SAMPLE-FOCUSED APPROACH TO ROBUST UNCERTAINTY QUANTIFICATION FOR LLMS

Roman Vashurin*
Mohamed bin Zayed University of AI
Abu Dhabi, UAE

Maiya Goloburda*
Mohamed bin Zayed University of AI
Abu Dhabi, UAE

Artem Shelmanov
Mohamed bin Zayed University of AI
Abu Dhabi, UAE

Preslav Nakov
Mohamed bin Zayed University of AI
Abu Dhabi, UAE

Maxim Panov
Mohamed bin Zayed University of AI
Abu Dhabi, UAE

ABSTRACT

Uncertainty quantification (UQ) methods for Large Language Models (LLMs) encompass a variety of approaches, with two major types being particularly prominent: information-based, which focus on model confidence expressed as token probabilities, and consistency-based, which assess the semantic relationship between multiple outputs generated using repeated sampling. Several recent methods have combined these two approaches and shown impressive performance in various applications. However, they sometimes fail to outperform much simpler baseline methods. Our investigation reveals distinctive characteristics of LLMs as probabilistic models, which help to explain why these UQ methods underperform in certain tasks. Based on these findings, we propose a new way of synthesizing model confidence and output consistency that leads to a family of efficient and robust UQ methods. We evaluate our approach across a variety of tasks such as question answering, abstractive summarization, and machine translation, demonstrating sizable improvements over state-of-the-art UQ approaches.

1 INTRODUCTION

Large Language Models (LLMs) have revolutionized natural language processing (NLP), enabling advancements in information retrieval, question answering, machine translation, and other language-driven applications. As these models become an integral part of everyday life, ensuring the reliability of their outputs is crucial, especially in high-stakes scenarios where errors or uncertainty can have serious consequences. One way to address this challenge is through uncertainty quantification (UQ), which measures how confident a model is in its outputs, and makes possible the rejection of generations with a high risk of being incorrect.

UQ for LLMs is a rapidly advancing research area, with new methods for estimating uncertainty emerging each year. The large portion of novel techniques is based on two fundamental approaches: information-theoretic analysis and the assessment of output consistency.

Information-theoretic methods quantify the confidence of a model by analyzing the probability distributions it induces for predictions (Malinin & Gales, 2021; Fomicheva et al., 2020). A key limitation of these methods is that they cannot account for the semantic variability across multiple possible outputs for the same input. Specifically, the model may generate answers with the same meaning but with very different assigned probabilities; see Figure 1. LLMs are trained to predict the

*Equal contribution.

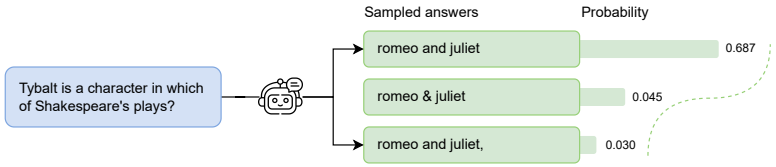

Figure 1: Example of inconsistent probabilities assigned to semantically identical answers by an LLM, demonstrating the limitation of relying solely on sequence-level information.

next token in a sequence based on patterns observed in vast amounts of data, resulting in varying probabilities for semantically equivalent output sequences.

Consistency-based methods, on the other hand, analyze the semantic relationships between the sampled outputs (Lin et al., 2024; Fomicheva et al., 2020), disregarding subjective model confidence. Information-theoretic and consistency-based methods have complementary strengths: the former provides insights into the model's internal confidence, while the latter captures the uncertainty as objective variability of meaning among sampled outputs. For this reason, recent state-of-the-art methods aimed to unify these approaches (Kuhn et al., 2023; Duan et al., 2024). Although such methods show good performance in various applications, they sometimes fail to outperform their simpler counterparts in certain scenarios (Vashurin et al., 2024).

Our investigation revealed distinctive characteristics of LLMs as probabilistic models, shedding light on why current state-of-the-art UQ methods that attempt to integrate both approaches often underperform in certain tasks. Specifically, we highlight the complexity of the token prediction process and the absence of a unified framework that simultaneously addresses model confidence and output variability, both of which can limit the effectiveness of existing UQ techniques. This insight drives our proposal for a novel family of methods that integrate model confidence with output consistency, resulting in more efficient and robust UQ techniques. Our approach combines the strengths of both information-based and consistency-based methods, providing a more comprehensive and accurate assessment of uncertainty.

Our main contributions can be summarized as follows:

- We identify key limitations in current UQ methods for LLMs, particularly in addressing both token- and sequence-level confidence and output consistency.
- We present a family of *Confidence and Consistency-based Approaches* (`CoCoA`) to UQ, offering a new way to merge information- and consistency-based measures for uncertainty quantification in LLMs.
- We evaluate our approaches across a variety of NLP tasks, including question answering, summarization, and translation. Our experiments demonstrate sizable improvements in the reliability and robustness of UQ compared to state-of-the-art methods.

2 Background

In this section, we introduce key concepts related to uncertainty quantification for LLMs, outline existing methods, discuss their limitations, and highlight the motivation for our approach. First and foremost, it is important to establish the concept of an *uncertainty function*. Let $\mathbf{y} = f(\mathbf{x})$ denote the output of an LLM given an input sequence $\mathbf{x}$. The model defines a probabilistic output distribution $p(\mathbf{y} \mid \mathbf{x})$, from which outputs can be sampled. An *uncertainty function* U is a mapping that quantifies the level of uncertainty u associated with the output of a model $\mathbf{y}$, conditioned on the input sequence $\mathbf{x}$, which we denote as

$$u = U(\mathbf{y} \mid \mathbf{x}). \tag{1}$$

2.1 Single-Sequence Information-Based Methods

Information-based methods rely on a single sample from the LLM and estimate the uncertainty of the generated sequence by aggregating the uncertainty scores of individual tokens. One of the simplest

techniques of this kind is *Maximum Sequence Probability (MSP)*. Several other measures fall into this category, including *Perplexity* and *Mean Token Entropy* (Fomicheva et al., 2020); see Appendix C.1 for details. While using only a single sample makes these techniques computationally efficient, they face three major challenges. First, LLMs provide the probability of a specific answer, even though the same meaning can often be expressed in multiple ways. To obtain a proper probability for the conveyed meaning, one would need to marginalize over various possible rephrasings, which is not feasible with only a single generated sample. Second, LLMs are trained to maximize likelihood at the token level rather than for complete answers. Consequently, their probability estimates may not align well with how uncertainty should be measured at the sequence level. For example, longer responses typically have lower probabilities simply due to length effects, making direct interpretation of these probabilities difficult. Third, these methods do not leverage any information about the flatness or variability of the distribution over possible answers.

2.2 Consistency-Based Methods

The aforementioned issues lead to the development of consistency-based methods based on repetitive sampling from the LLM. Consider that we have sampled a set of outputs $\{\mathbf{y}^{(i)}\}_{i=1}^{M}$, where $\mathbf{y}^{(i)} \sim p(\mathbf{y} \mid \mathbf{x})$. Consistency-based uncertainty quantification methods rely only on the diversity of answers $\mathbf{y}^{(i)}$ sampled from the LLM. The idea is that if the model outputs similar answers for the same prompt over and over again, it is confident in its predictions; otherwise, it is uncertain. These techniques do not require knowledge about the probability distribution of the tokens and can be applied in the black-box setting, when only the generated tokens are available. This case is quite common when LLMs are deployed as a service and are accessible through a limited API.

Formally, given M samples from the model, consistency-based methods compute a similarity matrix G, where each element g_{ij} represents some form of similarity between the sampled outputs $\mathbf{y}^{(i)}$ and $\mathbf{y}^{(j)}$:

$$g\big(\mathbf{y}^{(i)}, \mathbf{y}^{(j)}\big) \in [0, 1]. \tag{2}$$

The value $g\big(\mathbf{y}^{(i)}, \mathbf{y}^{(j)}\big) = 1$ indicates the complete equivalence between $\mathbf{y}^{(i)}$ and $\mathbf{y}^{(j)}$, and $g\big(\mathbf{y}^{(i)}, \mathbf{y}^{(j)}\big) = 0$ indicates that there is no similarity.

Similarity could be computed in various ways. *Lexical Similarity* (Fomicheva et al., 2020) is the surface form similarity, which calculates the overlap of words or phrases in the generations. More advanced techniques propose various methods for taking into account the semantic similarity of the generated answers by hard or soft clustering (Lin et al., 2024). The *Degree Matrix* approach considers a similarity matrix G, which is computed using a model for Natural Language Inference (NLI), which predicts the probabilities of entailment, $p_{\text{entail}}(\mathbf{y}, \mathbf{y}')$ and contradiction, $p_{\text{contra}}(\mathbf{y}, \mathbf{y}')$, between pairs of sentences $\mathbf{y}$ and $\mathbf{y}'$. The similarity between two sequences is then defined as either $g_{\text{entail}}(\mathbf{y}, \mathbf{y}') = p_{\text{entail}}(\mathbf{y}, \mathbf{y}')$ or $g_{\text{contra}}(\mathbf{y}, \mathbf{y}') = 1 - p_{\text{contra}}(\mathbf{y}, \mathbf{y}')$.

The detailed formulation of these methods can be found in Appendix C.2. The advantage of these techniques is that by generating multiple samples and analyzing their semantic similarity, they can obtain empirical probabilities for *meanings* instead of individual answers. The main drawback is that they discard the useful information that comes from the probability distribution represented by the LLM, including estimates of the probabilities of specific answers.

2.3 Information-Based Methods with Repeated Sampling

The natural idea is to somehow benefit from having multiple samples from the model while using important information contained in the output probabilities estimated by an LLM. Below, we examine several approaches that have sought to achieve this.

