MONTE CARLO TEMPERATURE: A ROBUST SAMPLING STRATEGY FOR LLM'S UNCERTAINTY QUANTIFICATION METHODS

Nicola Cecere*, **Andrea Bacciu***, **Ignacio Fernández Tobías**, **Amin Mantrach**
Amazon
nicola.cecere@mail.polimi.it
{andbac, tobiasi, mantrach}@amazon.com

ABSTRACT

Uncertainty quantification (UQ) in Large Language Models (LLMs) is essential for their safe and reliable deployment, particularly in critical applications where incorrect outputs can have serious consequences. Current UQ methods typically rely on querying the model multiple times using non-zero temperature sampling to generate diverse outputs for uncertainty estimation. However, the impact of selecting a given temperature parameter is understudied, and our analysis reveals that temperature plays a fundamental role in the quality of uncertainty estimates. The conventional approach of identifying optimal temperature values requires expensive hyperparameter optimization (HPO) that must be repeated for each new model-dataset combination. We propose Monte Carlo Temperature (MCT), a robust sampling strategy that eliminates the need for temperature calibration. Our analysis reveals that: 1) MCT provides more robust uncertainty estimates across a wide range of temperatures, 2) MCT improves the performance of UQ methods by replacing fixed-temperature strategies that do not rely on HPO, and 3) MCT achieves statistical parity with oracle temperatures, which represent the ideal outcome of a well-tuned but computationally expensive HPO process. These findings demonstrate that effective UQ can be achieved without the computational burden of temperature parameter calibration.

1 INTRODUCTION

Large Language Models (LLMs) have fundamentally transformed the way we interact with artificial intelligence, revolutionizing various domains, from content creation to complex problem-solving tasks (Bommasani et al., 2021; Wei et al., 2022; Orrù et al., 2023). However, these powerful models can sometimes produce unreliable or incorrect outputs, raising concerns about their deployment in critical applications (Rohrbach et al., 2018; Xiao & Wang, 2021; Bacciu et al., 2024). While significant research efforts have focused on improving LLMs' accuracy through techniques like Chain-of-Thought prompting (Wei et al., 2022) and Retrieval-Augmented Generation (Lewis et al., 2020), parallel work has emerged on developing uncertainty quantification (UQ) methods to estimate model confidence as an indicator of potential errors (Kadavath et al., 2022; Kuhn et al., 2023; Lin et al., 2024).

Existing UQ methods for LLMs can be used to predict the correctness of a LLM's output, either under white-blox or black-box assumptions. They fall into two broad categories: *single-sample* and *multi-sample* approaches. *Single-sample* methods analyze a single generation using metrics like perplexity or evaluating model's weight activations. In contrast, *multi-sample* methods, which we focus on in this work, rely on querying the model multiple times with the same input and non-zero fixed temperature sampling, to induce and measure diversity in the generations. To assess the effectiveness of UQ methods in distinguishing between correct and incorrect model outputs, they are typically evaluated as a classification procedure using the area under the receiver operator characteristic curve (AUROC) metric (Hanley & McNeil, 1982). However, the impact of selecting a

*Equal contribution

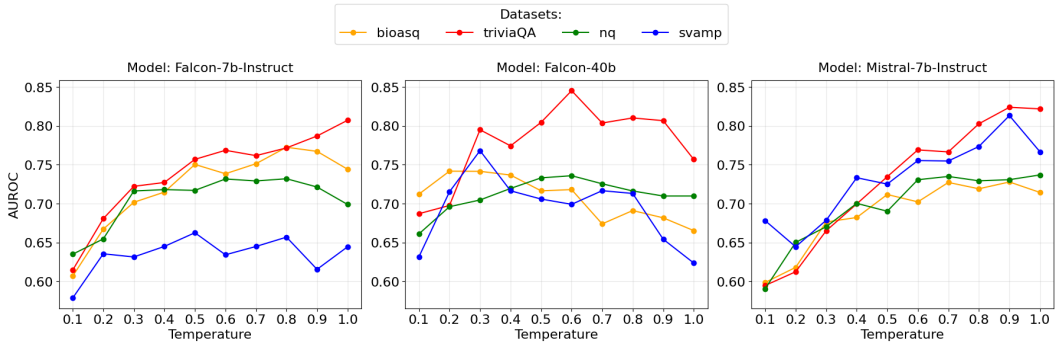

Figure 1: AUROC score distributions of the semantic entropy method across various model-dataset combinations and different fixed temperature values.

specific fixed temperature parameter is understudied, and our analysis reveals that temperature plays a fundamental role in the effectiveness of different UQ methods across scenarios in which different LLMs are employed to solve different tasks. Figure 1 exemplifies this behavior over four question-answering datsets and three models using the semantic entropy method[1] (Kuhn et al., 2023). The figure highlights three critical observations: (1) for a given model and dataset, performance varies significantly with changes in temperature; (2) no single temperature consistently optimizes performance across datasets for a given model; and (3) no universal temperature yields optimal results across models for a given dataset. For instance, the Falcon-40B model achieves peak performance on the TriviaQA dataset at a temperature of 0.6, but requires a lower temperature of 0.3 for the SVAMP dataset. Similarly, within the same TriviaQA dataset, optimal temperature values differ across different models: Falcon-40B performs best at 0.6, while Falcon-7B-Instruct achieves superior results at 1.0. This lack of robustness in maintaining consistent performance across different scenarios poses significant challenges for practitioners attempting to implement UQ methods and highlight the need for more robust approaches to temperature selection.

To address the challenges of selecting a specific fixed temperature in UQ methods, we introduce *Monte Carlo Temperature* (MCT), a sampling strategy that dynamically varies the temperature during multiple sentence generations, allowing UQ methods to generalize more effectively to different model-dataset combinations. This approach reduces sensitivity to specific temperature values and ensures more reliable uncertainty estimates.

We evaluate MCT against an *oracle* determined by selecting the temperature that yields the best results on the test set. By using an oracle as reference, we place ourselves in the most challenging evaluation scenario, as it represents an idealized outcome that hyperparameter optimization (HPO) may not achieve in practice.

Beyond this comparison, we assess MCT against two alternative model-dataset agnostic approaches, that do not require HPO: the *Best On Average Temperature*, which selects a single fixed value performing well across multiple models and datasets, and the *Fixed Random Temperature* approach that randomly chooses a single temperature.

Our results demonstrate that MCT consistently achieves statistical parity with the oracle, eliminating the need for expensive HPO. Additionally, MCT outperforms both the Best On Average Temperature and the Fixed Random Temperature strategies, further highlighting the benefits of structured temperature sampling.

The paper is structured as follows: in Section 2, we present an overview of multi-sample UQ methods. In Section 3, we introduce the MCT approach and describe its implementation. Section 4 details the experimental setup, including the LLMs, datasets, and evaluation metrics used. Section 5 presents the results of our experiments. Finally, in Section 6, we discuss the implications of our findings, acknowledge limitations, and outline potential future research directions.

[1]Similar plots for other UQ methods can be found in the Appendix A.

2 MULTI-SAMPLE UQ METHODS

In this section, we present an overview of popular multi-sample UQ methods that we selected to evaluate the MCT sampling strategy. These methods represent a diverse set of approaches commonly employed for estimating uncertainty in LLMs.

- **Naive Entropy (NE):** NE (Kuhn et al., 2023) computes the uncertainty of model predictions by measuring the entropy of the generated output sequences based on their probabilities. For a given input x, the probability of each output sequence y is computed using the chain rule of probability, which considers the joint probability of each token in the sequence. The entropy is then defined as:

$$H(x) = -\sum_{y \in S} \hat{p}(y|x) \log \hat{p}(y|x), \tag{1}$$

  where S represents the set of sampled sequences used for UQ.
- **Semantic Entropy (SE):** SE (Kuhn et al., 2023) quantifies uncertainty by evaluating entropy across semantic clusters of the generated outputs. These clusters are formed based on semantic similarity, identified using an entailment model (as described in section 4.3). For each cluster c, the probability $\hat{p}(c|x)$ is calculated by summing the probabilities of all sequences within the cluster, i.e., $\hat{p}(c|x) = \sum_{y \in c} \hat{p}(y|x)$, where y represents a sequence assigned to cluster c. Semantic entropy is then computed as:

$$SE(x) = -\sum_{c \in C} \hat{p}(c|x) \log \hat{p}(c|x), \tag{2}$$

  where C represents the set of semantic clusters.
- **Discrete Semantic Entropy (DSE):** Unlike SE, DSE (Farquhar et al., 2024) does not require model-provided probability scores. Instead, it approximates cluster probabilities using the relative frequency of samples within each cluster. This method is particularly effective in black-box settings where access to internal probability scores is restricted.
- **Number of Semantic Sets (NumSemSets):** NumSemSets (Lin et al., 2024) simplifies DSE by directly counting the number of unique semantic clusters identified by the entailment model, where a larger number of clusters indicates higher uncertainty in the model's outputs.
- **P(True):** This technique (Kadavath et al., 2022) is designed to capture the LLM's uncertainty by structuring the task as a multiple-choice question. The LLM first generates a set of candidate answers based on a given prompt and then re-evaluates these responses by assigning probabilities. Specifically, the model is asked to determine whether a generated answer is correct by selecting between ***True*** and ***False***, e.g., *Is the possible answer: (A) True (B) False?*. The probability assigned to **(A)** is recorded as an uncertainty measure. A few-shot prompting strategy with examples from the training set is used to provide contextual guidance.

