thought (CoT) and Tree of Thought (ToT), enhance LLM reasoning by exploring multiple reasoning paths. When combined with search algorithms like MCTS and Bandit methods, their effectiveness relies heavily on uncertainty estimation to prioritize paths that align with specific search objectives. *However, it remains unclear whether existing LLM uncertainty metrics adequately capture the diverse types of uncertainty required to guide different search objectives.* In this work, we introduce a framework for uncertainty benchmarking, identifying four distinct uncertainty types: Answer, Correctness, Aleatoric, and Epistemic Uncertainty. Each type serves different optimization goals in search. Our experiments demonstrate that current metrics often align with only a subset of these uncertainty types, limiting their effectiveness for objective-aligned search in some cases. These findings highlight the need for additional target-aware uncertainty estimators that can adapt to various optimization goals in LLM search. Code is available at link.

1 Introduction

LLM search methods, such as Chain of Thought (CoT)(Wei et al., 2022), Tree of Thought (ToT)(Yao et al., 2024), and ReAct (Yao et al., 2022), have significantly improved the performance of large language models (LLMs) by generating multiple intermediate reasoning steps. These methods can be further enhanced through algorithmic search, leveraging techniques like Monte Carlo Tree Search (e.g., LATS, STaR)(Zhou et al., 2023; Zelikman et al., 2022), bandit algorithms (e.g., LongPo)(Hsieh et al., 2023), or gradient-style optimization (e.g., OPRO) (Yang et al., 2023). A key component in these approaches is uncertainty estimation, which plays a crucial role in prioritizing reasoning paths that are aligned with specific search targets, such as maximizing correctness, reducing noise, or improving diversity.

Prior work on uncertainty estimation in LLMs has largely focused on metrics derived from generation likelihoods (Chen et al., 1998; Malinin and Gales, 2020; Kuhn et al., 2023), verbalized model confidence (Kadavath et al., 2022; Lin et al., 2022a), consistency across outputs (Xiong et al., 2023; Jiang et al., 2023b) or model knowledge (Ahdritz et al., 2024). While useful for calibration and risk assessment, *it remains unclear whether these metrics are well-suited for search/optimization tasks*, where uncertainty should help guide decision-making based on the targeted search objectives. For instance, search methods that optimize for factual accuracy need uncertainty signals that reflect model correctness, whereas strategies that encourage idea diversity need metrics sensitive to answer variability.

To address this gap, we first identify four key types of uncertainty that are relevant to target-aware search: *Answer Uncertainty* (variability in output answers), *Correctness Uncertainty* (uncertainty about the factual validity of an answer), *Aleatoric Uncertainty* (intrinsic ambiguity in the query input), *Epistemic Uncertainty* (model uncertainty due to limited knowledge).

We then present a unified benchmarking framework to evaluate how well existing uncertainty metrics approximate these uncertainty types in structured search scenarios. Our pipeline constructs large tree-structured reasoning traces through extensive sampling and Monte Carlo approximations to estimate accurate target uncertainties. Through experiments on multiple datasets and LLMs, we show that *current metrics often align with only a subset of uncertainty types, limiting their effectiveness for objective-aligned search in some cases.* These findings highlight the need for additional search-aware uncertainty estimators that can adapt to various optimization goals in LLM search.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 4230–4238
November 4-9, 2025 ©2025 Association for Computational Linguistics

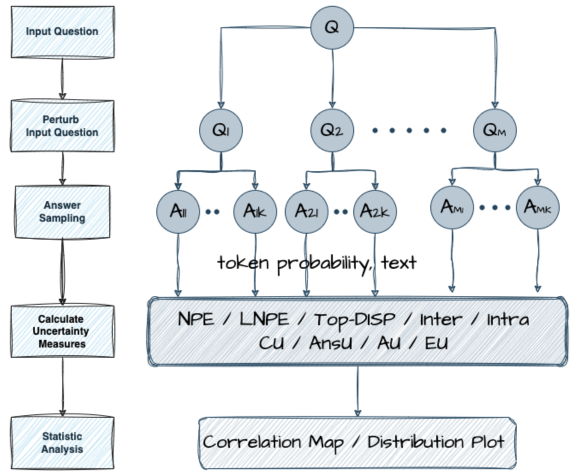

Figure 1: **Benchmarking Pipeline Overview.** For each question, our pipeline constructs a reasoning tree, generating numerous metric estimates and truth value pairs. We then apply bootstrapping on these pairs to perform robust statistical analysis. Pseudo code at Appendix D

2 Different Uncertainty Types for Search

Uncertainty can arise in various forms, each representing different aspects of the problem and guiding the search process in unique ways.