Averaging uncertainties. The uncertainty scores can be aggregated using simple Monte Carlo averaging:

$$u_{\text{MC}} = \frac{1}{M} \sum_{i=1}^{M} u_i. \tag{3}$$

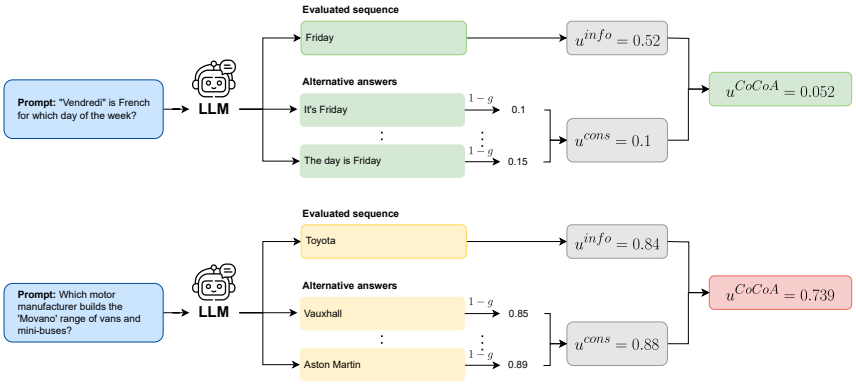

Figure 2: Illustration of the method: the LLM generates a response, evaluates the similarity to alternatives, computes the uncertainty, and combines the uncertainty score with the similarity measures. High similarity to alternatives reduces the uncertainty, while low similarity keeps it high.

For the case when using the MSP uncertainty measure, i.e., when $u_i = -\log p\big(\mathbf{y}^{(i)} \mid \mathbf{x}\big)$, we obtain $u_{\text{MC}} = -\frac{1}{M}\sum_{i=1}^{M} \log p\big(\mathbf{y}^{(i)} \mid \mathbf{x}\big)$. The other notable example is the Monte Carlo Sequence Entropy (Kuhn et al., 2023).

While simple averaging represents a natural way to aggregate uncertainties, it has certain issues related to the nature of LLMs. First of all, in the vast majority of applications, an LLM-based system should produce a single output $\mathbf{y}_*$ for an input query. When we consider u_{MC}, we essentially perform averaging of uncertainties of different sequences, thus somewhat assessing the uncertainty related to the entire generative distribution $p(\mathbf{y} \mid \mathbf{x})$ for the input $\mathbf{x}$, but not for a particular generated sequence $\mathbf{y}_*$. This averaged uncertainty might not be adequate for this particular sequence and, remarkably, often performs worse than the uncertainty $u_* = U(\mathbf{y}_* \mid \mathbf{x})$, which is related solely to the output $\mathbf{y}_*$. Moreover, although intuitive, this naïve aggregation method assumes that all outputs contribute equally to the final uncertainty estimate, regardless of their semantic relationships. This can lead to inconsistencies when semantically equivalent outputs have varying uncertainty scores or when outputs with low similarity are treated as equally important.

Semantically weighted averaging. The basic idea of aggregation approaches like Semantic Entropy Kuhn et al. (2023) or SAR Duan et al. (2024) is to perform a weighted averaging of output probabilities and give more weight to sequences semantically similar to the response shown to a user. All recently proposed techniques, such as SAR and Semantic Entropy, can be unified into a semantically-aware Generalized Monte Carlo uncertainty estimate, defined as

$$u_{\text{GMCU}} = \frac{1}{M}\sum_{i=1}^{M} h\left(\sum_{j=1}^{M} g_{ij}\, p_j\right). \tag{4}$$

Here, the inner summation aggregates sequence probabilities p_j weighted by their semantic similarity to the i-th output, and the outer summation averages these contributions across all samples. The function $h(\cdot)$ provides an additional layer of flexibility, transforming the reweighted uncertainty scores, making the method a generalized framework for uncertainty quantification. Existing methods, such as Semantic Entropy and SAR, can be considered as special cases of this more comprehensive approach, where the functions h and g are chosen appropriately.

Unfortunately, methods that fall under GMCU, while offering benefits, also inherit the aforementioned issues from both categories of methods. (1) The term $\sum_{j=1}^{M} g_{ij}\, p_j$ aims to average the probabilities of semantically similar sequences to obtain a more robust estimate of the probability. However, due to the extreme instability of the LLM probabilities, as shown in Figure 1, the aggregated probabilities often perform worse than non-aggregated baselines. (2) The outer summation in equation 4, similarly to the case of simple Monte Carlo averaging equation 3, often fails to outperform the uncertainty $u_* = U(\mathbf{y}_* \mid \mathbf{x})$ of a single generated sequence $\mathbf{y}_*$.

It is important to note that all uncertainty functions discussed in this section possess two key properties. First, **Non-Negativity** ensures that the uncertainty function produces nonnegative values, meaning $U(\mathbf{y}) \geq 0$ for all $\mathbf{y}$. Second, **Monotonicity** dictates that higher values of the uncertainty function U correspond to higher uncertainty. Specifically, if output $\mathbf{y}^{(1)}$ is deemed more uncertain than output $\mathbf{y}^{(2)}$, then $U\big(\mathbf{y}^{(1)}\big) \geq U\big(\mathbf{y}^{(2)}\big)$. These properties are crucial for performing the type of synthesis of confidence and consistency that will be outlined in the following section.

3 CoCoA: Bridging Confidence and Consistency for Better Uncertainty Quantification

We start by summarizing the benefits and drawbacks of various UQ approaches discussed above:

1. Both information-based and (semantic) consistency-based methods provide grounded and useful uncertainty quantification measures.
2. Output probabilities $p(\mathbf{y}^{(j)} \mid \mathbf{x})$, $j = 1, \ldots, M$ might have substantially different values for semantically equivalent outputs, which questions the usefulness of (weighted) averaging these probabilities for uncertainty quantification.
3. For various methods based on the aggregation over multiple samples, the result might be suboptimal due to the noise related to the averaging over all generated outputs. Focusing solely on a particular output sequence and its relation to other generated outputs might be beneficial.

In what follows, we present a family of UQ *Confidence and Consistency-based Approaches* (CoCoA), offering a new way to merge information- and consistency-based measures for uncertainty quantification in LLMs.

Let us consider an actual output sequence $\mathbf{y}_*$ and a set of sampled sequences $\mathbf{y}^{(i)}$, $i = 1, \ldots, M$. Here, $\mathbf{y}_*$ might be one of the sequences $\mathbf{y}^{(i)}$ or might be generated separately. In what follows, we will consider several possible cases, including $\mathbf{y}_*$ being a random sequence from a set $\{\mathbf{y}^{(i)}\}$, $\mathbf{y}_*$ being a sequence from a set $\{\mathbf{y}^{(i)}\}$ having the highest probability, and, finally, $\mathbf{y}_*$ being a sequence found via the beam search procedure.

First, consider an information-based uncertainty score of the output $\mathbf{y}_*$:

$$u_*^{\text{info}} = U^{\text{info}}(\mathbf{y}_* \mid \mathbf{x}), \tag{5}$$

where U^{info} might be MSP, perplexity, mean token entropy, or another uncertainty measure related solely to the generated sequence $\mathbf{y}_*$.

We quantify the consistency-based uncertainty via a direct measurement of the semantic similarity of generated sequence $\mathbf{y}_*$ to sampled sequences:

$$u_*^{\text{cons}} = \frac{1}{M} \sum_{i=1}^{M} (1 - g_{*i}), \tag{6}$$

where $g_{*i} = g\big(\mathbf{y}_*, \mathbf{y}^{(i)}\big)$. This formulation satisfies the desired properties of the uncertainty function – that is, their values are nonnegative and their values increase with increased inconsistency (decreasing value of g_{*i}). In our ablation study, we will show that such an uncertainty measure reliably outperforms consistency-based measures that aggregate the pairwise similarities of all the samples (see Appendix B.3).

Finally, we need to aggregate u_*^{info} and u_*^{cons} into a single uncertainty measure. We propose to aggregate them in a multiplicative way:

$$u_*^{\text{CoCoA}} = u_*^{\text{info}} \cdot u_*^{\text{cons}} \tag{7}$$

This formulation preserves the non-negativity and the monotonicity properties while integrating both global (semantic) and local (model-specific) uncertainty signals. It ensures that uncertainty is amplified for sequences that are both intrinsically uncertain (high u_*^{info}) and semantically inconsistent with the dataset (high u_*^{cons}), while keeping it low for the opposite scenario (see Figure 2).

Although the choice of the multiplicative aggregation function CoCoA is heuristic, it provides a practical and effective way to combine information- and consistency-based uncertainty signals in LLMs. In our ablation study, we also compare the multiplicative formulation in equation 7 to a simpler additive variant, $u_*^{\text{info}} + u_*^{\text{cons}}$ (see Appendix B.3). Empirically, the multiplicative combination is better at capturing the joint impact of both information-based and consistency-based uncertainty, yielding more reliable estimation across all tasks.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

To evaluate the effectiveness of the proposed method, we extended the LM-Polygraph library (Vashurin et al., 2024; Fadeeva et al., 2023) by implementing our approach within its framework. Since the library already includes tools for calculating other uncertainty scores, it provided a convenient and efficient environment for setting up and running experiments. The primary objective of our experiments is to evaluate whether our method offers improved performance in key tasks such as question answering (QA), summarization (SUM), and machine translation (MT), compared to existing baselines.

Datasets. For QA, we selected diverse datasets to capture a variety of challenges: TriviaQA (Joshi et al., 2017), an open-domain factual QA dataset; CoQA (Reddy et al., 2019), a conversational QA benchmark requiring multi-turn contextual understanding; MMLU (Hendrycks et al., 2021), a multi-task dataset spanning 57 topics to test broad knowledge; and GSM8k (Cobbe et al., 2021), which focuses on grade-school math problems requiring logical reasoning. For translation, we evaluated our method on WMT14 French-English (Bojar et al., 2014) and WMT19 German-English (Barrault et al., 2019). Finally, for summarization, we used XSum (Narayan et al., 2018), a dataset of complex documents paired with concise abstractive summaries. For all datasets, we follow (Vashurin et al., 2024) for selecting the subsets, for prompt formatting, and for the number and sourcing of few-shot examples.

Models. We evaluated our method using the base versions of three open-weight language models: LLaMA 3.1 8B (Touvron et al., 2023), Mistral 7B (Jiang et al., 2023), and Falcon 3 7B (Team, 2024). The open-source nature of these models provides full access to their token probabilities, which are essential for implementing our UQ method. For all models, we consider base versions, without instruction tuning.