3 ROBUSTNESS AND MCT SAMPLING FOR UQ

In this section, we define the concept of robustness in the context of UQ methods and formalize the MCT sampling strategy.

3.1 ROBUSTNESS DEFINITION IN UQ METHODS

Robustness in the context of UQ refers to the stability and generalization of a UQ method's performance when applied across different settings. In our use case, robustness captures the range to which a UQ method remains effective in assessing uncertainty under changes in the following dimensions:

- **Inference Parameters**[2]**:** Variability in parameters such as temperature, top-k sampling, or nucleus sampling, which govern the stochastic nature of responses generated by LLMs.

[2]In this work we focused on the study of the temperature parameter. Future work will focus on the other common generation parameters.

- **Model Diversity:** Differences in architectures, training objectives, and scales of LLMs, requiring the UQ method to adapt without significant degradation in performance.
- **Dataset Variability:** Application to datasets with differing domains, topics, or complexity levels, ensuring the UQ method's efficacy across tasks.

3.2 MONTE CARLO TEMPERATURE

MCT is a novel sampling strategy designed to improve robustness and avoid costly HPO by dynamically varying the temperature parameter across multiple queries for the same input. Traditional methods often rely on a fixed temperature value, τ, selected through HPO. In contrast, MCT eliminates the need for HPO by introducing a probabilistic mechanism that samples temperature values from a predefined distribution.

MCT can be directly applied to any existing UQ multi-sample strategy. Instead of determining the ideal fixed temperature through extensive tuning, MCT dynamically samples temperatures, enabling the same UQ multi-sample method to perform robustly without additional optimization. This approach ensures that the method adapts seamlessly across varying model-dataset combinations.

The process of applying MCT to a query x involves the following steps:

1. Define a temperature distribution $p(T)$ with support $[\tau_{\min}, \tau_{\max}]$, where $\tau_{\min}$ and $\tau_{\max}$ represent the minimum and maximum temperatures considered for sampling.
2. Draw k independent samples from the temperature distribution:
$$\tau_i \sim p(T), \quad i \in 1, \ldots, k.$$
3. Generate k responses y_i from a model $\mathcal{M}$, where each response is conditioned on the query x and the corresponding sampled temperature τ_i:
$$y_i = \mathcal{M}(x; \tau_i), \quad i \in \{1, \ldots, k\}.$$
4. Apply the selected UQ multisample method based on the generated responses $\{y_1, y_2, \ldots, y_k\}$.

For this work, we used a discrete distribution with possible temperature values selected as equidistant points between the specified bounds τ_{min} and τ_{max}. For a given number of generations k, the temperature values are drawn without replacement from the discrete set:

$$\{\tau_{\text{min}}, \tau_{\text{min}} + \Delta, \tau_{\text{min}} + 2\Delta, \ldots, \tau_{\text{max}}\}, \tag{3}$$

where $\Delta = \frac{\tau_{\text{max}} - \tau_{\text{min}}}{k-1}$.

4 EXPERIMENTAL SETUP

This section outlines the experimental framework employed to evaluate the performance of MCT and related UQ methods. We detail the configurations used for answer generation, the LLMs and datasets selected for evaluation, and the specific entailment and evaluation models utilized in the study.

4.1 CONFIGURATION FOR GENERATING ANSWERS

In this study, we applied UQ methods to the open question-answering task, focusing on sentence-length outputs. The temperature parameter for our experiments was sampled within the range $\tau_{\text{min}} = 0.1$ to $\tau_{\text{max}} = 1.0$. To ensure a balance between computational efficiency and statistical robustness, we generated $k = 5$ outputs per question. Prior research has demonstrated that using 5 generations provides results that closely approximate those obtained with 10 generations (Farquhar et al., 2024; Lin et al., 2024).

Once the parameters τ_{min}, τ_{max}, and k are defined, applying equation 3 yields the exact interval that we employed for MCT sampling: $\{0.100, 0.325, 0.550, 0.775, 1.000\}$.

4.2 LLMs and Datasets

We evaluated the following LLMs: Falcon-7B-Instruct (Almazrouei et al., 2023), Mistral-7B-Instruct-v0.3 (Jiang et al., 2023), Falcon-40B (Almazrouei et al., 2023), and LLaMA-8B-Instruct-v3.1 (Grattafiori et al., 2024). Note that due to the licensing of LLaMA models family, we accessed it via an API that provided text generations without likelihood scores.

Our experiments employed four open-question datasets covering various topics: TriviaQA (Joshi et al., 2017) and Natural Questions (Kwiatkowski et al., 2019) for general knowledge, SVAMP (Patel et al., 2021) for mathematics, and BIOASQ (Tsatsaronis et al., 2015) for biology.

We sampled 1,000 questions from each dataset, except for SVAMP, which contains fewer samples. In this case, all available questions were used. Notably, this represents a dataset size 2.5 times larger than that employed in the work of Farquhar et al. (2024).

4.3 Entailment and Evaluation Model

This study employs semantic clustering to assess bidirectional entailment between pairs of answers, following the methodology outlined in Farquhar et al. (2024). To implement it, we adopted an LLM-as-Judge approach, utilizing the Amazon Nova Micro (Intelligence, 2024) model to perform clustering tasks.

For response correctness evaluation, we employed the LLM-as-Judge paradigm, a method proven to be more reliable than traditional substring-overlap metrics (Santilli et al., 2024; Zheng et al., 2023). Claude Haiku 3.5 (Anthropic, 2024) served as the evaluation model, configured to assess correctness based on the original question and reference answer in the dataset. To maintain consistency with Farquhar et al. (2024), we ensured that correctness evaluation was conducted using an additional response generated with a fixed temperature of 0.1. This setting minimizes randomness, producing more deterministic outputs that serve as a stable basis for evaluation.

Our evaluation framework mirrors the dual LLM-as-Judge structure employed in Farquhar et al. (2024), where one model is dedicated to clustering and the other to correctness evaluation. However, while the original framework utilized GPT-3.5 for clustering and GPT-4 for evaluation (Brown et al., 2020; OpenAI et al., 2024), we relied on alternative LLMs.

To assess the effectiveness of UQ methods, we measured performance using AUROC, PR-AUC, and AURAC metrics (Hanley & McNeil, 1982; Davis & Goadrich, 2006; Farquhar et al., 2024). Confidence intervals at the 95% level were computed for all metrics via bootstrapping to ensure statistical relevance.

5 Results

In this section, we present the results of the MCT sampling strategy, comparing its performance against three baselines: (1) the oracle temperature, selected to maximize test set performance, (2) the Best On Average Temperature across model-dataset combinations, and (3) the Fixed Random Temperature approach. First, we assess how closely MCT approximates the oracle temperature and achieves statistical parity. Then, we compare MCT to the two baselines that do not rely on HPO. Our results reveal that a previously optimal temperature does not necessarily generalize well across different model-dataset settings, as the Best On Average Temperature still underperforms relative to MCT. Meanwhile, the random baseline highlights the drawbacks of uninformed selection, showing that arbitrary temperature choices lead to unpredictable and often suboptimal results.

5.1 Statistical Parity with Oracle Temperatures

Figure 2 demonstrates that MCT achieves statistical parity with optimal oracle-fixed temperatures across all UQ methods, models, and datasets, using statistical analysis at 95% confidence level. This finding suggests that MCT can effectively replace any fixed temperature sampling approach while eliminating the need for temperature tuning. These results are further validated by additional performance metrics (PRAUC and AURAC), with detailed visualizations available in Appendix A.

