# CLARA: CLARIFICATION-DRIVEN MEASUREMENT OF INPUT AMBIGUITY IN LLMS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Large Language Models (LLMs) perform well on question-answering tasks with well-specified inputs, but real-world queries are often vague or underspecified, leading to ambiguity and unreliable responses. Existing methods for ambiguity detection typically use a two-stage framework: (a) generating multiple clarifying reformulations of the input, and (b) answering each version to assess ambiguity based on the variation in responses. We introduce CLARA, a novel and complementary approach that quantifies ambiguity using only the clarification generation phase. We hypothesize that ambiguous inputs elicit a greater number and diversity of clarifications. CLARA estimates ambiguity by measuring the semantic dispersion of these LLM-generated clarifications, without requiring subsequent answering. This method requires no additional task-specific training, relying instead on an off-the-shelf similarity model, and thus offers two key benefits: (1) it is lightweight—reducing API calls and computational cost, and (2) it is more robust across LLMs—avoiding dependence on model-specific factual knowledge and reducing susceptibility to hallucinations. Empirical results across multiple LLMs and benchmark datasets demonstrate that CLARA provides an intuitive, scalable, and effective alternative to answer-based techniques, achieving comparable or superior performance.

## 1 INTRODUCTION

Large language models (LLMs) have achieved impressive performance in both open- and closed-domain question-answering tasks when presented with well-specified inputs Chung et al. (2024); Hoffmann et al. (2022); Pan et al. (2023). Their ability to retrieve or synthesize accurate information is largely enabled by the vast semantic knowledge they encode. However, in real-world applications, user queries are often vague or under-specified—failing to convey sufficient detail to elicit a precise answer. For instance, a query such as "When did he visit Australia?" leaves the referent ambiguous, while a query such as "How can I get a lift?" can be interpreted in multiple ways—ranging from requesting a ride to seeking emotional or physical assistance—depending on the user's intent.

Such ambiguity poses a significant challenge for LLMs, as they often respond based on a single, most probable interpretation rather than seeking clarification, which can lead to incorrect or unfaithful outputs Jiang et al. (2021); Liao et al. (2023). This not only degrades performance but also undermines user trust and reliability in downstream applications. Prior work has investigated methods for identifying ambiguous inputs and generating clarifying sub-questions Kuhn et al. (2022); Deng et al. (2023); Cole et al. (2023), as well as purely detecting ambiguity Hou et al. (2024); Kuhn et al. (2023); Tian et al. (2023); Shi et al. (2025). A dominant line of research in ambiguity detection relies on output variation, estimating ambiguity by quantifying disagreement across answers to different disambiguated reformulations of the input. While these answer-based methods implicitly depend on clarification diversity, they focus primarily on the variation in final model responses, potentially overlooking the semantic structure and diversity within the clarifications themselves. In contrast, our approach, CLARA, treats clarification generation not as a mere intermediate step, but as a primary signal, directly measuring interpretive dispersion without relying on answer generation. This results in a more lightweight, interpretable, and robust measure of input ambiguity.

In this paper, we propose CLARA, a novel and complementary perspective that shifts the focus from the answer space to the clarification space. We hypothesize that ambiguous inputs elicit a greater number and diversity of clarifications when processed by LLMs. Building on this insight, we introduce a lightweight method that quantifies ambiguity by measuring the semantic dispersion of model-generated clarifications. Crucially, CLARA requires no additional task-specific training: it leverages an existing similarity model to score clarifications, but does not involve supervised fine-tuning for ambiguity detection. This design makes CLARA significantly more efficient than existing answer-based techniques and reduces dependence on parametric model knowledge, which often introduces hallucinations and inconsistency across LLMs.

Through extensive evaluations across multiple LLMs and benchmark datasets, we demonstrate that the diversity of clarifications provides a strong and interpretable signal of input ambiguity. Our results show that this clarification-based signal enables reliable distinction between ambiguous and unambiguous queries, performing comparably to or better than existing answer-based baselines, and offering a scalable, robust alternative for ambiguity detection in LLMs.

## 2 Previous Literature

Ambiguity remains a persistent challenge in natural language processing, manifesting across tasks such as syntactic and semantic parsing Koller et al. (2008), open-domain and conversational question answering Min et al. (2020); Cole et al. (2023); Guo et al. (2021), and natural language inference Liu et al. (2023). Prior work has addressed this issue through both mitigation and detection strategies. For example, AmbigQA Min et al. (2020) introduced a benchmark to evaluate models on ambiguous questions, showing that standard LLMs often fail to recognize or resolve ambiguity without explicit guidance.