Metric	Llama			Mistral			Falcon		
	QA	MT	SUM	QA	MT	SUM	QA	MT	SUM
MCSE	0.357	0.380	0.192	0.453	0.406	0.162	0.460	0.409	0.128
MCNSE	0.380	0.429	0.186	0.466	0.489	0.196	0.530	0.424	0.153
Semantic Entropy	0.397	0.411	0.194	0.482	0.438	0.164	0.479	0.440	0.134
SAR	0.479	0.506	0.159	0.542	0.576	0.175	0.590	0.488	0.193
DegMat	0.422	0.342	0.191	0.465	0.425	0.205	0.543	0.386	0.177
EigValLaplacian	0.388	0.274	0.190	0.426	0.366	0.197	0.498	0.336	0.174
MSP	0.395	0.376	<u>0.464</u>	0.444	0.252	0.330	0.343	0.381	0.099
CoCoA$_{MSP}$	0.484 ↑	<u>0.607</u> ↑	**0.484** ↑	0.526 ↑	<u>0.721</u> ↑	0.366 ↑	0.529 ↑	<u>0.631</u> ↑	0.210 ↑
PPL	0.532	0.563	0.458	0.587	0.686	0.365	0.627	0.589	0.275
CoCoA$_{PPL}$	**0.571** ↑	**0.617** ↑	0.450	**0.613** ↑	**0.745** ↑	<u>0.372</u> ↑	**0.647** ↑	**0.648** ↑	**0.310** ↑
MTE	0.477	0.469	0.449	0.559	0.637	0.350	0.602	0.492	0.186
CoCoA$_{MTE}$	<u>0.548</u> ↑	0.579 ↑	0.451 ↑	<u>0.600</u> ↑	0.720 ↑	**0.373** ↑	<u>0.641</u> ↑	0.614 ↑	<u>0.289</u> ↑

Table 1: Results for Evaluated Sequence – Best Sample: Mean PRR across datasets for each task. The best performing method is in bold, and the second-best is underscored. Arrows indicate improvement in CoCoA over the base version.

Similarity Function. To measure the similarity, we use the RoBERTa-large cross-encoder model, fine-tuned on the Semantic Textual Similarity benchmark dataset (Liu et al., 2019; Reimers &

Gurevych, 2019; Cer et al., 2017). This model is widely regarded as one of the most reliable and commonly used approaches for evaluating sentence similarity. The cross-encoder processes two sequences jointly and directly outputs a similarity score ranging from 0 to 1, providing a nuanced measure. Appendix B.2 contains comparative experiments with cross-encoder and other choices of the similarity function, substantiating this choice.

Baselines. We compare the performance of the proposed method against a diverse set of baseline and state-of-the-art UQ scores, including confidence-based, consistency-based, and hybrid approaches. Information-based methods include Maximum Sequence Probability (MSP), Perplexity (PPL), Mean Token Entropy (MTE), Monte Carlo Sequence Entropy (MCSE), and Monte Carlo Normalized Sequence Entropy (MCNSE). Consistency-based methods include the Degree Matrix (DegMat) and the Sum of Eigenvalues of the Graph Laplacian (EigValLaplacian). Finally, Hybrid methods include Semantic Entropy and SAR. All formulations for these baselines can be found in Appendix C.

Evaluation measure. As our evaluation measure, we choose the Prediction Rejection Ratio (PRR), which measures the effectiveness of the uncertainty scores for identifying high-quality predictions (Malinin & Gales, 2021) (see Appendix D)

Quality Measures. The Predictive Rejection Ratio (PRR) requires an appropriate quality measure for each specific task to effectively evaluate the model output. For question-answering tasks, we use *Accuracy* to directly evaluate whether the generated answers match the ground truth in short-form QA tasks (e.g., MMLU), and we use the *AlignScore* between correct answer and generated sequence for assessing the performance for long-form QA tasks (Zha et al., 2023). For summarization tasks, we use *AlignScore* to measure the alignment between the output summary and the input document. For translation tasks, we use *COMET* (Rei et al., 2020).

Generation Setup. We discuss the generation parameters, the decoding strategy and sample selection procedure in depth in Appendix F. In short, we report evaluation results in two distinct setups: greedy decoding and stochastic sampling with focus on the most probable sequence among the generated outputs (*best-sample*). These two setups offer the highest-quality outputs and are the most reasonable generation approaches in practice.

4.2 RESULTS

Main results. Table 1 shows the PRR scores for the *best-sample* generation. Results for *greedy* generation setup can be found in Appendix E. We calculate a single representative PRR for each task – question answering, machine translation (MT), and summarization (SUM) – by averaging the results across all relevant datasets (e.g., TriviaQA, MMLU, CoQA, GSM8k for QA). This aggregated score provides a concise measure of the performance for each model for each task. Detailed results for each dataset separately can be found in Appendix F.

We can see that our `CoCoA` methods are *the best* across all tasks and models in our experiments. They outperform existing consistency-based and hybrid state-of-the-art approaches, like Semantic Entropy and SAR. In addition, our proposed `CoCoA` approach consistently surpasses the baseline UE metrics: for example, $\texttt{CoCoA}_{PPL}$ outperforms standard Perplexity, illustrating the advantage of combining token-level confidence with semantic consistency. This pattern holds for other information-based metrics as well, demonstrating that using the consistency between multiple sampled outputs reliably enhances uncertainty quantification.

Ablation study. As a part of our ablation study (see Appendix B.1), we evaluate the performance of the average dissimilarity component (u_*^{cons}) independently to assess its effectiveness as a standalone uncertainty measure and to investigate whether it could potentially outweigh the contribution of the information-based component (u_*^{info}) in the enriched uncertainty measure. This evaluation enables us to isolate and better understand the complementary roles and the relative importance of each component. Our experiments demonstrate that the combination of consistency- and confidence-based metrics outperforms the pure consistency-based measure on a vast majority of tasks. Notably, in the few cases where the pure consistency measure outperforms the combined approach, the performance difference is minimal. It is possible that a more suitable choice of a similarity measure or confidence-based metric for the task could further improve the performance.

This leads us to the next part of our ablation study, where we investigate the impact of different similarity measures (see Appendix B.2). We find that for some tasks, the similarity score computed

by the Cross-encoder does not yield optimal performance. For example, for question-answering tasks on CoQA and Trivia, NLI-derived similarity performs better than the Cross-encoder similarity and outperforms the pure consistency-based uncertainty discussed above.

The next section of our ablation study focuses on alternative forms of combining model confidence u_*^{info} and consistency u_*^{cons} (see Apendix B.3). First, we consider an additive form of combining them: $\mathrm{U}_{\text{AdditiveCoCoA}} = u_*^{\text{info}} + u_*^{\text{cons}}$. The results show that this additive formulation does not perform as well compared to the multiplicative one. The additive form tends to underemphasize the interaction between the two components, which is critical for capturing the nuanced relationships between confidence and consistency.

We also consider an alternative formulation of the consistency term u_*^{cons}, as the average of the full pairwise dissimilarity. In this formulation, u_*^{cons} represents the average inconsistency across all samples rather than focusing solely on the dissimilarity of the evaluated sequence with the other samples. Our experiments demonstrate that this formulation is not very strong. By distributing the consistency computation across all samples, it loses focus on the specific sequence being evaluated.

Lastly, in Appendix B.3, we consider alternative formulations of the information-based metric that do not rely on logarithmic transformations. While we primarily use logarithms due to their numerical stability, we explore an alternative approach by converting these values back to probabilities and analyzing their impact on uncertainty estimation. Our findings indicate that both formulations exhibit consistent performance and yield similar results. This suggests that the choice between log-based and probability-based formulations does not affect much the overall performance.

5 LIMITATIONS

While our proposed `CoCoA` approach demonstrates robust empirical performance, several important considerations remain.

Task and Domain Dependency. Our method relies on both an information-based confidence score and a semantic similarity function. The effectiveness of each can vary across models, tasks, and domains. For open-ended tasks with multiple equally valid outputs (e.g., creative generation), consistent rephrasing may inflate the perceived certainty. Conversely, in domains that demand highly precise factual or logical correctness (e.g., math problem solving), small deviations in reasoning can lead to large outcome differences that are not fully captured by a generic similarity measure. Adapting both the confidence measure and the similarity function to specific domains or prompt types is an important direction for future work.

Limited Sample Size. `CoCoA` estimates the model's consistency by sampling multiple outputs and comparing them. In practice, generating a large number of samples can be computationally expensive and may increase the inference latency. Consequently, our experiments (like many sampling-based approaches) rely on relatively small sample sets. Although even a handful of samples can provide a meaningful estimate of consistency, it may not fully capture the diversity of the underlying distribution for certain tasks or for more complex prompts.

Quality Metric. Finally, the `CoCoA`'s performance assessment depends on quality metrics (e.g., COMET for machine translation, and Accuracy for QA) that may not capture every nuance of textual outputs. Automatic metrics can have blind spots, particularly in evaluating coherence, factual correctness, or subtle aspects of style. Further refining or extending quality metrics to account for deeper reasoning, factual faithfulness, and stylistic appropriateness would better align uncertainty scores with real-world perceptions of model correctness.

6 CONCLUSION

We presented `CoCoA`, a unified approach that integrates **Co**nfidence and **Co**nsistency for uncertainty quantification in LLMs. By combining token-level confidence scores with semantic similarity between multiple sampled outputs, CoCoA offers a more holistic view of uncertainty than either approach alone. In extensive evaluations on question answering, summarization, and translation, our approach outperformed existing baselines and state-of-the-art UQ methods. Moreover, `CoCoA`'s flexible design allows easy adaptation to a variety of tasks and settings.

Moving forward, several directions are open for further exploration. These include incorporating more adaptive sampling strategies that efficiently capture the model output space, refining semantic similarity functions for domain-specific tasks, and improving calibration techniques to strengthen the confidence metrics of the model.

REFERENCES

Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. Findings of the 2019 conference on machine translation (WMT19). In Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, André Martins, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana Neves, Matt Post, Marco Turchi, and Karin Verspoor (eds.), Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1), pp. 1–61, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5301. URL https://aclanthology.org/W19-5301/.