Answer Uncertainty (AnsU) reflects the model's confidence and the diversity of possible answers. AnsU describes how consistently the model produces the same answer after repeated sampling, but it does not guarantee correctness. If the model lacks necessary knowledge to answer the given question, it is reasonable for the output answers to be incorrect, even if low AnsU is observed after repeated sampling. AnsU is measured as the entropy of the output answer distribution. $AnsU(x) = -\sum_i p(y_i|x) \log p(y_i|x)$ where $p(y_i|x)$ is the probability of output answer y_i obtain from $\{\mathbf{x}_j\}$ given the input x. AnsU can guide search algorithms to explore a richer solution space, increasing the variability of generated content. This can be particularly beneficial for tasks where diversity are highly valued.

Correctness Uncertainty (CU) provides insights into the likelihood of answer correctness. High CU indicates that the model's prediction may be unreliable, suggesting the need for additional verification. CU is directly related to the accuracy of the prediction. When CU is low, the model's predictions are less likely to include a majority of false positives. CU is calculated as the entropy of the output correctness distribution. $CU(x) = -\sum_i p(c_i|x) \log p(c_i|x)$ where $p(c_i|x)$ represents the probability of correctness c_i (whether an answer is correct or incorrect) obtain from $\{\mathbf{x}_j\}$ given the input x. CU can guide search by narrow down the solution space and acquire the correct answer, reducing the likelihood of incorrect or irrelevant answers.

Aleatoric Uncertainty (AU) and Epistemic Uncertainty (EU) are two distinct sources of model uncertainty. AU arises from the inherent noise in the data itself. EU, on the other hand, originates from the model's limitations and is related to its knowledge and understanding. AU and EU are calculated using the Deep-Ensemble-Decomposition method (Balaji Lakshminarayanan, 2016), where total model uncertainty is the sum of AU and EU. In our context, θ, which represents model parameters in the original paper, corresponds to the rephrased questions in our setting. EU captures the disagreement between different perturbations, measured by the mutual information $I(Y;\theta|X)$. AU reflects the inherent data noise and is represented as $\mathbb{E}_{q(\theta|D)}[H(q(Y|X,\theta))]$, where the expectation is taken over the questions θ. The total model uncertainty is expressed as :

$$H(q(Y|X)) = \underbrace{I(Y;\theta|X)}_{\text{EU}} + \underbrace{\mathbb{E}_{q(\theta|D)}[H(q(Y|X,\theta))]}_{\text{AU}} \quad (1)$$

AU and EU help identify whether uncertainty stems from data noise or model limitations, enabling targeted improvements like rephrasing questions or adding few-shot examples.

3 Benchmarking Pipeline

A reliable uncertainty metric should serve as an accurate estimator of its target uncertainty. In this study, we introduce a novel benchmarking pipeline designed to assess the effectiveness of uncertainty metrics in estimating target uncertainties within search (Figure 1).

3.1 Pipeline Workflow

To assess how well a metric quantifies the target uncertainty in a search task, we first establish an accurate estimation of the uncertainty values as ground truth in each reasoning step. This requires generating large tree-structured reasoning traces by rephrasing input prompts and sampling output multiple times with varying temperatures. These traces serve two purposes: (1) tree-structured reasoning traces represent a predominant approach in current sampling-based search methods, making them relevant for real-world evaluation, and (2) they thoroughly explore the solution space, enabling robust uncertainty estimation via Monte Carlo methods.

Once ground truth values are obtained, we compute metric estimates at each reasoning node and evaluate their alignment using statistical analysis.

3.2 Rationale for Target Uncertainties

We define target uncertainties according to specific objectives—such as maximizing correctness, promoting diversity, or reducing ambiguity—to capture fine-grained reasoning properties that can vary even within a single task. For example, when a model summarizes a news article, one objective may prioritize accuracy, ensuring all facts are correct, while another may prioritize diversity, generating multiple distinct summaries. This approach allows our benchmark to capture uncertainty signals that are goal-sensitive but not tied to any particular task, making it broadly applicable across tasks with diverse solution criteria. By avoiding reliance on task-specific assumptions, our benchmark evaluates how well uncertainty metrics generalize under flexible, evolving search objectives, even when predefined task labels are unavailable or insufficient.