Several recent studies focus on detecting ambiguity at the input level. Hou et al. Hou et al. (2024) approach the problem through uncertainty quantification, using aleatoric uncertainty in LLM answers as an indicator of potential ambiguity. Kuhn et al. Kuhn et al. (2023) similarly compute output entropy as a proxy for input uncertainty. Tian et al. Tian et al. (2023) propose eliciting a model's ambiguity judgment via scalar scores, a method adapted by Hou et al. to measure LLM confidence. Shi et al. Shi et al. (2025) study context-dependent ambiguity, analyzing how the surrounding context affects interpretability, while Piryani et al. Piryani et al. (2024) focus on temporal ambiguity in questions.

While existing methods quantify ambiguity by measuring variability in generated answers, we propose a different yet complementary approach. Our method, CLARA, assesses ambiguity directly from the clarification space produced by an LLM. Rather than analyzing output diversity, CLARA leverages both the number and semantic diversity of clarification questions generated in response to a given input. This perspective captures the model's interpretive uncertainty–the range of plausible interpretations it identifies–without requiring it to resolve them.

## 3 Methodology

### 3.1 Motivation

The general ambiguity classification pipeline typically involves two stages: (1) clarified question generation, and (2) clarified question answering. In the first stage, multiple disambiguating reformulations of the initial question are generated. In the second, each clarified version is answered. Existing approaches assess ambiguity by leveraging the self-consistency of answers, based on the hypothesis that ambiguous inputs yield divergent outputs due to underspecified intent. While this approach is valid, it focuses exclusively on output variation, neglecting the clarification space itself.

Figures 2a and 2b illustrate how ambiguity manifests in the behavior of LLMs during clarification of questions from the AmbigQA dataset Min et al. (2020). Specifically, LLMs tend to generate more clarifications for ambiguous questions, and these clarifications are less semantically similar to each other than clarifications for unambiguous inputs. Building on these observations, this work proposes a novel and lightweight method for ambiguity classification that operates solely in the clarification space—quantifying ambiguity based on the number and semantic diversity of generated clarifications, without requiring answers to them. Figure 1 summarises our approach and compares it to the predominant paradigm in this research direction.

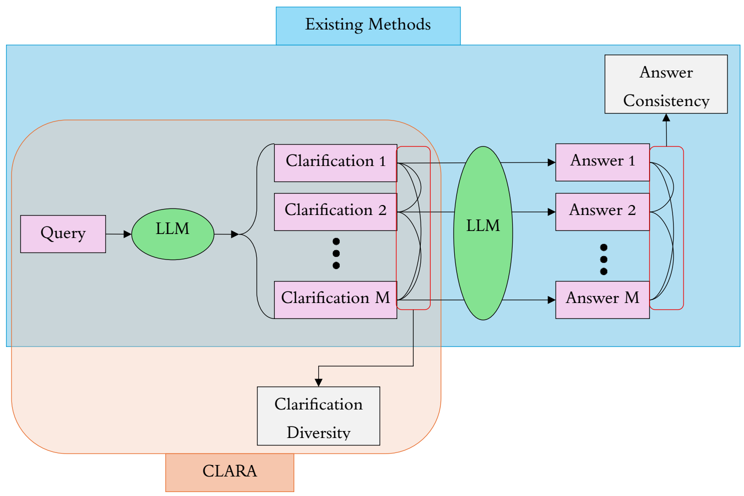
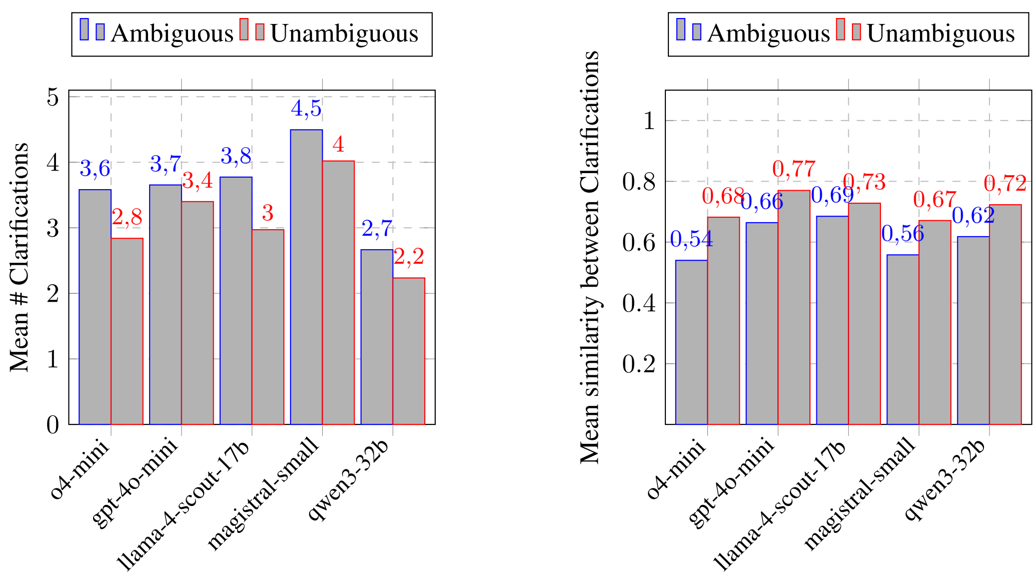



Figure 1: Conceptual comparison between answer-based ambiguity detection methods and our proposed CLARA framework. Traditional approaches estimate ambiguity by generating multiple clarifications, answering each using an LLM, and measuring answer-level divergence. In contrast, CLARA bypasses the answering stage and quantifies ambiguity directly from the semantic dispersion of generated clarifications. In each prompting run, the LLM produces a batch of $M$ clarifications per query.