Ondrej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Ales Tamchyna. Findings of the 2014 Workshop on Statistical Machine Translation. In Proceedings of the Ninth Workshop on Statistical Machine Translation, pp. 12–58, Baltimore, Maryland, USA, June 2014. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W/W14/W14-3302.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In Steven Bethard, Marine Carpuat, Marianna Apidianaki, Saif M. Mohammad, Daniel Cer, and David Jurgens (eds.), Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), pp. 1–14, Vancouver, Canada, August 2017. Association for Computational Linguistics. doi: 10.18653/v1/S17-2001. URL https://aclanthology.org/S17-2001/.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 5050–5063, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.276. URL https://aclanthology.org/2024.acl-long.276.

Ekaterina Fadeeva, Roman Vashurin, Akim Tsvigun, Artem Vazhentsev, Sergey Petrakov, Kirill Fedyanin, Daniil Vasilev, Elizaveta Goncharova, Alexander Panchenko, Maxim Panov, Timothy Baldwin, and Artem Shelmanov. LM-polygraph: Uncertainty estimation for language models. In Yansong Feng and Els Lefever (eds.), Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 446–461, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-demo.41. URL https://aclanthology.org/2023.emnlp-demo.41/.

Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. Unsupervised quality estimation for neural machine translation. Transactions of the Association for Computational Linguistics, 8: 539–555, 2020. doi: 10.1162/tacl_a_00330. URL https://aclanthology.org/2020.tacl-1.35.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. DeBERTa: Decoding-enhanced bert with disentangled attention. In 9th International Conference on Learning Representations,

ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net, 2021. URL https://openreview.net/forum?id=XPZIaotutsD.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. arXiv preprint arXiv:2310.06825, 2023. URL https://doi.org/10.48550/arXiv.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.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023. OpenReview.net, 2023. URL https://openreview.net/pdf?id=VD-AYtP0dve.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In Text Summarization Branches Out, pp. 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013/.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Generating with confidence: Uncertainty quantification for black-box large language models. Transactions on Machine Learning Research, 2024. URL https://openreview.net/pdf?id=DWkJCSxKU5.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692, 2019. URL https://arxiv.org/abs/1907.11692.

Andrey Malinin and Mark J. F. Gales. Uncertainty estimation in autoregressive structured prediction. In 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net, 2021. URL https://openreview.net/forum?id=jN5y-zb5Q7m.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018, pp. 1797–1807. Association for Computational Linguistics, 2018. doi: 10.18653/v1/d18-1206. URL https://doi.org/10.18653/v1/d18-1206.

Siva Reddy, Danqi Chen, and Christopher D. Manning. CoQA: A conversational question answering challenge. Transactions of the Association for Computational Linguistics, 7:249–266, 2019. doi: 10.1162/tacl_a_00266. URL https://aclanthology.org/Q19-1016.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. COMET: A neural framework for MT evaluation. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 2685–2702, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.213. URL https://aclanthology.org/2020.emnlp-main.213.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, 2019. URL https://aclanthology.org/D19-1410.pdf.

Falcon-LLM Team. The falcon 3 family of open models, December 2024. URL https://huggingface.co/blog/falcon3.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288, 2023. URL https://doi.org/10.48550/arXiv.2307.09288.

Roman Vashurin, Ekaterina Fadeeva, Artem Vazhentsev, Lyudmila Rvanova, Akim Tsvigun, Daniil Vasilev, Rui Xing, Abdelrahman Boda Sadallah, Kirill Grishchenkov, Sergey Petrakov, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, Maxim Panov, and Artem Shelmanov. Benchmarking uncertainty quantification methods for large language models with lm-polygraph. arXiv preprint arXiv:2406.15627, 2024. URL https://arxiv.org/abs/2406.15627.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. Alignscore: Evaluating factual consistency with a unified alignment function. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 11328–11348, 2023. URL https://aclanthology.org/2023.acl-long.634.

A Decoding Strategy and Sample Selection

Modern LLMs are capable of producing output using a wide range of decoding strategies, and it is not readily apparent which one to use as a foundation for UQ experiments. On top of that, when sampling multiple outputs stochastically, one has to decide which sample to select for comparison with the target sequence and UQ purposes.

To facilitate the choice of decoding and sample selection strategies for our experiments, we conducted an evaluation of model performance with different approaches to both. Table **??** shows average values of corresponding quality metrics for all combinations of models and datasets. We considered 4 approaches for the selection of output that subsequently is used to calculate the quality of generation.

- **Greedy decoding** produces single output by selecting top-1 candidate token at each generation step, thus not further selection of sample is needed.
- **Random sample** corresponds to the case where random output is selected among the number of samples produced by repeatedly prompting the model with the same question. In practice we use first generated sample, highlighting model performance when stochastic decoding is done only once.
- **Best (normalized) sample** selects the output with highest model-assigned (length-normalized) probability among several sampled outputs.

We note that selecting a random sample from the model outputs incurs a decrease in the quality of results on several datasets, most prominently on GSM8k. Based on these observations, we evaluate the efficacy of UE on two setups: greedy decoding and stochastic sampling with a focus on the highest-probability sample.

In all experiments, we performed stochastic sampling with temperature $t = 1.0$, top-k equal to 50, and top-p equal to 1.0.

Method	**Metric**	**Generation setup**			
		Greedy	Random	Best	Best Normalized
Mistral7b-Base					
Trivia	Algin Score	0.743	0.655	0.750	0.751
MMLU	Accuracy	0.633	0.558	0.632	0.632
CoQa	Algin Score	0.574	0.403	0.591	0.528
GSM8k	Accuracy	0.382	0.169	0.190	0.290
Xsum	Align Score	0.803	0.578	0.775	0.777
WMT14FrEn	Comet	0.863	0.812	0.830	0.855
WMT19DeEn	Comet	0.864	0.805	0.836	0.851
Llama8b-Base					
Trivia	Algin Score	0.686	0.625	0.687	0.689
MMLU	Accuracy	0.590	0.430	0.597	0.587
CoQa	Algin Score	0.499	0.359	0.529	0.462
GSM8k	Accuracy	0.548	0.234	0.261	0.432
Xsum	Align Score	0.848	0.608	0.825	0.830
WMT14FrEn	Comet	0.863	0.819	0.852	0.859
WMT19DeEn	Comet	0.870	0.816	0.854	0.860
Falcon7b-Base					
Trivia	Algin Score	0.557	0.473	0.568	0.562
MMLU	Accuracy	0.713	0.639	0.713	0.713
CoQa	Algin Score	0.512	0.408	0.560	0.471
GSM8k	Accuracy	0.776	0.313	0.205	0.593
Xsum	Align Score	0.842	0.734	0.782	0.809
WMT14FrEn	Comet	0.867	0.833	0.857	0.863
WMT19DeEn	Comet	0.846	0.807	0.826	0.838

Table 2: Base quality metrics for models for different evaluated sequence choice.

B Ablation

B.1 Average Dissimilarity as Uncertainty Measure

Tables 3 and 4 report PRRs of CoCoA-family methods with uncertainty estimates based solely on average dissimilarity of samples, as proposed in equation equation 6 We observe that it is still widely beneficial to synthesize consistency of outputs with model confidence, even when limiting consistency evaluation to the particular sample to be scored.

Method	**Dataset**						
	XSum	WMT14FrEn	WMT19DeEn	CoQa	Trivia	MMLU	GSM8k
Mistral7b-Base							
AveDissimilarity	0.051	0.285	0.500	0.379	0.647	0.423	0.435
CoCoA$_{MSP}$	**0.330**	**0.396**	**0.598**	**0.383**	0.670	**0.466**	**0.517**
CoCoA$_{PPL}$	0.286	0.375	0.568	0.369	**0.674**	**0.466**	0.467
CoCoA$_{MTE}$	0.288	0.374	0.564	0.355	0.673	0.447	0.491
Llama8b-Base							
AveDissimilarity	0.024	0.389	0.453	**0.375**	**0.614**	0.392	0.368
CoCoA$_{MSP}$	0.378	**0.456**	**0.582**	0.349	0.597	**0.485**	0.372
CoCoA$_{PPL}$	**0.387**	0.448	0.514	0.338	0.593	0.452	0.433
CoCoA$_{MTE}$	0.380	0.446	0.511	0.337	0.601	0.402	**0.447**
Falcon7b-Base							
AveDissimilarity	0.226	0.337	0.496	**0.408**	0.656	0.485	0.426
CoCoA$_{MSP}$	**0.257**	0.433	0.578	0.396	0.684	**0.529**	0.436
CoCoA$_{PPL}$	0.229	0.436	**0.580**	0.406	0.677	**0.529**	0.478
CoCoA$_{MTE}$	0.228	**0.439**	0.577	0.395	**0.685**	0.517	**0.510**

Table 3: Comparison of PRRs of CoCoA-family methods with similarity of greedy output and other samples taken in isolation.

Method	**Dataset**						
	XSum	WMT14FrEn	WMT19DeEn	CoQa	Trivia	MMLU	GSM8k
Mistral7b-Base							
AveDissimilarity	0.071	0.670	0.708	0.405	0.614	0.423	0.846
CoCoA$_{MSP}$	0.366	0.712	0.730	**0.430**	0.644	**0.466**	0.562
CoCoA$_{PPL}$	0.372	**0.735**	**0.755**	0.402	**0.648**	**0.466**	**0.937**
CoCoA$_{MTE}$	**0.373**	0.708	0.732	0.373	0.645	0.447	0.935
Llama8b-Base							
AveDissimilarity	0.030	0.473	0.598	**0.395**	**0.600**	0.353	0.795
CoCoA$_{MSP}$	**0.484**	0.529	0.685	0.384	0.587	**0.452**	0.513
CoCoA$_{PPL}$	0.450	**0.544**	**0.689**	0.364	0.573	0.422	**0.925**
CoCoA$_{MTE}$	0.451	0.520	0.638	0.346	0.582	0.363	0.900
Falcon7b-Base							
AveDissimilarity	0.282	0.491	0.651	0.416	0.627	0.484	0.979
CoCoA$_{MSP}$	0.210	0.564	0.698	**0.428**	**0.659**	**0.530**	0.498
CoCoA$_{PPL}$	**0.310**	**0.579**	**0.717**	0.415	0.644	**0.530**	**1.000**
CoCoA$_{MTE}$	0.289	0.551	0.678	0.402	0.646	0.517	0.998

Table 4: Comparison of PRRs of CoCoA-family methods with similarity of samples with best sample taken in isolation.