3.3 Pipeline Scalability

In our pipeline, MC sampling is used to estimate the ground truth uncertainty. Hence, a large number of samples is necessary to ensure that the estimated value is accurate. However, this is a one-time cost. Once we identify the best metrics for specific tasks, they can be used in downstream applications with minimal sampling(based on metric design). Additionally, during sampling, we can store the generated logits and text, allowing reuse of sampling trees for any metric that operates on these outputs. Moreover, MC sampling can be limited to a representative subset of questions (e.g., based on difficulty in the MATH dataset), further reducing cost. These tricks make our approach efficient and scalable for future metric evaluations.

4 Experiments

In this section, we use our benchmarking pipeline to evaluate six existing uncertainty metrics against the target uncertainty defined in Section 2. More implementation details and prompt templates are shown in Appendix A and B.

LLM Uncertainty Metrics We benchmark six *existing task-agnostic uncertainty metrics*, which can be categorized into three types. Token-liklihood based metrics estimate uncertainty from the predictive entropy of model's output distribution, including NORMALIZED PREDICTIVE ENTROPY (NPE), LENGTH-NORMALIZED PREDICTIVE ENTROPY (LNPE) (Malinin and Gales, 2020), and SEMANTIC ENTROPY (SE) (Kuhn et al., 2023). Verbalized-based measures assess confidence by directly prompting the model to express its certainty, such as VERBALIZED CONFIDENCE (VC) and P(TRUE) (Kadavath et al., 2022). Lexical-based metrics, such as LEXICAL SIMILARITY(Lin et al., 2022b), evaluate uncertainty based on the consistency of lexical overlap between multiple responses. Since P(TRUE), and VC are originally formulated to express model confidence, we transform them into uncertainty measures by taking their complements: specifically, PTRUE-COMP, and VC-NEG represent the complement of P(True) and the negative of VC.

While some uncertainty metrics are specifically tuned to particular tasks or datasets, we focus on task-agnostic metrics because they do not require task-specific supervision, retraining, or labeled validation data. This makes them especially valuable in real-world scenarios—such as open-ended generation, under-specified tasks, or low-resource settings—where task formats or ground-truth labels may be unavailable or unreliable.

Datasets and LLMs We select four diverse datasets: MATH (mathematical reasoning) (Hendrycks et al., 2021), COMMONSENSEQA (commonsense reasoning) (Talmor et al., 2018), TRIVIAQA (reading comprehension) (Joshi et al., 2017), and TRUTHFULQA (truthfulness) (Lin et al., 2021). MATH and TRIVIAQA feature open-ended answers, while COMMONSENSEQA and TRUTHFULQA use multiple-choice formats. These datasets are evaluated using three open-source language models (Grattafiori et al., 2024; Team et al., 2024; Jiang et al., 2023a) of similar sizes: LLAMA-3-8B, GEMMA-2-9B, and MISTRAL-7B-V0.3. This setup ensures diversity across task types and model architectures, enhancing the robustness of our evaluation.

5 Results and Analysis

Metric Correlation with Uncertainty Types Figure 2 shows the rank correlation between benchmarked uncertainty metrics and the breakdown into different uncertainty types. As shown, most metrics correlate more strongly with AnsU, AU, and EU than with CU across all datasets. This suggests