(a) Average number of clarifications per LLM for ambiguous and unambiguous questions from the AmbigQA dataset Min et al. (2020).

(b) Average similarity between clarifications per LLM for ambiguous and unambiguous questions from the AmbigQA dataset Min et al. (2020).

Figure 2: Clarification Diversity (frequency) and Semantic Dispersion (similarity) of LLMs on Ambiguous vs. Unambiguous Questions (AmbigQA).

### 3.2 Approach

#### 3.2.1 Clarifications Generation

Let $q$ be a query. A query can be a question or an instruction. An LLM is given the query in a prompt $\mathbf{p}(q)$ and is prompted $N$ times to generate multiple clarification batches $C_i^{(q)} = \{c_{ij}^{(q)}\}_j, i \in [1, N]$ for query $q$. The prompt does not specify how many clarifications each batch should contain, but the LLM is explicitly instructed to generate diverse clarifications to avoid redundancy. Figures 4 and 5 show the prompt $\mathbf{p}$ designed for clarification generation in the case of questions and instructions, respectively.

#### 3.2.2 Score Calculation

In this step, the ambiguity of a query is assessed by computing a score that reflects the degree of variation among its generated clarifications. This involves: (a) calculating clarification scores to quantify the diversity within each batch of generated clarifications, and (b) aggregating these scores to obtain an overall ambiguity score for the query. The process is described in more detail below.

*a) Clarification Score*: The score for clarification batch $C_i^{(q)}$ is calculated as follows:

$$g(C_i^{(q)}) = \sum_{(c_{ij}^{(q)}, c_{ij'}^{(q)}) \in C_i^{(q)} \times C_i^{(q)}} \left(1 - \mathrm{sim}(c_{ij}^{(q)}, c_{ij'}^{(q)})\right) \tag{1}$$

where sim denotes a function that computes the pairwise semantic similarity between sentences or questions. A variant of the clarification score design is described in the appendix. Unlike generic embedding-based measures like cosine similarity, which capture lexical or topical proximity, we require a similarity measure that reflects functional equivalence, capturing sentence-level semantics serving as a pragmatic proxy for intent similarity.

We adopt a BERT-based similarity model Devlin et al. (2019), fine-tuned on the Quora Question Pairs (QQP) dataset Sharma et al. (2019), to quantify interpretive variability in LLM-generated clarifications. The QQP dataset, comprising over 400,000 real-world question pairs annotated for semantic equivalence, provides a close match to our objective of distinguishing plausible interpretations of ambiguous queries. The model is trained end-to-end with a binary classification objective, producing a probability that two questions express the same intent; this probability defines our similarity function sim, yielding scores in $[0, 1]$. By leveraging sentence-level semantic representations, the model effectively captures functional similarity and is well suited for measuring semantic dispersion among clarifications, where subtle meaning differences signal underlying ambiguity. Importantly, similarity is computed exhaustively across all ordered clarification pairs, rather than assuming symmetry, as the BERT-based model is inherently asymmetric due to its concatenation-based encoding scheme. Consequently, input order can affect predictions, making it necessary to preserve directionality in similarity computations. It should be noted that training using QQP makes the model well-suited for questions but less directly aligned with instruction-style inputs. While this introduces some domain mismatch, we show empirically that CLARA remains competitive.

Equation 1 defines a clarification-based score that quantifies semantic dispersion within a batch of generated clarifications, serving as a proxy for input ambiguity. A higher score indicates greater interpretive variability, suggesting that the query is underspecified and admits multiple plausible readings, while a lower score reflects stronger semantic similarity and a more constrained interpretive space. Crucially, the metric incorporates both the degree of semantic spread (dispersion) and the number of clarifications (diversity): when average pairwise similarity is fixed, increasing the number of clarifications proportionally raises the score, aligning with the intuition that ambiguous queries elicit a broader range of potential disambiguations. Formulated as the sum of pairwise inverse similarities across all clarifications, the score thus jointly captures diversity and quantity, ensuring monotonic growth with either dimension. This design provides a principled and scalable method for estimating intent uncertainty directly from clarifications, eliminating the need for answer generation.