B.2 Choice of Similarity Function

For sample consistency estimation, one could come up with a variety of similarity functions $g(\mathbf{y}, \mathbf{y}')$. We perform a comparison of the effectiveness of `CoCoA`-family methods using several such functions. We consider the following functions:

- AlignScore Zha et al. (2023) with `AlignScore-large` model
- RougeL Lin (2004)
- NLI He et al. (2021) based on `microsoft/deberta-large-mnli` model
- CrossEncoder Liu et al. (2019) based on `cross-encoder/stsb-roberta-large` model.

Tables 5 and 6 report these results. There exists a considerable variation of relative effectiveness of proposed methods with various similarity function choices, depending on the task at hand. We opt to report all results in other sections with CrossEncoder-based similarity as it by itself provides an improvement over baselines, and for consistency and ease of comparison reasons. However, when applying these methods to a particular task, we encourage users to select the appropriate underlying similarity function for the best results.

Method	Dataset						
	XSum	WMT14FrEn	WMT19DeEn	CoQa	Trivia	MMLU	GSM8k
Mistral7b-Base							
CoCoA$_{MSP}$							
AlignScore	**0.334**	0.293	0.445	0.354	0.655	0.466	0.550
RougeL	0.289	0.358	0.546	0.369	0.649	0.466	0.536
NLI	0.308	0.313	0.477	**0.403**	0.677	**0.470**	**0.568**
CrossEncoder	0.327	**0.397**	**0.595**	0.381	0.671	0.466	0.505
CoCoA$_{PPL}$							
AlignScore	0.307	0.308	0.489	0.373	0.666	0.466	0.536
RougeL	0.226	0.369	0.531	0.352	0.653	0.466	0.466
NLI	0.233	0.316	0.501	0.376	**0.682**	**0.470**	0.480
CrossEncoder	0.281	0.371	0.565	0.365	0.674	0.466	0.465
CoCoA$_{MTE}$							
AlignScore	0.302	0.299	0.477	0.366	0.664	0.450	0.555
RougeL	0.212	0.377	0.528	0.345	0.652	0.449	0.497
NLI	0.219	0.313	0.488	0.362	0.681	0.453	0.490
CrossEncoder	0.282	0.368	0.560	0.351	0.673	0.448	0.486
Llama8b-Base							
CoCoA$_{MSP}$							
AlignScore	0.367	0.331	0.452	0.308	0.596	**0.484**	0.401
RougeL	0.336	0.393	0.545	0.321	0.563	0.474	0.375
NLI	0.344	0.352	0.467	**0.364**	0.606	0.478	0.419
CrossEncoder	0.375	**0.454**	**0.583**	0.350	0.598	0.480	0.367
CoCoA$_{PPL}$							
AlignScore	**0.422**	0.346	0.450	0.337	0.596	0.453	0.446
RougeL	0.370	0.408	0.486	0.319	0.552	0.441	0.418
NLI	0.374	0.354	0.438	0.348	0.600	0.446	0.409
CrossEncoder	0.380	0.444	0.514	0.339	0.593	0.447	0.429
CoCoA$_{MTE}$							
AlignScore	0.419	0.340	0.438	0.339	0.605	0.411	**0.459**
RougeL	0.362	0.417	0.481	0.319	0.560	0.390	0.440
NLI	0.366	0.342	0.428	0.340	**0.612**	0.396	0.420
CrossEncoder	0.374	0.441	0.511	0.337	0.601	0.394	0.444
Falcon7b-Base							
CoCoA$_{MSP}$							
AlignScore	**0.278**	0.306	0.475	0.361	0.677	0.528	0.470
RougeL	0.205	0.394	0.499	0.378	0.678	0.527	0.417
NLI	0.236	0.361	0.511	0.407	0.684	**0.532**	0.532
CrossEncoder	0.253	0.436	0.577	0.396	0.685	0.529	0.428
CoCoA$_{PPL}$							
AlignScore	0.252	0.340	0.523	**0.410**	0.678	0.528	0.521
RougeL	0.170	0.409	0.537	0.389	0.668	0.527	0.439
NLI	0.193	0.364	0.531	0.408	0.680	**0.532**	0.499
CrossEncoder	0.226	0.437	**0.579**	0.405	0.677	0.529	0.474
CoCoA$_{MTE}$							
AlignScore	0.253	0.337	0.519	0.403	0.683	0.515	**0.554**
RougeL	0.170	0.426	0.540	0.382	0.673	0.514	0.472
NLI	0.190	0.364	0.525	0.398	**0.687**	0.521	0.514
CrossEncoder	0.223	**0.438**	0.575	0.395	0.685	0.517	0.505

Table 5: Comparison of PRRs of CoCoA-family methods with different choices of similarity function with greedy sample taken in isolation.

Method	Dataset						
	XSum	WMT14FrEn	WMT19DeEn	CoQa	Trivia	MMLU	GSM8k
Mistral7b-Base							
CoCoA$_{MSP}$							
AlignScore	0.393	0.448	0.491	0.399	0.626	0.467	0.476
RougeL	0.344	0.602	0.597	0.420	0.622	0.466	0.538
NLI	0.340	0.615	0.604	**0.445**	0.651	**0.470**	0.456
CrossEncoder	0.366	0.712	0.730	0.430	0.644	0.466	0.562
CoCoA$_{PPL}$							
AlignScore	0.474	0.619	0.657	0.408	0.638	0.467	0.910
RougeL	0.362	0.710	0.717	0.391	0.627	0.466	0.950
NLI	0.370	0.677	0.684	0.414	**0.657**	**0.470**	0.941
CrossEncoder	0.372	**0.735**	**0.755**	0.402	0.648	0.466	0.937
CoCoA$_{MTE}$							
AlignScore	**0.492**	0.547	0.590	0.383	0.633	0.449	0.914
RougeL	0.355	0.695	0.684	0.366	0.624	0.448	**0.959**
NLI	0.364	0.656	0.658	0.387	0.656	0.453	0.918
CrossEncoder	0.373	0.708	0.732	0.373	0.645	0.447	0.935
Llama8b-Base							
CoCoA$_{MSP}$							
AlignScore	0.520	0.332	0.491	0.354	0.587	**0.457**	0.401
RougeL	0.471	0.470	0.588	0.362	0.551	0.446	0.499
NLI	0.466	0.442	0.577	**0.386**	0.597	0.446	0.470
CrossEncoder	0.484	0.529	0.685	0.384	0.587	0.452	0.513
CoCoA$_{PPL}$							
AlignScore	0.546	0.406	0.561	0.376	0.577	0.429	0.875
RougeL	0.452	0.518	0.639	0.352	0.532	0.417	**0.931**
NLI	0.458	0.466	0.597	0.365	0.583	0.418	0.912
CrossEncoder	0.450	**0.544**	**0.689**	0.364	0.573	0.422	0.925
CoCoA$_{MTE}$							
AlignScore	**0.561**	0.325	0.497	0.365	0.589	0.380	0.821
RougeL	0.448	0.496	0.598	0.336	0.539	0.361	0.921
NLI	0.449	0.446	0.565	0.344	**0.598**	0.359	0.881
CrossEncoder	0.451	0.520	0.638	0.346	0.582	0.363	0.900
Falcon7b-Base							
CoCoA$_{MSP}$							
AlignScore	0.181	0.378	0.473	0.410	0.654	0.528	0.239
RougeL	0.122	0.531	0.581	0.420	0.655	0.528	0.426
NLI	0.120	0.496	0.607	0.437	0.658	**0.533**	0.458
CrossEncoder	0.210	0.564	0.698	0.428	**0.659**	0.530	0.498
CoCoA$_{PPL}$							
AlignScore	**0.384**	0.454	0.586	**0.440**	0.648	0.528	0.994
RougeL	0.280	0.565	0.668	0.410	0.637	0.528	**1.000**
NLI	0.283	0.515	0.671	0.424	0.647	**0.533**	0.998
CrossEncoder	0.310	**0.579**	**0.717**	0.415	0.644	0.530	**1.000**
CoCoA$_{MTE}$							
AlignScore	0.292	0.386	0.498	0.435	0.648	0.515	0.972
RougeL	0.222	0.545	0.607	0.400	0.633	0.515	0.998
NLI	0.201	0.498	0.636	0.415	0.645	0.521	0.987
CrossEncoder	0.289	0.551	0.678	0.402	0.646	0.517	0.998

Table 6: Comparison of PRRs of CoCoA-family methods with different choices of similarity function with best sample taken in isolation.

B.3 Different Ways of Combining Confidence and Consistency

We justify the particular form of equation equation 7 by considering alternative ways to combine sample-focused confidence with consistency estimation. Results are presented in Tables 7 and 8. In particular, we investigate the performance of the additive approach (AdditiveCoCoA):

$$\mathrm{U}_{AdditiveCoCoA} = u_*^{\text{info}} + u_*^{\text{cons}}, \tag{8}$$

and the same multiplicative combination, replacing sample-focused dissimilarity from equation 6 with the average of the full pairwise dissimilarity matrix equation 16:

$$\mathrm{U}_{FullSampleCoCoA} = u_*^{\text{info}} \cdot U_{Deg}. \tag{9}$$

It is evident that on average the multiplicative form proposed in equation equation 7 with both confidence and consistency terms focused on a single sample is the better performing variant.