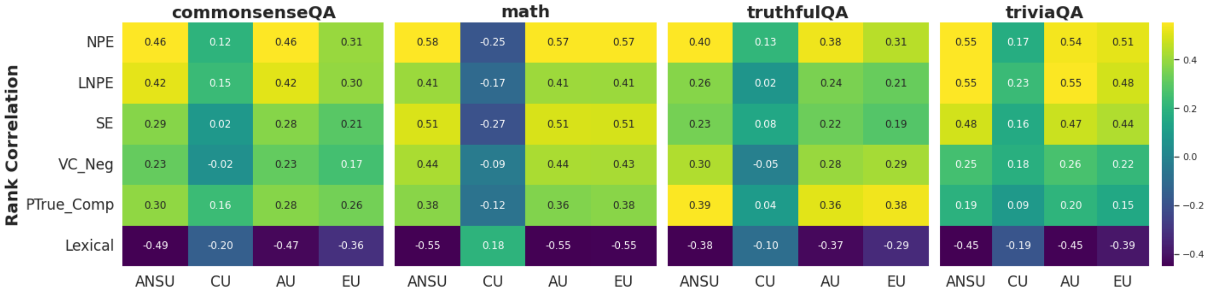

Figure 2: **Dependencies between Metric and Target Uncertainty.** Figure shows the Spearman correlation between benchmarked uncertainty metrics and target uncertainty. A higher correlation indicates that the metric better captures the ground truth uncertainty rankings, making it more reliable for guiding uncertainty-aware decisions. Each value in figure is averaged across models.

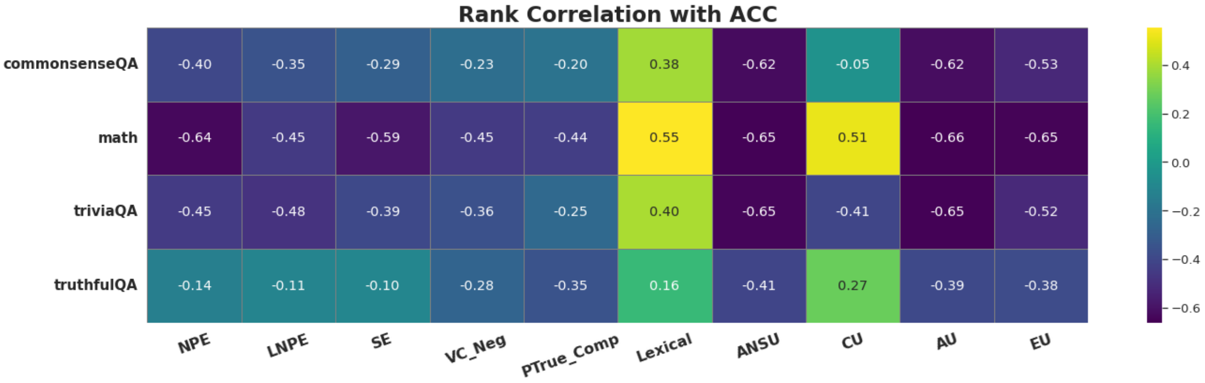

Figure 3: **Correlation with Correctness.** Figure shows the Spearman correlation between benchmarked uncertainty metrics and dataset-level accuracy. A lower correlation suggests that the metric is more aligned with actual model performance across datasets (high uncertainty low correctness). Each value in figure is averaged across models.

that existing metrics are better at capturing answer diversity, data ambiguity, and model disagreement, but struggle to reflect and guide output correctness. This limitation likely stems from their reliance on model output distributions, which are shaped more by confidence and variability than factual accuracy.

To dive deeper, generation-likelihood-based metrics (NPE, LNPE, SE) show strong correlation with AnsU, AU, and EU. In contrast, verbalized-based metrics (VC-NEG, PTrue-Comp) correlate less consistently, possibly due to noise introduced by additional language model inference. Lexical metrics show weak or negative correlation with all targets, indicating that surface-level variation alone doesn't capture meaningful uncertainty. We also present the mutual information results in Appendix C, which show similar patterns.

Interdependencies Among Uncertainty Types From Figure 2, we observe that CU exhibits weaker correlation with the other uncertainty types, underscoring its distinct focus on answer correctness rather than output consensus or variability. In contrast, AnsU, AU, and EU show similar correlation patterns, suggesting a close relationship: ambiguous or under-specified queries (high AU) may increase epistemic uncertainty (high EU), which can lead to more diverse or inconsistent outputs (high AnsU).

Correlation with Correctness Figure 3 presents the correlation between uncertainty metrics and dataset accuracy. We observe that most metrics show some degree of alignment with accuracy. Generation-likelihood-based metrics (e.g., NPE, LNPE, SE) perform well on CommonsenseQA, Math, and TriviaQA, while verbalized-based metrics (e.g., VC-NEG, PTrue-Comp) outperform them on TruthfulQA. This may be because TruthfulQA includes questions prone to hallucination, and verbalized methods involving self-reflection via a second inference step help capture uncertainty more effectively. Lexical metrics continue to underperform, likely due to their limited semantic insight.