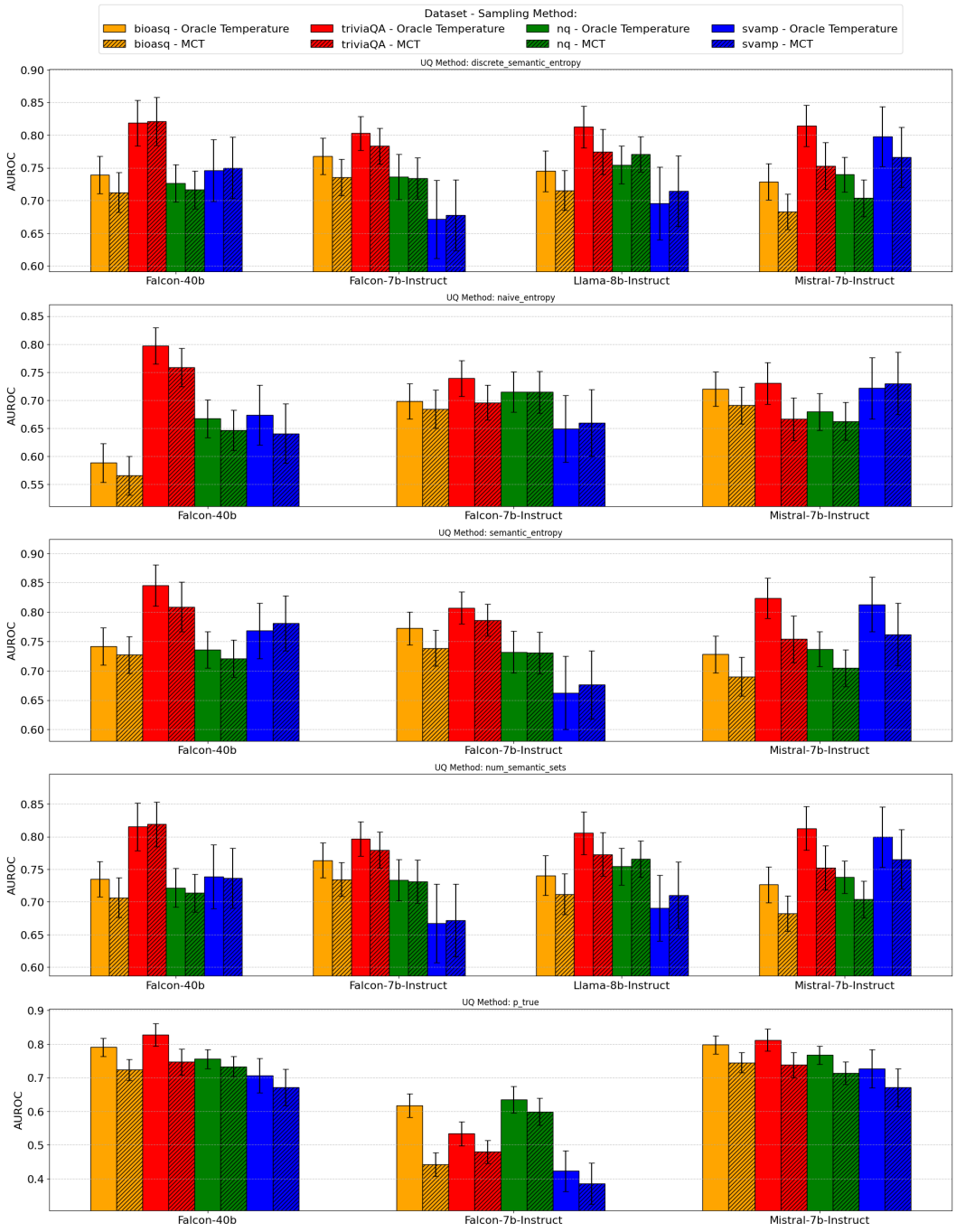

Figure 2: Comparison between oracle-fixed temperature performance and MCT sampling strategy performance across different UQ methods using the AUROC metric.

5.2 Comparison with the Best On Average and Fixed Random Temperature

We evaluated MCT against a baseline approach that determined the best fixed temperature by averaging the scores obtained with each fixed temperature across all model-dataset combinations. To ensure an unbiased comparison, we applied leave-one-out cross-validation, systematically excluding each selected model along with all its associated datasets, as well as each selected dataset along with all its associated models, in the tested combination. The optimal temperature was then determined by averaging performance across the remaining combinations. This approach ensured that the test combination did not influence the temperature selection, effectively eliminating potential bias.

Additionally, we performed a comparison against a random baseline. To construct this baseline, we randomly sampled a fixed temperature 100 times from the same discrete range as MCT and computed the average performance across these simulations. This ensures a robust estimation of the expected performance when selecting a temperature at random, serving as an additional reference point for evaluating MCT's effectiveness.

To assess performance, we quantified the relative difference, denoted as Δ, which measures the deviation of each method (MCT, the best average fixed temperature, and the random baseline) from the oracle temperature's performance. The results show that MCT consistently achieves a lower average Δ across all model-dataset configurations. Specifically, the average Δ for the best average fixed temperature method is 5.34%, while for the random baseline, it is higher at 5.85%. In contrast, MCT achieves an average Δ of 3.77%, demonstrating its superior adaptability and accuracy.

Moreover, this advantage translates into strong win-rate performance for MCT. It outperforms the Best Average Fixed Temperature method in 63.24% of cases and achieves an even greater win rate of 72.03% against the Random Baseline, further confirming its robustness.

Fine-grained results supporting these findings are provided in Table 1 for the AUROC metric and in Appendix A for the other metrics (PR-AUC, AURAC).

6 Conclusion

In this work, we introduced MCT, a general and robust sampling method for UQ in LLMs. Our approach eliminates the need for expensive HPO of temperature parameters, providing consistent performance across a wide range of models, datasets, and UQ methods. The experimental results demonstrate that MCT achieves statistical parity with oracle-fixed temperatures obtained through computationally intensive optimization. Additionally, it outperforms the Best On Average and Fixed Random Temperature baselines by reducing performance variability and enhancing robustness across diverse configurations.

MCT's flexibility makes it applicable to any UQ method requiring multiple generations, and its dynamic temperature sampling effectively addresses challenges associated with fixed temperature configurations. This adaptability highlights MCT as a practical solution for deploying UQ methods in real-world scenarios where computational resources are limited.

Despite its positive results, this study has some limitations. While we validated MCT across a diverse set of UQ techniques and LLMs, further exploration is necessary to evaluate its effectiveness on larger-scale models and alternative architectures. Additionally, this work primarily focused on temperature as the inference parameter; future studies should investigate the impact of other sampling techniques and inference configurations, such as top-P and top-k sampling, to extend MCT's applicability.

7 Acknowledge

We want to thank Marcello Federico for his valuable support and feedback on this paper.