Method	**Dataset**						
	XSum	WMT14FrEn	WMT19DeEn	CoQa	Trivia	MMLU	GSM8k
Mistral7b-Base							
AdditiveCoCoA$_{MSP}$	0.290	0.319	0.459	0.351	0.654	**0.471**	0.472
FullSampleCoCoA$_{MSP}$	0.319	0.385	0.590	0.357	0.668	0.467	0.505
ProbCoCoA$_{MSP}$	0.059	0.302	0.520	**0.390**	0.671	0.461	0.435
CoCoA$_{MSP}$	**0.330**	**0.396**	**0.598**	0.383	0.670	0.466	**0.517**
AdditiveCoCoA$_{PPL}$	0.262	0.392	0.564	0.369	0.671	0.464	0.494
FullSampleCoCoA$_{PPL}$	0.277	0.373	0.551	0.334	0.672	0.467	0.435
ProbCoCoA$_{PPL}$	0.297	0.369	0.566	0.373	**0.674**	0.464	0.475
CoCoA$_{PPL}$	0.286	0.375	0.568	0.369	**0.674**	0.466	0.467
AdditiveCoCoA$_{MTE}$	-0.279	-0.058	-0.072	0.098	0.312	0.079	0.187
FullSampleCoCoA$_{MTE}$	0.274	0.368	0.543	0.309	0.668	0.442	0.456
CoCoA$_{MTE}$	0.288	0.374	0.564	0.355	0.673	0.447	0.491
Llama8b-Base							
AdditiveCoCoA$_{MSP}$	0.330	0.345	0.462	0.301	0.566	**0.502**	0.326
FullSampleCoCoA$_{MSP}$	0.358	0.434	0.564	0.333	0.589	0.488	0.354
ProbCoCoA$_{MSP}$	0.031	0.405	0.471	**0.371**	**0.612**	0.461	0.368
CoCoA$_{MSP}$	0.378	**0.456**	**0.582**	0.349	0.597	0.485	0.372
AdditiveCoCoA$_{PPL}$	0.368	0.431	0.504	0.336	0.595	0.455	0.437
FullSampleCoCoA$_{PPL}$	**0.389**	0.420	0.487	0.314	0.580	0.450	0.399
ProbCoCoA$_{PPL}$	0.381	0.445	0.513	0.345	0.599	0.446	0.438
CoCoA$_{PPL}$	0.387	0.448	0.514	0.338	0.593	0.452	0.433
AdditiveCoCoA$_{MTE}$	-0.331	-0.042	-0.122	0.089	0.321	-0.122	0.117
FullSampleCoCoA$_{MTE}$	0.383	0.410	0.481	0.308	0.588	0.363	0.414
CoCoA$_{MTE}$	0.380	0.446	0.511	0.337	0.601	0.402	**0.447**
Falcon7b-Base							
AdditiveCoCoA$_{MSP}$	0.203	0.318	0.409	0.350	0.674	**0.533**	0.379
FullSampleCoCoA$_{MSP}$	0.225	0.423	0.571	0.388	0.678	**0.533**	0.404
ProbCoCoA$_{MSP}$	0.226	0.367	0.515	**0.416**	0.680	0.526	0.426
CoCoA$_{MSP}$	**0.257**	0.433	0.578	0.396	0.684	0.529	0.436
AdditiveCoCoA$_{PPL}$	0.222	0.433	**0.580**	0.413	0.677	0.525	0.489
FullSampleCoCoA$_{PPL}$	0.204	0.425	0.565	0.393	0.669	**0.533**	0.437
ProbCoCoA$_{PPL}$	0.235	0.433	0.576	0.410	0.680	0.528	0.482
CoCoA$_{PPL}$	0.229	0.436	**0.580**	0.406	0.677	0.529	0.478
AdditiveCoCoA$_{MTE}$	0.001	-0.103	-0.106	0.114	0.041	0.138	0.221
FullSampleCoCoA$_{MTE}$	0.201	0.425	0.557	0.377	0.675	0.519	0.470
CoCoA$_{MTE}$	0.228	**0.439**	0.577	0.395	**0.685**	0.517	**0.510**

Table 7: Comparison of PRRs of CoCoA-family methods with alternative formulations with greedy sample taken in isolation.

Method	Dataset						
	XSum	WMT14FrEn	WMT19DeEn	CoQa	Trivia	MMLU	GSM8k
Mistral7b-Base							
AdditiveCoCoA$_{MSP}$	0.333	0.239	0.310	0.406	0.631	**0.472**	0.311
FullSampleCoCoA$_{MSP}$	0.354	0.543	0.565	0.412	0.643	0.468	0.428
ProbCoCoA$_{MSP}$	0.076	0.684	0.721	0.428	0.643	0.464	0.846
CoCoA$_{MSP}$	0.366	0.712	0.730	**0.430**	0.644	0.466	0.562
AdditiveCoCoA$_{PPL}$	0.368	0.737	0.751	0.406	0.644	0.465	0.939
FullSampleCoCoA$_{PPL}$	**0.383**	0.714	0.723	0.379	**0.649**	0.468	0.933
ProbCoCoA$_{PPL}$	0.369	**0.738**	**0.756**	0.401	**0.649**	0.467	0.935
CoCoA$_{PPL}$	0.372	0.735	0.755	0.402	0.648	0.466	0.937
AdditiveCoCoA$_{MTE}$	0.368	0.723	0.702	0.332	0.643	0.452	**0.942**
FullSampleCoCoA$_{MTE}$	0.380	0.661	0.653	0.331	0.643	0.442	0.929
CoCoA$_{MTE}$	0.373	0.708	0.732	0.373	0.645	0.447	0.935
Llama8b-Base							
AdditiveCoCoA$_{MSP}$	0.466	0.349	0.425	0.333	0.555	**0.473**	0.285
FullSampleCoCoA$_{MSP}$	0.476	0.462	0.619	0.363	0.574	0.464	0.379
ProbCoCoA$_{MSP}$	0.035	0.491	0.617	**0.398**	**0.598**	0.433	0.795
CoCoA$_{MSP}$	**0.484**	0.529	0.685	0.384	0.587	0.452	0.513
AdditiveCoCoA$_{PPL}$	0.454	0.536	0.673	0.358	0.575	0.425	0.923
FullSampleCoCoA$_{PPL}$	0.459	0.525	0.649	0.343	0.556	0.430	0.914
ProbCoCoA$_{PPL}$	0.438	**0.547**	**0.689**	0.364	0.574	0.419	0.923
CoCoA$_{PPL}$	0.450	0.544	**0.689**	0.364	0.573	0.422	**0.925**
AdditiveCoCoA$_{MTE}$	0.457	0.496	0.579	0.304	0.561	0.361	0.901
FullSampleCoCoA$_{MTE}$	0.455	0.464	0.577	0.313	0.563	0.341	0.878
CoCoA$_{MTE}$	0.451	0.520	0.638	0.346	0.582	0.363	0.900
Falcon7b-Base							
AdditiveCoCoA$_{MSP}$	0.100	0.397	0.394	0.393	0.649	**0.534**	-0.156
FullSampleCoCoA$_{MSP}$	0.144	0.531	0.607	0.416	0.654	0.533	0.189
ProbCoCoA$_{MSP}$	0.282	0.522	0.670	**0.434**	0.658	0.529	0.978
CoCoA$_{MSP}$	0.210	0.564	0.698	0.428	**0.659**	0.530	0.498
AdditiveCoCoA$_{PPL}$	0.297	0.582	0.706	0.417	0.643	0.526	**1.000**
FullSampleCoCoA$_{PPL}$	0.297	0.560	0.670	0.405	0.641	0.533	**1.000**
ProbCoCoA$_{PPL}$	**0.311**	**0.587**	**0.718**	0.414	0.648	0.531	**1.000**
CoCoA$_{PPL}$	0.310	0.579	0.717	0.415	0.644	0.530	**1.000**
AdditiveCoCoA$_{MTE}$	0.253	0.554	0.634	0.383	0.630	0.523	0.997
FullSampleCoCoA$_{MTE}$	0.237	0.502	0.554	0.383	0.636	0.519	0.989
CoCoA$_{MTE}$	0.289	0.551	0.678	0.402	0.646	0.517	0.998

Table 8: Comparison of PRRs of CoCoA-family methods with alternative formulations of with best sample taken in isolation.

C Detailed Description of Uncertainty Quantification Methods

In this section, we provide a detailed description of the uncertainty quantification methods used in this study.

C.1 Information-Based Methods

Information-based methods are commonly used to estimate uncertainty by analyzing the probability distributions of tokens within a given output. These methods examine different levels of model generation, such as the model's confidence in producing a specific sequence, its ability to predict individual tokens at each generation step, and the variability in the token-level predictions across the sequence.

Maximum Sequence Probability (MSP) is one of the simplest and most direct methods for estimating uncertainty. It measures the probability of the most likely output sequence given a specific input. Thus, uncertainty is quantified by calculating the probability of the sequence with the highest likelihood, under the assumption that the model is most confident in this output. It is defined as:

$$U_{MSP}(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}) = -\log P(\mathbf{y} \mid \mathbf{x}). \tag{10}$$

Perplexity (PPL) is another widely used metric for estimating uncertainty in language models (Fomicheva et al., 2020). It measures the model's confidence by evaluating the average likelihood of generating the sequence tokens:

$$U_{\mathrm{PPL}}(\mathbf{y}, \mathbf{x}) = -\frac{1}{L} \log P(\mathbf{y} \mid \mathbf{x}). \tag{11}$$

Mean Token Entropy takes a broader view of uncertainty by considering the token-level predictions across the entire sequence (Fomicheva et al., 2020). Instead of evaluating the model's confidence in a single output or individual token predictions, Mean Token Entropy calculates the average entropy of the token probability distributions for each token in the sequence:

$$U_{\mathcal{H}_T}(\mathbf{y}, \mathbf{x}) = \frac{1}{L} \sum_{l=1}^{L} \mathcal{H}(y_l \mid \mathbf{y}_{<l}, \mathbf{x}), \tag{12}$$

where $\mathcal{H}(y_l \mid \mathbf{y}_{<l}, \mathbf{x})$ is an entropy of the token distribution $P(y_l \mid \mathbf{y}_{<l}, \mathbf{x})$.

The *TokenSAR* method, introduced in Duan et al. (2024), generalizes length-normalized log probability by computing a weighted average of the negative log probabilities of generated tokens, where weights are based on token relevance to the overall text. Using a similarity function $g(\cdot, \cdot)$ and token relevance function $R_T(y_k, \mathbf{y}, \mathbf{x}) = 1 - g(\mathbf{x} \cup \mathbf{y}, \mathbf{x} \cup \mathbf{y} \setminus y_k)$, the uncertainty estimate is calculated as:

$$U_{\mathrm{TokenSAR}}(\mathbf{y}, \mathbf{x}) = -\sum_{l=1}^{L} \tilde{\mathrm{R}}_T(y_l, \mathbf{y}, \mathbf{x}) \log P(y_l \mid \mathbf{y}_{<l}, \mathbf{x}), \tag{13}$$

where

$$\tilde{\mathrm{R}}_T(y_k, \mathbf{y}, \mathbf{x}) = \frac{\mathrm{R}_T(y_k, \mathbf{y}, \mathbf{x})}{\sum_{l=1}^{L} \mathrm{R}_T(y_l, \mathbf{y}, \mathbf{x})}. \tag{14}$$

This measure is central for computing *SAR* uncertainty measure.