Surprisingly, as shown in figure, CU does not correlate well with accuracy, despite its intuitive formulation. We believe this failure is due to poor calibration of the model predicted correctness distribution $p(c_i|x)$. When these predicted probabilities do not reflect the true likelihood of correctness, the resulting entropy (CU) fails to represent actual uncertainty. CU can be misleading when the model is over-confidently correct or wrong, making it unreliable as a signal of factual correctness. This highlights a critical mismatch between the model's predicted confidence and the actual correctness, underscoring the need for better-calibrated correctness estimators.

6 Future Directions

Search-Aware Uncertainty Metrics A possible direction for search-aware uncertainty metrics is to decompose total uncertainty into finer-grained sources—such as decoding instability, instruction ambiguity, or semantic under-specification. Rather than relying on a single aggregate score, such metrics could combine multiple signals—for example, a composite score integrating token-level predictive entropy, model self-reported confidence, and variation across sampled outputs, weighted according to the reasoning objective. This type of approach could provide a more detailed view of how different uncertainty sources contribute to generation outcomes, enabling more interpretable, controllable, and diagnostic LLM behavior.

Limitation and Improvement of CU Correctness Uncertainty, as currently estimated via entropy over predicted correctness distribution, suffers from model calibration issues, limiting its reliability as a ground-truth signal. Future work could explore alternative estimators, such as variance-based uncertainty from reward models or scoring functions, which may better capture answer quality and align with correctness objectives. However, these approaches may require additional model training, which could reduce generalizability. Post-hoc calibration methods—such as temperature scaling, isotonic regression, or ensemble-based calibration—could also help improve the alignment between CU and actual correctness, particularly in scenarios where labeled supervision is available.

7 Conclusion

We present a benchmarking framework for evaluating how well existing LLM uncertainty metrics capture four types of uncertainty relevant to target-aware search: AnsU, AU, EU, and CU. Through extensive experiments across models and datasets, we find that while many metrics correlate with certain uncertainty types—such as output diversity or input ambiguity—they often fail to align with correctness-oriented objectives. This mismatch reveals a major limitation in current uncertainty estimation methods, especially for applications requiring reliability and goal-directed reasoning. Our findings highlight the need for developing new, search-aware uncertainty metrics that can adapt to diverse optimization goals and better guide decision-making in LLM search.

Limitations

Due to computational constraints, we evaluate on a sampled subset of each benchmark dataset. While bootstrapping is employed to reduce sampling bias, our findings may still be influenced by the limited scale. Additionally, random sampling of LLM outputs is constrained by budget limitations on scaling depth, width, and the number of questions. Also, our analysis focuses on a representative, but non-exhaustive, set of existing uncertainty metrics, meaning it may not capture all emerging approaches or domain-specific adaptations. Finally, our experiments are conducted using a fixed prompting and sampling strategy, so results may vary under different decoding settings or model configurations.

Use of AI Assistants

ChatGPT was utilized to refine paper writing. The authors paid careful attention to ensuring that AI generated content is accurate and aligned with the author's intentions.

References

Gustaf Ahdritz, Tian Qin, Nikhil Vyas, Boaz Barak, and Benjamin L Edelman. 2024. Distinguishing the knowable from the unknowable with language models. *arXiv preprint arXiv:2402.03563*.

Charles Blundell Balaji Lakshminarayanan, Alexander Pritzel. 2016. Simple and scalable predictive uncertainty estimation using deep ensembles. arXiv:1612.01474. Version 3.

Stanley F Chen, Douglas Beeferman, and Roni Rosenfeld. 1998. Evaluation metrics for language models.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *NeurIPS*.

Cho-Jui Hsieh, Si Si, Felix X Yu, and Inderjit S Dhillon. 2023. Automatic engineering of long prompts. *arXiv preprint arXiv:2311.10117*.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023a. Mistral 7b. *Preprint*, arXiv:2310.06825.

Mingjian Jiang, Yangjun Ruan, Sicong Huang, Saifei Liao, Silviu Pitis, Roger Baker Grosse, and Jimmy Ba. 2023b. Calibrating language models via augmented prompt ensembles.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. triviaqa: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. *arXiv e-prints*, arXiv:1705.03551.

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

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2021. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022a. Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334*.

Zi Lin, Jeremiah Zhe Liu, and Jingbo Shang. 2022b. Towards collaborative neural-symbolic graph semantic parsing via uncertainty. *Findings of the Association for Computational Linguistics: ACL 2022*.