Model	Dataset	Oracle	MCT	Best Avg.	Random	MCT Δ (%)	Best Avg. Δ (%)	Random Δ (%)
Discrete Semantic Entropy								
Falcon-7b-Instruct	triviaQA	0.8028	0.7832	0.7880	0.7453	2.44	**1.84**	7.16
	bioasq	0.7681	0.7353	0.7639	0.7178	4.27	**0.54**	6.55
	svamp	0.6713	0.6778	0.6307	0.6267	**-0.97**	6.05	6.64
	nq	0.7361	0.7339	0.7282	0.7036	**0.29**	1.07	4.41
Mistral-7b-Instruct	triviaQA	0.8143	0.7528	0.7326	0.7217	**7.55**	10.03	11.38
	bioasq	0.7286	0.6829	0.7226	0.6880	6.27	**0.82**	5.57
	svamp	0.7981	0.7662	0.7604	0.7192	**3.99**	4.73	9.88
	nq	0.7397	0.7036	0.6937	0.6923	**4.88**	6.22	6.40
Falcon-40b	nq	0.7262	0.7164	0.6966	0.7025	**1.34**	4.07	3.25
	triviaQA	0.8185	0.8208	0.7882	0.7733	**-0.28**	3.71	5.52
	svamp	0.7462	0.7498	0.6255	0.6718	**-0.49**	16.17	9.96
	bioasq	0.7394	0.7125	0.6617	0.6966	**3.64**	10.51	5.79
Llama-8b-Instruct	triviaQA	0.8125	0.7746	0.8023	0.7872	4.66	**1.25**	3.11
	nq	0.7544	0.7708	0.7517	0.7407	**-2.17**	0.35	1.81
	bioasq	0.7450	0.7155	0.7142	0.7197	3.96	4.13	**3.40**
	svamp	0.6957	0.7144	0.6957	0.6637	**-2.69**	0.00	4.59
Naive Entropy								
Falcon-7b-Instruct	triviaQA	0.7391	0.6959	0.6960	0.7021	5.85	5.84	**5.00**
	bioasq	0.6983	0.6842	0.6865	0.6768	2.03	**1.70**	3.08
	svamp	0.6489	0.6595	0.6157	0.6258	**-1.64**	5.12	3.56
	nq	0.7147	0.7145	0.7075	0.6895	**0.04**	1.02	3.54
Mistral-7b-Instruct	triviaQA	0.7303	0.6663	0.7016	0.6556	8.77	**3.93**	10.23
	bioasq	0.7201	0.6907	0.7057	0.6852	4.08	**1.99**	4.84
	svamp	0.7215	0.7298	0.7050	0.6933	**-1.16**	2.28	3.90
	nq	0.6794	0.6626	0.6606	0.6576	**2.47**	2.77	3.20
Falcon-40b	nq	0.6670	0.6464	0.6414	0.6499	3.10	3.84	**2.57**
	triviaQA	0.7973	0.7587	0.7692	0.7665	4.85	**3.53**	3.86
	svamp	0.6733	0.6406	0.5901	0.6338	**4.85**	12.36	5.86
	bioasq	0.5882	0.5656	0.5415	0.5614	**3.86**	7.94	4.57
Semantic Entropy								
Falcon-7b-Instruct	triviaQA	0.8072	0.7861	0.7716	0.7348	**2.61**	4.40	8.97
	bioasq	0.7725	0.7386	0.7725	0.7208	4.39	**0.00**	6.69
	svamp	0.6626	0.6763	0.6343	0.6356	**-2.06**	4.27	4.09
	nq	0.7320	0.7305	0.7320	0.7045	0.21	**0.00**	3.76
Mistral-7b-Instruct	triviaQA	0.8239	0.7538	0.7347	0.7276	**8.50**	10.83	11.69
	bioasq	0.7279	0.6900	0.7023	0.6907	5.21	**3.52**	5.11
	svamp	0.8132	0.7620	0.7554	0.7396	**6.29**	7.10	9.05
	nq	0.7369	0.7046	0.7294	0.6935	4.38	**1.03**	5.90
Falcon-40b	nq	0.7359	0.7209	0.7098	0.7133	**2.05**	3.55	3.08
	triviaQA	0.8452	0.8090	0.8102	0.7742	4.29	**4.15**	8.40
	svamp	0.7682	0.7808	0.6543	0.6888	**-1.64**	14.82	10.34
	bioasq	0.7418	0.7271	0.6818	0.7085	**1.98**	8.09	4.48
Number of Semantic Sets								
Falcon-7b-Instruct	triviaQA	0.7966	0.7795	0.7871	0.7305	2.14	**1.20**	8.30
	bioasq	0.7638	0.7346	0.7624	0.7283	3.82	**0.18**	4.65
	svamp	0.6669	0.6720	0.6215	0.6299	**-0.76**	6.81	5.56
	nq	0.7336	0.7313	0.7265	0.7062	**0.32**	0.98	3.74
Mistral-7b-Instruct	triviaQA	0.8127	0.7526	0.7085	0.7416	**7.40**	12.82	8.74
	bioasq	0.7265	0.6826	0.7189	0.6760	6.05	**1.05**	6.95
	svamp	0.7994	0.7654	0.7606	0.7327	**4.26**	4.85	8.34
	nq	0.7380	0.7038	0.6952	0.6892	**4.64**	5.80	6.62
Falcon-40b	nq	0.7215	0.7138	0.6920	0.6996	**1.07**	4.09	3.03
	triviaQA	0.8153	0.8191	0.7809	0.7724	**-0.46**	4.21	5.26
	svamp	0.7388	0.7365	0.6160	0.6589	**0.31**	16.61	10.81
	bioasq	0.7349	0.7067	0.6567	0.6950	**3.83**	10.65	5.43
Llama-8b-Instruct	triviaQA	0.8056	0.7728	0.8024	0.7877	4.08	**0.39**	2.22
	nq	0.7542	0.7658	0.7506	0.7393	**-1.54**	0.48	1.98
	bioasq	0.7405	0.7120	0.7085	0.7188	3.85	4.32	**2.93**
	svamp	0.6907	0.7104	0.6907	0.6602	**-2.86**	0.00	4.41
P(True)								
Falcon-7b-Instruct	triviaQA	0.5335	0.4796	0.4924	0.4858	10.11	**7.72**	8.95
	bioasq	0.6170	0.4421	0.5442	0.5398	28.33	**11.80**	12.51
	svamp	0.4228	0.3852	0.3802	0.3941	8.89	10.07	**6.78**
	nq	0.6352	0.5990	0.6232	0.6024	5.71	**1.90**	5.17
Mistral-7b-Instruct	triviaQA	0.8122	0.7383	0.7417	0.7680	9.09	8.68	**5.44**
	bioasq	0.7983	0.7445	0.7532	0.7564	6.73	5.65	**5.25**
	svamp	0.7273	0.6709	0.6540	0.6848	7.76	10.09	**5.85**
	nq	0.7672	0.7137	0.7342	0.7393	6.97	4.30	**3.63**
Falcon-40b	nq	0.7556	0.7330	0.6575	0.6899	**2.99**	12.98	8.69
	triviaQA	0.8282	0.7469	0.8005	0.7915	9.82	**3.35**	4.43
	svamp	0.7070	0.6713	0.5797	0.6583	**5.05**	18.00	6.88
	bioasq	0.7906	0.7234	0.7208	0.7573	8.50	8.83	**4.22**

Table 1: Performance comparison of UQ methods using AUROC score. Bold values show best performance per scenario, with Δ indicating difference from oracle baseline (lower Δ is better). Note: MCT Δ may be negative when performance exceeds the oracle baseline.

REFERENCES

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic,

Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. The falcon series of open language models, 2023. URL https://arxiv.org/abs/2311.16867.

Anthropic. The claude 3 model family: Opus, sonnet, haiku, 2024. URL https://www.anthropic.com/news/claude-3-family. Accessed: 2025-01-24.

Andrea Bacciu, Marco Damonte, Marco Basaldella, and Emilio Monti. Handling ontology gaps in semantic parsing. In Danushka Bollegala and Vered Shwartz (eds.), *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (*SEM 2024)*, pp. 345–359, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.starsem-1.28. URL https://aclanthology.org/2024.starsem-1.28/.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, and et al. On the opportunities and risks of foundation models. *CoRR*, abs/2108.07258, 2021. URL https://arxiv.org/abs/2108.07258.

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

Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, pp. 233–240. ACM, 2006.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, June 2024. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-024-07421-0. URL https://www.nature.com/articles/s41586-024-07421-0.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, and et al. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

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.

Amazon Artificial General Intelligence. The amazon nova family of models: Technical report and model card. *Amazon Technical Reports*, 2024. URL https://www.amazon.science/publications/the-amazon-nova-family-of-models-technical-report-and-model-card.

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. Mistral 7b, 2023. URL https://arxiv.org/abs/2310.06825.

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, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know, 2022. URL http://arxiv.org/abs/2207.05221.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation, April 2023. URL http://arxiv.org/abs/2302.09664. arXiv:2302.09664.

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/.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33: 9459–9474, 2020.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Generating with Confidence: Uncertainty Quantification for Black-box Large Language Models, May 2024. URL http://arxiv.org/abs/2305.19187. arXiv:2305.19187 [cs, stat].

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, and et al. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

Graziella Orrù, Andrea Piarulli, Ciro Conversano, and Angelo Gemignani. Human-like problem-solving abilities in large language models using ChatGPT. *Frontiers in Artificial Intelligence*, 6, 2023. ISSN 2624-8212. doi: 10.3389/frai.2023.1199350. URL https://www.frontiersin.org/journals/artificial-intelligence/articles/10.3389/frai.2023.1199350.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are NLP models really able to solve simple math word problems? *CoRR*, abs/2103.07191, 2021. URL https://arxiv.org/abs/2103.07191.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. 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. 4035–4045, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1437. URL https://aclanthology.org/D18-1437/.

Andrea Santilli, Miao Xiong, Michael Kirchhof, Pau Rodriguez, Federico Danieli, Xavier Suau, Luca Zappella, Sinead Williamson, and Adam Golinski. On the protocol for evaluating uncertainty in generative question-answering tasks. In *Neurips Safe Generative AI Workshop 2024*, 2024. URL https://openreview.net/forum?id=jGtL0JFdeD.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R. Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, Yannis Almirantis, John Pavlopoulos, Nicolas Baskiotis, Patrick Gallinari, Thierry Artiéres, Axel-Cyrille Ngonga Ngomo, Norman Heino, Eric Gaussier, Liliana Barrio-Alvers, Michael Schroeder, Ion Androutsopoulos, and Georgios Paliouras. An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition. *BMC Bioinformatics*, 16(1):138, April 2015. ISSN 1471-2105. doi: 10.1186/s12859-015-0564-6. URL https://doi.org/10.1186/s12859-015-0564-6.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Yijun Xiao and William Yang Wang. On hallucination and predictive uncertainty in conditional language generation. In Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty (eds.), *Proceedings*

of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pp. 2734–2744, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.236. URL https://aclanthology.org/2021.eacl-main.236/.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. URL https://arxiv.org/abs/2306.05685.

A Appendix

This appendix provides additional quantitative results supporting the findings presented in the main text. The following figures and tables illustrate the performance of MCT compared to the oracle temperature and non-HPO fixed-temperature strategies, including the best average temperature and random selection, across various model-dataset combinations.

Figures 3, 4, and 5 present the AUROC, PR-AUC, and AURAC score distributions for different UQ methods across a range of fixed temperature values, complementing Figure 1 in the main text. These distributions highlight the significant impact of temperature selection on performance and underscore the limitations of static temperature choices.