C.2 Consistency-Based Methods

Consistency-based methods assess the uncertainty of a language model by evaluating the semantic consistency of its predictions across multiple outputs for the same prompt. The core idea is that semantically similar outputs indicate higher confidence, while diverse or conflicting outputs suggest greater uncertainty. Since language models can express the same meaning in different surface forms, these methods construct a semantic similarity matrix $G = (g_{ij})$, where each entry represents the degree of similarity between pairs of responses. By clustering responses into groups with equivalent meanings, these methods provide a semantic measure of the model's consistency.

Lin et al. (2024) offers two similarity measures to evaluate the similarity of sequences. The first is the Jaccard similarity, which treats sequences as sets of words and calculates the proportion of shared words to the total number of unique words in both sequences: $g(\mathbf{y}, \mathbf{y}') = |\mathbf{y} \cap \mathbf{y}'|/|\mathbf{y} \cup \mathbf{y}'|$.

Natural Language Inference (NLI) provides another method for computing similarity between sequences. We use the DeBERTa-large NLI model He et al. (2021), following Kuhn et al. (2023). For each pair of sequences, an NLI model predicts two probabilities: $p_{\text{entail}}(\mathbf{y}, \mathbf{y}')$, indicating entailment, and $p_{\text{contra}}(\mathbf{y}, \mathbf{y}')$, indicating contradiction. Similarity is then defined as either $g_{\text{entail}}(\mathbf{y}, \mathbf{y}') = p_{\text{entail}}(\mathbf{y}, \mathbf{y}')$ or $g_{\text{contra}}(\mathbf{y}, \mathbf{y}') = 1 - p_{\text{contra}}(\mathbf{y}, \mathbf{y}')$.

Among the simplest consistency-based approaches are the *Number of Semantic Sets* and the *Sum of Eigenvalues of the Graph Laplacian* (Lin et al., 2024). *Number of Semantic Sets* estimates how many distinct "meanings" the model produces by clustering its outputs with an NLI model. The number of semantic sets is initially equal to the total number of generated answers, M. Two sentences are grouped into the same cluster if the following conditions are satisfied: $p_{\text{entail}}(\mathbf{y}^i, \mathbf{y}^j) > p_{\text{contra}}(\mathbf{y}^i, \mathbf{y}^j)$ and $p_{\text{entail}}(\mathbf{y}^j, \mathbf{y}^i) > p_{\text{contra}}(\mathbf{y}^j, \mathbf{y}^i)$. This computation is performed for all pairs of answers, and the final number of distinct clusters is denoted by $U_{\text{NumSemSets}}$.

Sum of Eigenvalues of the Graph Laplacian examines global diversity: it constructs a similarity matrix among the sampled outputs and computes a continuous uncertainty score from the eigenvalues of the Laplacian of that similarity graph. Lin et al. (2024) proposes computing an averaged similarity matrix as $g_{ij} = \big(g(\mathbf{y}^{(i)}, \mathbf{y}^{(j)}) + g(\mathbf{y}^{(j)}, \mathbf{y}^{(i)})\big)/2$. The Laplacian for the matrix G is defined as $L = I - D^{-\frac{1}{2}} G D^{-\frac{1}{2}}$, where D is a diagonal matrix with elements $D_{ii} = \sum_{j=1}^{M} g_{ij}$. Consequently, the following formula is derived:

$$U_{\text{EigV}} = \sum_{i=1}^{M} \max(0, 1 - \lambda_i). \tag{15}$$

Both *Number of Semantic Sets* and *Sum of Eigenvalues of the Graph Laplacian* effectively capture overall variation in generated text but cannot produce an individual uncertainty score for each output. To address this, Lin et al. (2024) proposes to use the diagonal *Degree Matrix* D which represents the total similarity of each answer with all others. The corrected trace of D provides an average pairwise distance between answers, and uncertainty is computed as:

$$U_{\text{DegMat}} = 1 - \text{trace}(D)/M^2. \tag{16}$$

C.3 Information-Based Methods with Repeated Sampling

In this section we detail methods that integrate model confidence with consistency.

We can compute the entropy on the sequence level $\mathbb{E}\big[-\log P(\mathbf{y} \mid \mathbf{x})\big]$, where the expectation is taken over the sequences $\mathbf{y}$ randomly generated from the distribution $P(\mathbf{y} \mid \mathbf{x})$. Unfortunately, while for token level, we have an exact way of computing the entropy, for the sequence level, we need to adhere to some approximations. In practice, we can use Monte-Carlo integration, i.e. generate several sequences $\mathbf{y}^{(i)}$, $i = 1, \ldots, M$ via random sampling and compute *Monte Carlo Sequence Entropy*:

$$U_{\mathcal{H}_S}(\mathbf{x}) = -\frac{1}{M} \sum_{i=1}^{M} \log P(\mathbf{y}^{(i)} \mid \mathbf{x}). \tag{17}$$

We can replace $P(\mathbf{y}^{(i)} \mid \mathbf{x})$ with its length-normalized version $\bar{P}(\mathbf{y}^{(i)} \mid \mathbf{x})$ leading to a more reliable uncertainty measure in some cases.

Semantic Entropy Kuhn et al. (2023) addresses the issue of generated sequences with similar meanings but differing probabilities according to the model, which can heavily influence the resulting entropy value equation 17. The method clusters generated sequences $\mathbf{y}^{(i)}$, $i = 1, \ldots, M$ into semantically homogeneous groups $\mathcal{C}_k$, $k = 1, \ldots, K$ (where $K \leq M$) using a bi-directional entailment algorithm. Probabilities of sequences are averaged within each cluster. The entropy estimate is then defined as:

$$U_{\text{SE}}(\mathbf{x}) = -\sum_{k=1}^{K} \frac{|\mathcal{C}_k|}{M} \log \hat{P}_k(\mathbf{x}), \tag{18}$$

where $\hat{P}_k(\mathbf{x}) = \sum_{\mathbf{y} \in \mathcal{C}_k} P(\mathbf{y} \mid \mathbf{x})$ represents the aggregated probability for cluster $\mathcal{C}_k$.

SentenceSAR Duan et al. (2024) enhances the probability of sentences that are more relevant. It uses a sentence relevance measure $g\big(\mathbf{y}^{(j)}, \mathbf{y}^{(k)}\big)$ to evaluate the relevance of $\mathbf{y}^{(j)}$ with respect to $\mathbf{y}^{(k)}$. SentenceSAR is calculated as:

$$U_{\mathrm{SentSAR}}(\mathbf{x}) = -\frac{1}{M} \sum_{i=1}^{M} \log\Big(P(\mathbf{y}^{(i)} \mid \mathbf{x}) + \frac{1}{t} \mathrm{R}_S(\mathbf{y}^{(i)}, \mathbf{x}) \Big), \tag{19}$$

where t is a temperature parameter used to control the scale of shifting to relevance, and

$$\mathrm{R}_S(\mathbf{y}^{(j)}, \mathbf{x}) = \sum_{k \neq j} g\big(\mathbf{y}^{(j)}, \mathbf{y}^{(k)}\big) P\big(\mathbf{y}^{(k)} \mid \mathbf{x}\big). \tag{20}$$

The combination of SentenceSAR and TokenSAR results in a unified method called ***SAR*** Duan et al. (2024). In this approach, the generative probability $P(\mathbf{y} \mid \mathbf{x})$ in the SentenceSAR formula is replaced with the token-shifted probability $P'(\mathbf{y} \mid \mathbf{x}) = \exp\big\{-\mathrm{TokenSAR}(\mathbf{y}, \mathbf{x})\big\}$, creating a comprehensive measure that integrates both sentence- and token-level adjustments.

D Detailed Description of Evaluation Metric

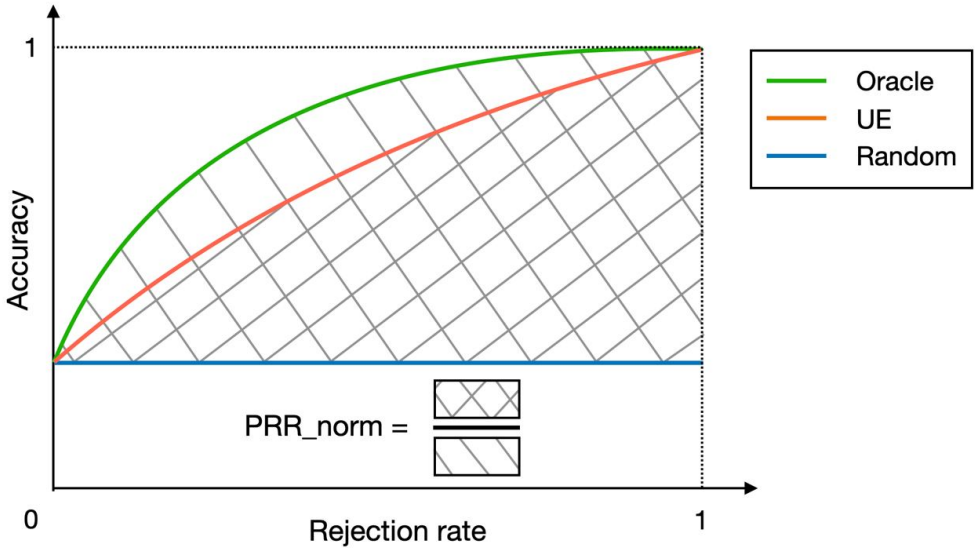

Figure 3: Prediction-Rejection Ratio (PRR) Curve. The curve illustrates the quality of the non-rejected predictions as a function of the rejection rate. *Oracle* represents the optimal rejection strategy, *Random* is a random rejection, and *UE* is rejection based on the evaluated uncertainty estimation method.