Andrey Malinin and Mark Gales. 2020. Uncertainty estimation in autoregressive structured prediction. *arXiv preprint arXiv:2002.07650*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

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

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. 2023. Large language models as optimizers. *ArXiv*, abs/2309.03409.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2024. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. 2022. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488.

Andy Zhou, Kai Yan, Michal Shlapentokh-Rothman, Haohan Wang, and Yu-Xiong Wang. 2023. Language agent tree search unifies reasoning acting and planning in language models. *arXiv preprint arXiv:2310.04406*.

A Implementation Details

Due to computational constraints, we randomly sample 150 questions from each dataset. To ensure robustness and statistical significance, we apply bootstrapping with 500 resamples across all experiments. For benchmark metric evaluations, we use: 32 samples for NPE, LNPE, and SE; 3 samples for VC-NEG and PTRUE-COMP; 5 paraphrased variants for SPUQ-COMP; and 4 iterative chains (depth 5) for IPT-EU. For SE, we use JINAAI/JINA-EMBEDDINGS-V3 as embedding model to cluster semantic group. All experiments are conducted using a single NVIDIA GeForce RTX 3090 GPU. We use the vLLM engine for efficient inference and encoding text into vector.

B Prompt Templates

Please answer the following question. Think carefully and in a step-by-step fashion.

At the end of your solution, indicate your final answer by writing the answer choice (A, B, C, D, or E) inside a boxed environment, like: $\boxed{A}$.

Q: {q}

Choices: {c}

Your answer:

Figure 4: Sampling Prompt Template for MC Questions

Following is your previous response to the question.

Q: {q}

Choices: {c}

Your previous response: {a}

Check your previous response carefully and solve the same question again.

At the end of your solution, indicate your final answer by writing one of the answer choice (only letter : A, B, C, D, or E) inside a boxed environment, like: $\boxed{A}$.

Output:

Figure 5: Check Prompt Template for MC Questions

Read the following passage and answer the question.

Passage : {p}

Question : {q}

At the end of your solution, indicate your final answer inside a boxed environment, like: $\boxed{answer}$.

Figure 6: Sampling Prompt Template for RC Questions

Following is your previous response to the question:

Read the following passage and answer the question.

Passage : {p}

Question : {q}

Your previous response: {a}

Check your previous response carefully and respond the question again.

At the end of your solution, indicate your final answer inside a boxed environment, like: $\boxed{answer}$.

Figure 7: Check Prompt Template for RC Questions

Please answer the following question.

Think carefully and in a step-by-step fashion.

At the end of your solution, put your final result in a boxed environment, e.g. $\boxed{answer}$.

Q: {q}

Figure 8: Sampling Prompt Template for Essay Questions

Following is your previous response to the question.

Q: {q}

Your previous response: {a}

Check your previous response carefully and solve the same question again step by step.

At the end of your solution, put your final result in a boxed environment, eg. ($\boxed{answer}$).

Output:

Figure 9: Check Prompt Template for Essay Questions

C Additional Result

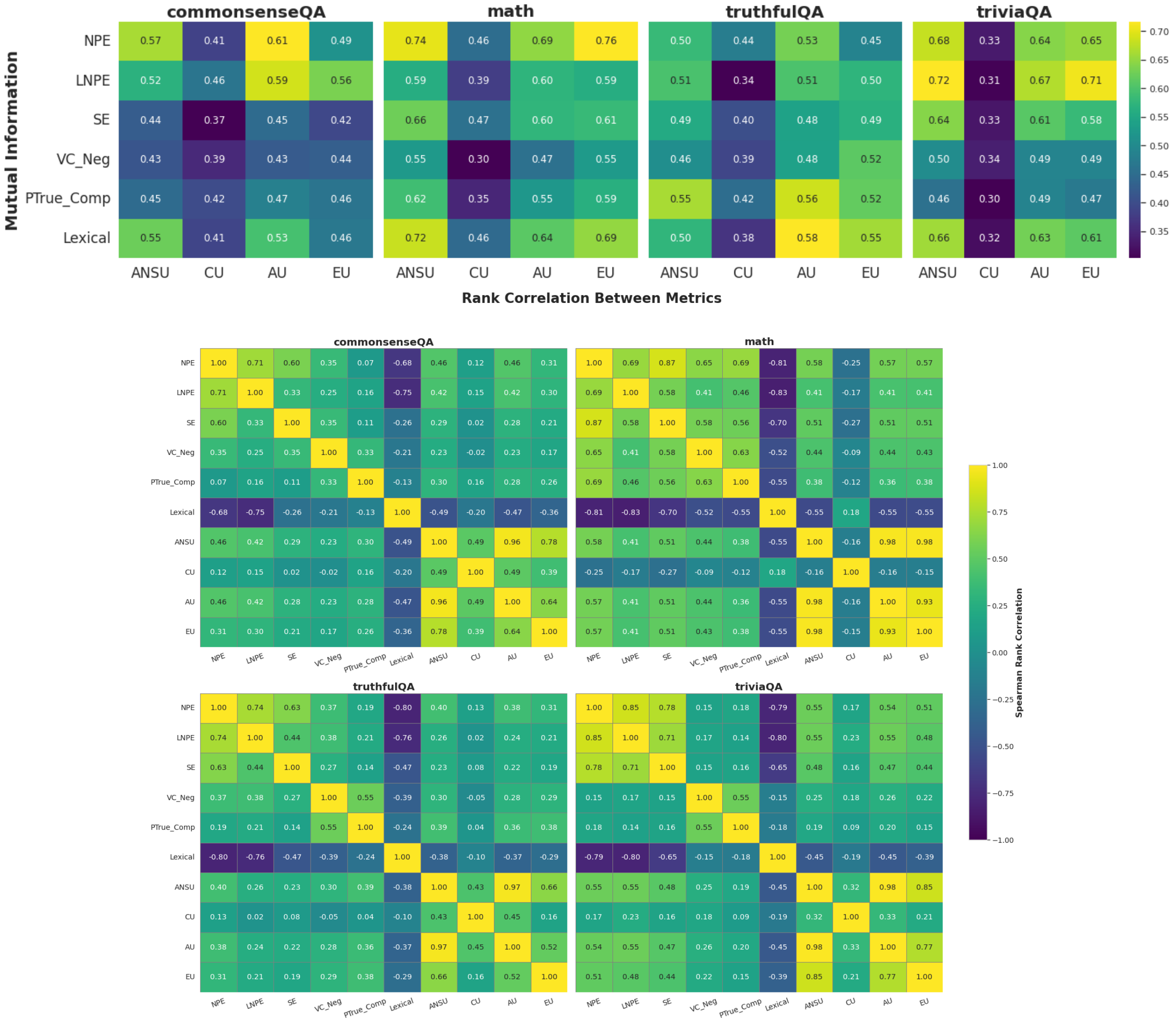

Figure 10: **Mutual Information and Rank Correlation** Top: MI between benchmarked uncertainty metrics and target uncertainties. Higher MI indicates stronger relation. Bottom: Rank correlation between metrics and target uncertainties. Each value is averaged across models.

D Benchmarking Pipeline Pseudo Code

```
Algorithm 1: Pipeline Workflow
foreach input question $Q$ do
    Initialize a reasoning tree with a root node containing $Q$;
    while the reasoning tree is not fully constructed do
        foreach node in the tree that has not yet terminated do
            Step 1: Input Perturbation
            Generate $M$ rephrased inputs $\{x_j\}_{j=1}^{M}$ from the current node's input $\mathbf{x}$;
            Step 2: Random Sampling
            For each rephrased input $\mathbf{x}_j$, sample $K$ responses $\{y_{jk}\}_{k=1}^{K}$;
        Expand the tree by adding child nodes using the newly generated responses;
    Step 3: Ground Truth Uncertainty Calculation
    For each node, calculate the ground truth uncertainty using the answers in its subtree's leaves,
      as described in Section 2;
    Step 4: Uncertainty Metric Calculation
    For each node, compute the estimated uncertainty metric based on its own input and output,
      following the formulas in Section 4;
Step 5: Statistical Analysis
Collect all $(U_{\text{metric}}, U_{\text{true}})$ pairs from every node across all trees.;
Compute Corr$(U_{\text{metric}}, U_{\text{true}})$ by bootstrapping and visualize these relationships to assess how well
  the estimated metrics align with the ground truth.;
```

Note that increasing M and K enhances the Monte Carlo approximation of ground truth values, thereby improving evaluation quality.