Figures 6 and 7 compare the performance of MCT with oracle-fixed temperature values using PR-AUC and AURAC metrics, complementing the results shown in Figure 2.

Tables 2 and 3 provide detailed performance comparisons for each UQ method across multiple models and datasets using the PR-AUC and AURAC metrics, complementing the results shown in Table 1.

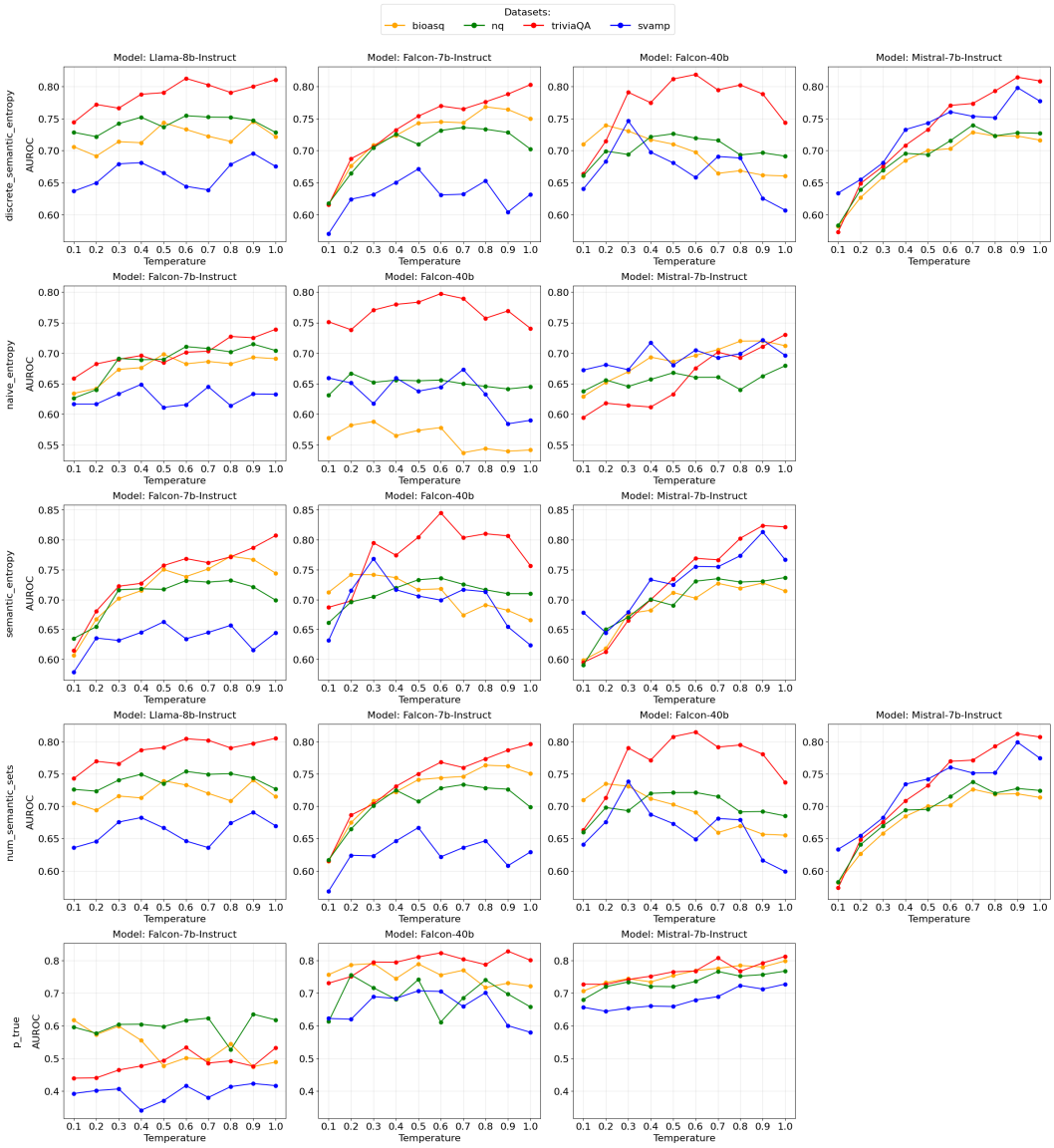

Figure 3: AUROC score distributions of tested UQ methods across various model-dataset combinations at different fixed temperature values.

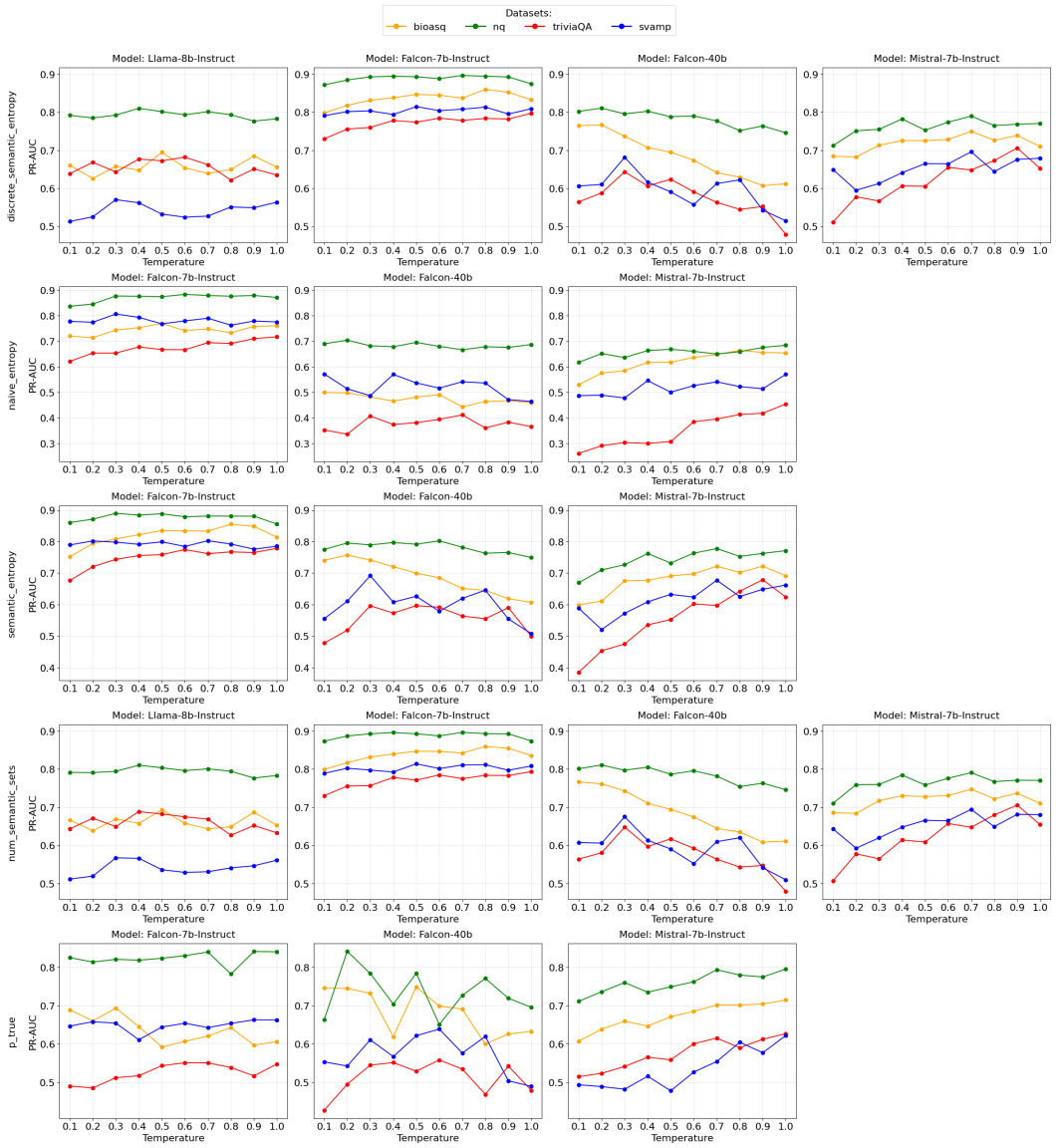

Figure 4: PR-AUC score distributions of tested UQ methods across various model-dataset combinations at different fixed temperature values.