PRR operates by progressively rejecting predictions with uncertainty scores above a threshold a and observing how the average quality $Q(f(\mathbf{x}_i), \mathbf{y}_i)$ of the remaining predictions changes. The metric is calculated as the ratio of two areas: the area between the Prediction Rejection (PR) curves for the evaluated uncertainty score and a random baseline, and the area between the oracle (the ideal uncertainty score that perfectly ranks instances by quality) and the random baseline. Formally, PRR is defined as follows:

$$PRR = \frac{\text{AUC}_{\text{unc}} - \text{AUC}_{\text{rnd}}}{\text{AUC}_{\text{oracle}} - \text{AUC}_{\text{rnd}}}. \tag{21}$$

Higher PRR values indicate better alignment of uncertainty scores with prediction quality, approaching the performance of an oracle. To ensure practical applicability, we compute PRR only up to a rejection threshold of 50%, preventing cases where excessive rejection artificially inflates quality measures. Figure 3 gives a visual representation of the PRR calculation, highlighting the relationship between the uncertainty threshold and the quality measures.

E RESULTS FOR EVALUATED SEQUENCE - GREEDY SEQUENCE

Metric	Llama			Mistral			Falcon		
	QA	**MT**	**SUM**	**QA**	**MT**	**SUM**	**QA**	**MT**	**SUM**
MCSE	0.310	0.323	0.033	0.389	0.304	0.007	0.414	0.317	0.159
MCNSE	0.309	0.393	0.022	0.384	0.410	0.009	0.405	0.422	0.108
Semantic Entropy	0.356	0.343	0.033	0.423	0.327	0.008	0.439	0.348	0.164
SAR	0.414	0.455	0.077	0.462	0.435	0.094	0.481	0.458	0.144
DegMat	0.406	0.302	0.081	0.423	0.305	0.137	0.483	0.353	0.201
EigValLaplacian	0.375	0.238	0.079	0.391	0.267	0.132	0.459	0.312	0.201
MSP	0.409	0.399	0.328	0.475	0.383	0.287	0.475	0.356	0.201
CoCoA$_{MSP}$	<u>0.451</u> ↑	**0.519** ↑	0.378 ↑	**0.509** ↑	**0.497** ↑	**0.330** ↑	0.511 ↑	0.505 ↑	**0.257** ↑
PPL	0.381	0.386	0.369	0.424	0.427	0.204	0.456	0.450	0.155
CoCoA$_{PPL}$	**0.454** ↑	<u>0.481</u> ↑	**0.387** ↑	<u>0.494</u> ↑	<u>0.472</u> ↑	0.286 ↑	<u>0.523</u> ↑	<u>0.508</u> ↑	<u>0.229</u> ↑
MTE	0.353	0.382	0.357	0.417	0.438	0.182	0.456	0.473	0.152
CoCoA$_{MTE}$	0.447 ↑	0.478 ↑	<u>0.380</u> ↑	0.492 ↑	0.469 ↑	<u>0.288</u> ↑	**0.527** ↑	**0.508** ↑	0.228 ↑

Table 9: Results for Evaluated Sequence – Greedy Sample: Mean PRR across datasets for each task. The best-performing method is shown in bold, and the second-best is underscored. Arrows indicate improvement in CoCoA over the base version.

F Detailed Experimental Results

In this section, we present detailed experimental results, which were used for computing values in Tables 1 and 9.

Method	**Dataset**						
	XSum	WMT14FrEn	WMT19DeEn	CoQa	Trivia	MMLU	GSM8k
Mistral7b-Base							
MCSE	0.007	0.257	0.350	0.247	0.496	0.337	0.475
MCNSE	0.009	0.342	0.478	0.238	0.540	0.356	0.401
Semantic Entropy	0.008	0.271	0.382	0.271	0.562	0.387	0.472
SAR	0.094	0.353	0.517	0.313	0.644	0.419	0.471
DegMat	0.137	0.229	0.382	0.336	0.646	0.410	0.299
EigValLaplacian	0.132	0.207	0.328	0.301	0.624	0.398	0.241
MSP	0.287	0.315	0.451	0.326	0.628	**0.474**	0.471
CoCoA$_{MSP}$	**0.330**	**0.396**	**0.598**	**0.383**	0.670	0.466	**0.517**
PPL	0.204	0.365	0.489	0.281	0.632	**0.474**	0.311
CoCoA$_{PPL}$	0.286	0.375	0.568	0.369	**0.674**	0.466	0.467
MTE	0.182	0.392	0.484	0.243	0.619	0.456	0.350
CoCoA$_{MTE}$	0.288	0.374	0.564	0.355	0.673	0.447	0.491
Llama8b-Base							
MCSE	0.033	0.293	0.354	0.237	0.482	0.171	0.351
MCNSE	0.022	0.370	0.415	0.219	0.501	0.170	0.344
Semantic Entropy	0.033	0.297	0.389	0.272	0.549	0.229	0.375
SAR	0.077	0.427	0.483	0.311	0.595	0.352	0.398
DegMat	0.081	0.250	0.355	**0.353**	**0.622**	0.342	0.309
EigValLaplacian	0.079	0.198	0.278	0.332	0.604	0.292	0.273
MSP	0.328	0.342	0.456	0.277	0.526	**0.508**	0.324
CoCoA$_{MSP}$	0.378	**0.456**	**0.582**	0.349	0.597	0.485	0.372
PPL	0.369	0.351	0.422	0.253	0.507	0.461	0.303
CoCoA$_{PPL}$	**0.387**	0.448	0.514	0.338	0.593	0.452	0.433
MTE	0.357	0.357	0.408	0.239	0.497	0.350	0.326
CoCoA$_{MTE}$	0.380	0.446	0.511	0.337	0.601	0.402	**0.447**
Falcon7b-Base							
MCSE	0.159	0.297	0.337	0.258	0.549	0.420	0.427
MCNSE	0.108	0.371	0.474	0.293	0.586	0.442	0.299
Semantic Entropy	0.164	0.307	0.389	0.294	0.581	0.463	0.418
SAR	0.144	0.398	0.517	0.381	0.649	0.508	0.387
DegMat	0.201	0.274	0.431	**0.407**	0.651	0.480	0.395
EigValLaplacian	0.201	0.229	0.394	0.381	0.645	0.454	0.358
MSP	0.201	0.312	0.400	0.321	0.662	**0.539**	0.377
CoCoA$_{MSP}$	**0.257**	0.433	0.578	0.396	0.684	0.529	0.436
PPL	0.155	0.375	0.525	0.316	0.644	**0.539**	0.326
CoCoA$_{PPL}$	0.229	0.436	**0.580**	0.406	0.677	0.529	0.478
MTE	0.152	0.409	0.537	0.291	0.633	0.533	0.367
CoCoA$_{MTE}$	0.228	**0.439**	0.577	0.395	**0.685**	0.517	**0.510**

Table 10: Detailed experimental results with greedy sample taken in isolation.

Method	**Dataset**						
	XSum	WMT14FrEn	WMT19DeEn	CoQa	Trivia	MMLU	GSM8k
Mistral7b-Base							
MCSE	0.162	0.406	0.407	0.289	0.492	0.339	0.693
MCNSE	0.196	0.471	0.507	0.277	0.529	0.358	0.700
Semantic Entropy	0.164	0.434	0.442	0.312	0.554	0.389	0.675
SAR	0.175	0.563	0.590	0.347	0.620	0.421	0.780
DegMat	0.205	0.439	0.410	0.376	0.618	0.410	0.454
EigValLaplacian	0.197	0.388	0.344	0.342	0.600	0.399	0.361
MSP	0.330	0.212	0.291	0.388	0.607	**0.476**	0.307
CoCoA$_{MSP}$	0.366	<u>0.712</u>	0.730	**0.430**	0.644	0.466	0.562
PPL	0.365	0.695	0.676	0.327	0.615	**0.476**	0.931
CoCoA$_{PPL}$	<u>0.372</u>	**0.735**	**0.755**	<u>0.402</u>	**0.648**	0.466	**0.937**
MTE	0.350	0.668	0.606	0.254	0.594	0.457	0.932
CoCoA$_{MTE}$	**0.373**	0.708	<u>0.732</u>	0.373	<u>0.645</u>	0.447	<u>0.935</u>
Llama8b-Base							
MCSE	0.192	0.366	0.395	0.259	0.465	0.158	0.546
MCNSE	0.186	0.377	0.480	0.239	0.484	0.165	0.634
Semantic Entropy	0.194	0.371	0.451	0.286	0.528	0.213	0.559
SAR	0.159	0.441	0.571	0.327	0.578	0.340	0.668
DegMat	0.191	0.274	0.409	<u>0.367</u>	**0.606**	0.320	0.397
EigValLaplacian	0.190	0.216	0.333	0.340	<u>0.587</u>	0.274	0.351
MSP	<u>0.464</u>	0.339	0.413	0.304	0.514	**0.483**	0.281
CoCoA$_{MSP}$	**0.484**	<u>0.529</u>	<u>0.685</u>	**0.384**	<u>0.587</u>	<u>0.452</u>	0.513
PPL	0.458	0.504	0.622	0.294	0.483	0.441	<u>0.912</u>
CoCoA$_{PPL}$	0.450	**0.544**	**0.689**	0.364	0.573	0.422	**0.925**
MTE	0.449	0.437	0.501	0.239	0.458	0.326	0.884
CoCoA$_{MTE}$	0.451	0.520	0.638	0.346	0.582	0.363	0.900
Falcon7b-Base							
MCSE	0.128	0.399	0.419	0.285	0.535	0.421	0.598
MCNSE	0.153	0.395	0.452	0.318	0.588	0.443	0.771
Semantic Entropy	0.134	0.420	0.460	0.319	0.566	0.463	0.567
SAR	0.193	0.455	0.521	0.385	0.642	0.509	0.826
DegMat	0.177	0.350	<u>0.422</u>	0.422	0.637	0.480	0.633
EigValLaplacian	0.174	0.289	0.382	0.393	0.622	0.454	0.522
MSP	0.099	0.385	0.378	0.369	0.638	**0.540**	-0.175
CoCoA$_{MSP}$	0.210	<u>0.564</u>	<u>0.698</u>	**0.428**	**0.659**	0.530	0.498
PPL	0.275	0.541	0.637	0.353	0.614	**0.540**	**1.000**
CoCoA$_{PPL}$	**0.310**	**0.579**	**0.717**	0.415	0.644	0.530	**1.000**
MTE	0.186	0.475	0.510	0.317	0.573	0.534	0.984
CoCoA$_{MTE}$	<u>0.289</u>	0.551	0.678	0.402	<u>0.646</u>	0.517	0.998

Table 11: Detailed experimental results with best sample taken in isolation.