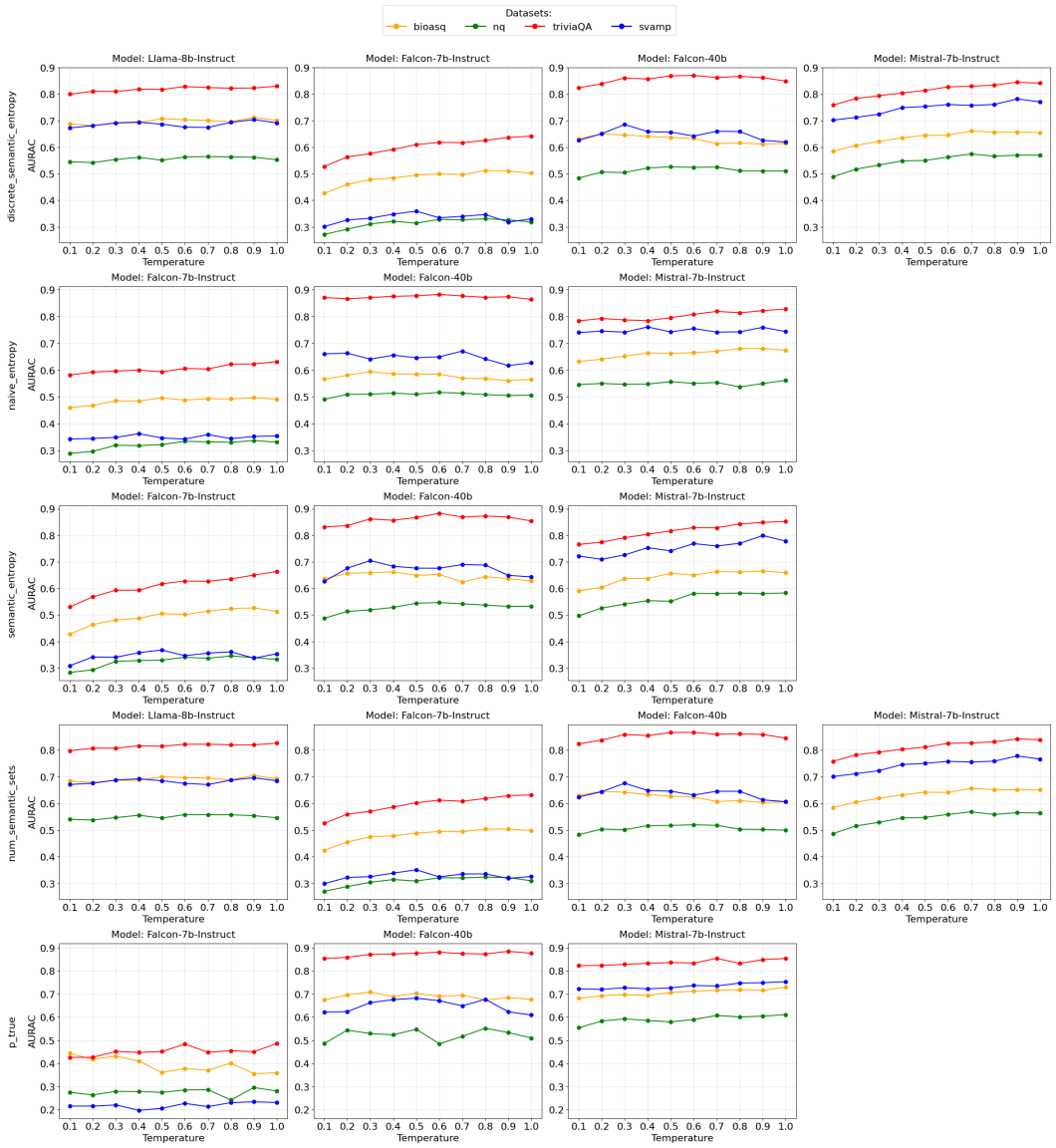

Figure 5: AURAC score distributions of tested UQ methods across various model-dataset combinations at different fixed temperature values.

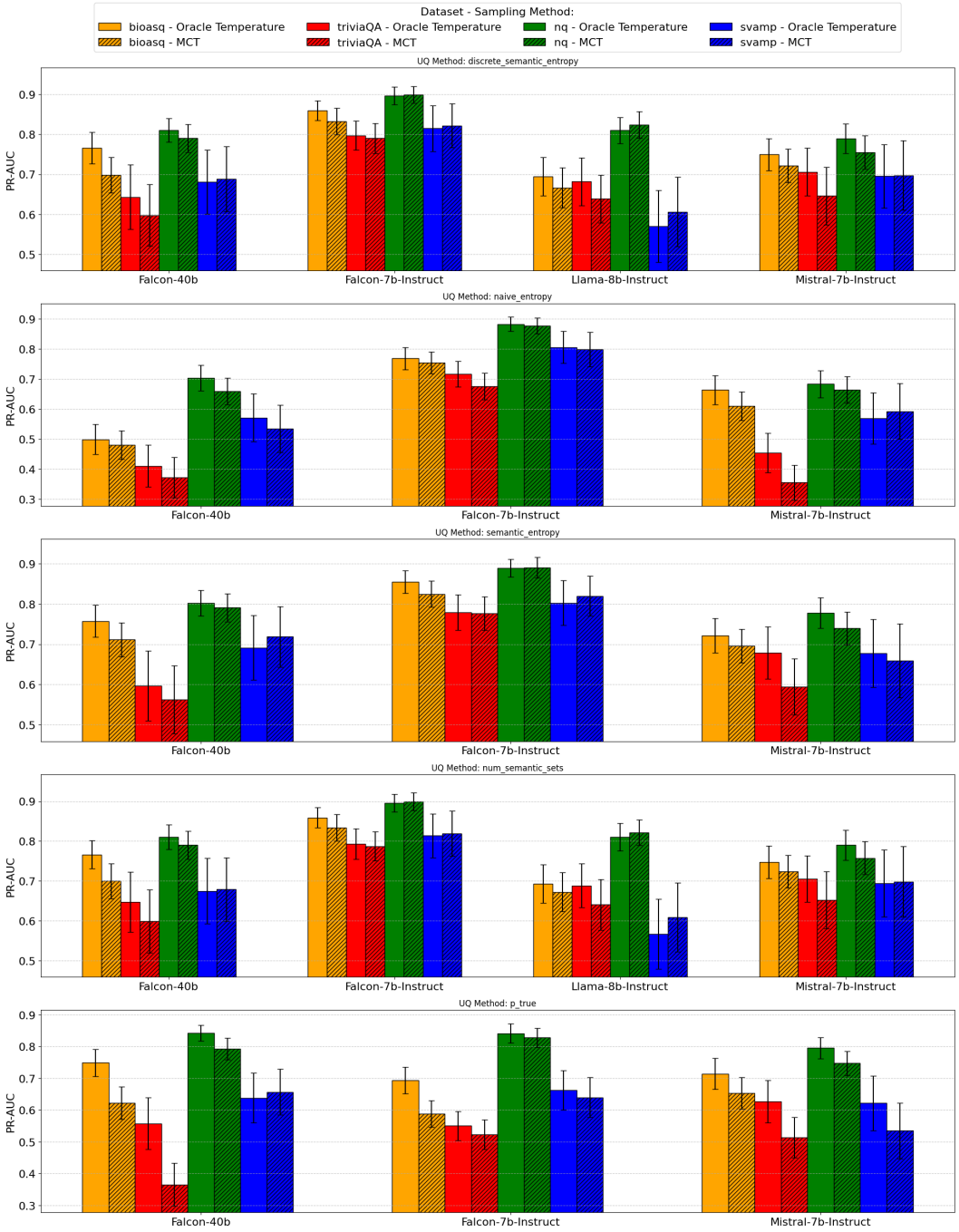

Figure 6: Comparison between oracle-fixed temperature performance and MCT sampling strategy performance across different UQ methods using the PR-AUC metric.

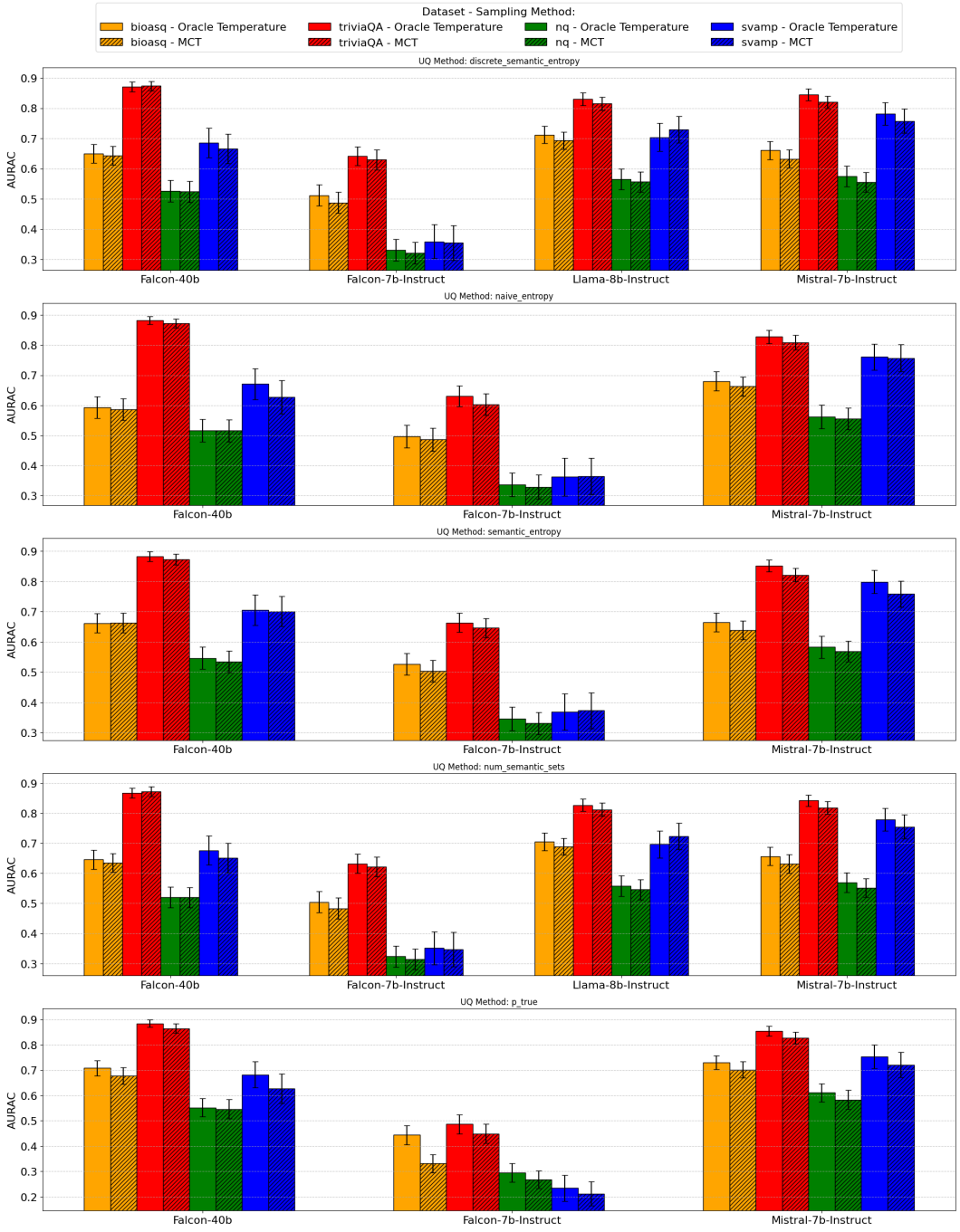

Figure 7: Comparison between oracle-fixed temperature performance and MCT sampling strategy performance across different UQ methods using the AURAC metric.

Model	Dataset	Oracle	MCT	Best Avg.	Random	MCT Δ (%)	Best Avg. Δ (%)	Random Δ (%)
Discrete Semantic Entropy								
Falcon-7b-Instruct	triviaQA	0.7970	0.7899	0.7597	0.7738	**0.89**	4.68	2.92
	bioasq	0.8594	0.8325	0.8379	0.8333	3.13	**2.51**	3.04
	svamp	0.8145	0.8219	0.7937	0.8015	**-0.90**	2.56	1.60
	nq	0.8962	0.8987	0.8925	0.8866	**-0.28**	0.41	1.08
Mistral-7b-Instruct	triviaQA	0.7057	0.6461	0.5665	0.6111	**8.46**	19.73	13.42
	bioasq	0.7493	0.7213	0.7129	0.7206	**3.74**	4.85	3.84
	svamp	0.6956	0.6974	0.6648	0.6496	**-0.25**	4.44	6.61
	nq	0.7893	0.7545	0.7547	0.7613	4.41	4.38	**3.55**
Falcon-40b	nq	0.8103	0.7899	0.7637	0.7839	**2.52**	5.76	3.26
	triviaQA	0.6431	0.5978	0.5630	0.5745	**7.05**	12.47	10.68
	svamp	0.6812	0.6884	0.5426	0.5978	**-1.06**	20.35	12.25
	bioasq	0.7661	0.6980	0.6412	0.6836	**8.89**	16.31	10.77
Llama-8b-Instruct	triviaQA	0.6817	0.6386	0.6427	0.6523	6.31	5.72	**4.30**
	nq	0.8100	0.8240	0.7921	0.7924	**-1.73**	2.21	2.17
	bioasq	0.6945	0.6668	0.6390	0.6568	**3.98**	7.98	5.42
	svamp	0.5701	0.6060	0.5618	0.5399	**-6.28**	1.46	5.30
Naive Entropy								
Falcon-7b-Instruct	triviaQA	0.7169	0.6753	0.6770	0.6758	5.80	**5.56**	5.73
	bioasq	0.7688	0.7540	0.7599	0.7440	1.92	**1.16**	3.22
	svamp	0.8058	0.7984	0.7746	0.7796	**0.92**	3.88	3.26
	nq	0.8826	0.8773	0.8783	0.8695	0.60	**0.49**	1.49
Mistral-7b-Instruct	triviaQA	0.4536	0.3554	0.2993	0.3488	**21.66**	34.02	23.11
	bioasq	0.6634	0.6096	0.6474	0.6134	8.12	**2.42**	7.54
	svamp	0.5695	0.5928	0.5133	0.5161	**-4.08**	9.87	9.39
	nq	0.6832	0.6637	0.6627	0.6582	**2.86**	3.01	3.67
Falcon-40b	nq	0.7034	0.6598	0.6864	0.6822	6.20	**2.41**	3.01
	triviaQA	0.4107	0.3730	0.3644	0.3746	9.17	11.28	**8.80**
	svamp	0.5710	0.5341	0.4634	0.5191	**6.46**	18.84	9.10
	bioasq	0.4988	0.4804	0.4580	0.4734	**3.70**	8.19	5.09
Semantic Entropy								
Falcon-7b-Instruct	triviaQA	0.7789	0.7767	0.7617	0.7473	**0.28**	2.21	4.06
	bioasq	0.8552	0.8247	0.8331	0.8193	3.57	**2.58**	4.19
	svamp	0.8029	0.8200	0.7845	0.7933	**-2.12**	2.29	1.21
	nq	0.8896	0.8908	0.8811	0.8779	**-0.14**	0.95	1.31
Mistral-7b-Instruct	triviaQA	0.6788	0.5939	0.4749	0.5537	**12.51**	30.04	18.43
	bioasq	0.7218	0.6956	0.6755	0.6818	**3.64**	6.42	5.54
	svamp	0.6772	0.6591	0.6324	0.6172	**2.69**	6.62	8.87
	nq	0.7778	0.7395	0.7266	0.7413	4.92	6.58	**4.68**
Falcon-40b	nq	0.8025	0.7909	0.7655	0.7809	**1.44**	4.62	2.69
	triviaQA	0.5964	0.5626	0.5634	0.5551	5.66	**5.52**	6.91
	svamp	0.6915	0.7190	0.5544	0.5896	**-3.97**	19.83	14.74
	bioasq	0.7572	0.7114	0.6187	0.6890	**6.05**	18.29	9.01
Number of Semantic Sets								
Falcon-7b-Instruct	triviaQA	0.7931	0.7871	0.7565	0.7681	**0.76**	4.62	3.16
	bioasq	0.8592	0.8341	0.8394	0.8386	2.92	**2.31**	2.40
	svamp	0.8136	0.8193	0.7922	0.8024	**-0.70**	2.64	1.38
	nq	0.8961	0.8992	0.8925	0.8889	**-0.35**	0.41	0.81
Mistral-7b-Instruct	triviaQA	0.7053	0.6518	0.5646	0.6314	**7.58**	19.96	10.48
	bioasq	0.7470	0.7240	0.7168	0.7168	**3.07**	4.04	4.04
	svamp	0.6942	0.6984	0.6655	0.6553	**-0.60**	4.14	5.61
	nq	0.7904	0.7578	0.7594	0.7621	4.13	3.92	**3.57**
Falcon-40b	nq	0.8105	0.7903	0.7630	0.7808	**2.50**	5.86	3.67
	triviaQA	0.6478	0.5994	0.5636	0.5773	**7.47**	13.01	10.88
	svamp	0.6748	0.6791	0.5408	0.5864	**-0.64**	19.85	13.11
	bioasq	0.7662	0.7001	0.6444	0.6894	**8.62**	15.89	10.01
Llama-8b-Instruct	triviaQA	0.6884	0.6406	0.6490	0.6597	6.94	5.72	**4.17**
	nq	0.8100	0.8217	0.7940	0.7943	**-1.45**	1.98	1.95
	bioasq	0.6930	0.6726	0.6431	0.6609	**2.94**	7.20	4.64
	svamp	0.5671	0.6088	0.5659	0.5383	**-7.36**	0.21	5.07
P(True)								
Falcon-7b-Instruct	triviaQA	0.5504	0.5229	0.5380	0.5274	4.99	**2.26**	4.17
	bioasq	0.6931	0.5879	0.6425	0.6404	15.18	**7.30**	7.61
	svamp	0.6626	0.6399	0.6424	0.6480	3.43	3.05	**2.19**
	nq	0.8413	0.8280	0.8305	0.8255	1.58	**1.29**	1.89
Mistral-7b-Instruct	triviaQA	0.6263	0.5140	0.5407	0.5748	17.93	13.67	**8.22**
	bioasq	0.7144	0.6528	0.6707	0.6708	8.63	**6.11**	**6.11**
	svamp	0.6216	0.5351	0.4813	0.5406	13.91	22.57	**13.03**
	nq	0.7955	0.7473	0.7602	0.7637	6.06	4.43	**3.99**
Falcon-40b	nq	0.8417	0.7927	0.6954	0.7326	**5.82**	17.39	12.96
	triviaQA	0.5577	0.3654	0.4780	0.5111	34.49	14.29	**8.35**
	svamp	0.6386	0.6569	0.4888	0.5747	**-2.88**	23.45	10.00
	bioasq	0.7486	0.6228	0.6323	0.6832	16.80	15.54	**8.73**

Table 2: Performance comparison of UQ methods using PR-AUC score. Bold values show best performance per scenario, with Δ indicating difference from oracle baseline (lower Δ is better). Note: MCT Δ may be negative when performance exceeds the oracle baseline.

Model	Dataset	Oracle	MCT	Best Avg.	Random	MCT Δ (%)	Best Avg. Δ (%)	Random Δ (%)
Discrete Semantic Entropy								
Falcon-7b-Instruct	triviaQA	0.6418	0.6295	0.6367	0.6046	1.91	**0.80**	5.79
	bioasq	0.5120	0.4878	0.5098	0.4833	4.72	**0.42**	5.60
	svamp	0.3592	0.3546	0.3341	0.3320	**1.28**	7.00	7.59
	nq	0.3304	0.3209	0.3265	0.3135	2.88	**1.18**	5.12
Mistral-7b-Instruct	triviaQA	0.8448	0.8200	0.8137	0.8103	**2.93**	3.67	4.08
	bioasq	0.6608	0.6327	0.6567	0.6387	4.25	**0.63**	3.34
	svamp	0.7819	0.7576	0.7609	0.7427	3.11	**2.70**	5.01
	nq	0.5747	0.5557	0.5502	0.5488	**3.30**	4.26	4.51
Falcon-40b	nq	0.5261	0.5245	0.5104	0.5127	**0.32**	2.99	2.56
	triviaQA	0.8703	0.8738	0.8619	0.8569	**-0.40**	0.97	1.53
	svamp	0.6853	0.6654	0.6258	0.6480	**2.90**	8.69	5.46
	bioasq	0.6494	0.6430	0.6115	0.6297	**1.00**	5.84	3.03
Llama-8b-Instruct	triviaQA	0.8298	0.8149	0.8222	0.8183	1.79	**0.91**	1.38
	nq	0.5649	0.5565	0.5634	0.5558	1.49	**0.27**	1.60
	bioasq	0.7121	0.6934	0.6961	0.6971	2.62	2.25	**2.11**
	svamp	0.7042	0.7290	0.7042	0.6863	**-3.52**	0.00	2.55
Naive Entropy								
Falcon-7b-Instruct	triviaQA	0.6314	0.6029	0.6001	0.6056	4.51	4.96	**4.08**
	bioasq	0.4968	0.4867	0.4932	0.4858	2.03	**0.72**	2.22
	svamp	0.3623	0.3652	0.3423	0.3492	**-0.82**	5.50	3.59
	nq	0.3369	0.3287	0.3320	0.3218	2.43	**1.43**	4.47
Mistral-7b-Instruct	triviaQA	0.8278	0.8085	0.8191	0.8022	2.33	**1.05**	3.09
	bioasq	0.6805	0.6631	0.6699	0.6601	2.56	**1.56**	3.00
	svamp	0.7608	0.7573	0.7550	0.7466	**0.46**	0.76	1.88
	nq	0.5618	0.5554	0.5537	0.5500	**1.14**	1.44	2.09
Falcon-40b	nq	0.5166	0.5159	0.5049	0.5087	**0.14**	2.28	1.52
	triviaQA	0.8823	0.8730	0.8735	0.8724	1.05	**1.00**	1.12
	svamp	0.6708	0.6276	0.6270	0.6460	6.44	6.53	**3.69**
	bioasq	0.5938	0.5872	0.5653	0.5765	**1.10**	4.79	2.91
Semantic Entropy								
Falcon-7b-Instruct	triviaQA	0.6637	0.6464	0.6356	0.6074	**2.62**	4.24	8.48
	bioasq	0.5266	0.5036	0.5233	0.4942	4.35	**0.63**	6.15
	svamp	0.3680	0.3728	0.3462	0.3471	**-1.29**	5.93	5.69
	nq	0.3458	0.3308	0.3458	0.3242	4.33	**0.00**	6.25
Mistral-7b-Instruct	triviaQA	0.8522	0.8215	0.8428	0.8148	3.60	**1.10**	4.38
	bioasq	0.6650	0.6390	0.6617	0.6440	3.90	**0.49**	3.16
	svamp	0.7990	0.7584	0.7700	0.7573	5.08	**3.63**	5.22
	nq	0.5829	0.5685	0.5825	0.5554	2.47	**0.07**	4.73
Falcon-40b	nq	0.5466	0.5350	0.5321	0.5300	**2.12**	2.66	3.03
	triviaQA	0.8827	0.8730	0.8688	0.8587	**1.10**	1.57	2.72
	svamp	0.7050	0.7012	0.6491	0.6680	**0.54**	7.92	5.25
	bioasq	0.6622	0.6632	0.6284	0.6449	**-0.15**	5.11	2.62
Number of Semantic Sets								
Falcon-7b-Instruct	triviaQA	0.6320	0.6218	0.6084	0.5912	**1.61**	3.73	6.46
	bioasq	0.5042	0.4834	0.5042	0.4850	4.13	**0.00**	3.81
	svamp	0.3511	0.3475	0.3246	0.3293	**1.03**	7.55	6.20
	nq	0.3240	0.3140	0.3221	0.3100	3.08	**0.57**	4.30
Mistral-7b-Instruct	triviaQA	0.8420	0.8187	0.8118	0.8158	**2.77**	3.59	3.12
	bioasq	0.6567	0.6312	0.6516	0.6302	3.89	**0.78**	4.04
	svamp	0.7783	0.7548	0.7581	0.7469	3.02	**2.59**	4.04
	nq	0.5687	0.5512	0.5478	0.5435	**3.06**	3.67	4.42
Falcon-40b	nq	0.5200	0.5197	0.5025	0.5068	**0.06**	3.38	2.54
	triviaQA	0.8669	0.8715	0.8592	0.8550	**-0.54**	0.89	1.37
	svamp	0.6759	0.6509	0.6134	0.6357	**3.70**	9.24	5.94
	bioasq	0.6456	0.6349	0.6035	0.6246	**1.66**	6.52	3.26
Llama-8b-Instruct	triviaQA	0.8265	0.8120	0.8223	0.8164	1.76	**0.51**	1.23
	nq	0.5582	0.5459	0.5575	0.5498	2.22	**0.13**	1.51
	bioasq	0.7052	0.6887	0.6957	0.6920	2.35	**1.36**	1.87
	svamp	0.6965	0.7233	0.6965	0.6820	**-3.85**	0.00	2.08
P(True)								
Falcon-7b-Instruct	triviaQA	0.4866	0.4485	0.4547	0.4543	7.82	**6.54**	6.63
	bioasq	0.4436	0.3315	0.4009	0.3975	25.26	**9.62**	10.38
	svamp	0.2340	0.2121	0.2340	0.2178	9.37	**0.00**	6.93
	nq	0.2953	0.2673	0.2747	0.2773	9.48	7.00	**6.10**
Mistral-7b-Instruct	triviaQA	0.8542	0.8272	0.8282	0.8376	3.17	3.04	**1.94**
	bioasq	0.7296	0.7012	0.7067	0.7059	3.90	**3.13**	3.25
	svamp	0.7532	0.7207	0.7274	0.7359	4.31	3.42	**2.29**
	nq	0.6109	0.5832	0.5925	0.5936	4.54	3.00	**2.84**
Falcon-40b	nq	0.5519	0.5463	0.5100	0.5231	**1.00**	7.59	5.22
	triviaQA	0.8844	0.8652	0.8759	0.8718	2.17	**0.95**	1.42
	svamp	0.6819	0.6272	0.6091	0.6496	8.03	10.69	**4.74**
	bioasq	0.7084	0.6782	0.6771	0.6901	4.26	4.41	**2.58**

Table 3: Performance comparison of UQ methods using AURAC score. Bold values show best performance per scenario, with Δ indicating difference from oracle baseline (lower Δ is better). Note: MCT Δ may be negative when performance exceeds the oracle baseline