*b) Query Score*: The overall score of each query is calculated as follows:

$$\mathrm{Score}(q) = \mathrm{Agg}(\{f(g(C_i^{(q)}))\}_{i=1}^{N}) \tag{2}$$

where Agg is an aggregation function used to combine the per-clarification batch scores, and $f$ transforms the clarification score. Here, we define Agg as the mean over the set of clarifications, and $f$ as $f(x) = \log(1 + x)$. The log transform is applied to compress the scale of clarification counts, attenuating the dominance of batches with large clarification counts and preserving the relative significance of batches with lower counts. The use of the mean as an aggregation function ensures that the final question score reflects the average ambiguity across multiple independent clarification attempts, offering robustness to stochasticity in LLM generations.

# 4 Experiments and Results

## 4.1 Datasets

We experimented with two datasets AmbigQA Min et al. (2020) and AmbigInst Hou et al. (2024). Following the setting described in Hou et al. (2024), we use a sample of 200 examples from AmbigQA's validation set, and the full AmbigInst (364 examples).

## 4.2 Models

Various models were used to evaluate the generalizability of CLARA: o4-mini Hurst et al. (2024), gpt-4o-mini Hurst et al. (2024), llama-4-scout-17b Meta.AI (2025), magistral-small Rastogi et al. (2025), qwen3-32b Team (2025). The models were prompted via API calls to different services: OpenAI (for o4-mini, gpt-4o-mini), Groq (for llama-4-scout-17b, qwen3-32b) and Mistral (for magistral-small).

## 4.3 Baselines

We compared CLARA to various approaches from the literature, including: Ask4conf Tian et al. (2023), which is a technique based on eliciting the verbal confidence of the LLM in its answer. In the context of ambiguity detection, Ask4conf asks the LLM for the confidence of the ambiguity of the input Hou et al. (2024). Aleatoric Uncertainty (AU) and Total Uncertainty (TU) (combining both aleatoric and epistemic uncertainties) were introduced in Hou et al. (2024). These measures decompose LLM uncertainty by generating a set of clarifications for the input, passing them through the model, and aggregating the resulting predictions via ensembling.

## 4.4 Experimental Setting

We used a model trained on the QQP dataset to compute the similarity between clarifications. The model is available on Huggingface[1]. Most of the experiments were conducted in a Colab notebook using API calls to OpenAI, Groq, and Mistral. For AmbigQA, we used the training split as an external database to retrieve the most similar ambiguous and unambiguous questions along with their corresponding clarifications. These examples were used in an 8-shot prompt. Similar to Hou et al. (2024), we opted not to use few-shot examples in the AmbigInst prompts due to the simplicity of its instructions. We report both the area under the receiver operating characteristic curve (AUROC) and the best F1 score for the baselines and CLARA on the two datasets. To assess the consistency of the various approaches, each experiment is conducted over five independent runs. For each approach, we report the mean and standard deviation of the results from these runs. The temperature is set to 1 for all models except gpt-4o-mini, for which it is set to 0.5. This choice of temperature is based on manual verification. We set the number of clarification batches for CLARA to $N = 5$. This was initially motivated by experimenting on one model for one run. We later found that CLARA's performance is consistently saturated after $N = 5$.

## 4.5 Results

Table 1 presents the experimental results for ambiguity detection in questions from the AmbigQA dataset and instructions from the AmbigInst dataset.

[1]https://huggingface.co/rambodazimi/bert-base-uncased-finetuned-FFT-QQP

***1) Question Ambiguity Detection*** Comparing CLARA with various baselines (AU, TU, Ask4conf) across five different LLMs. Both AUROC and F1 scores are reported. CLARA consistently performs at or near the top across models, demonstrating its effectiveness and generalizability.

Notably, CLARA achieves the best overall performance with the o4-mini model, yielding an AUROC of 0.727$\pm$ 0.014 and an F1 score of 0.736$\pm$ 0.009—the highest values in the entire table. This demonstrates CLARA's ability to effectively identify ambiguous inputs while maintaining strong classification performance. The next-best results for o4-mini (AUROC: 0.643 from Ask4conf; F1: 0.688 from TU) reveal a substantial performance gap, suggesting that CLARA leverages clarification diversity more effectively than baselines based on answer consistency or confidence elicitation.

Across other models, CLARA remains competitive. For instance, in the gpt-4o-mini and magistral-small settings, it outperforms all baselines on both AUROC and F1, with statistically significant margins, underscoring its robustness across different LLM architectures. Even when CLARA is not the top-performing method—for example, with llama-4-scout-17b, where AU achieves a slightly higher AUROC (0.672 vs. 0.645)—its performance remains close, and its F1 score (0.687) is still competitive.

CLARA exhibits low to moderate standard deviation. These results suggest that CLARA is consistently reliable across different models, with relatively small performance fluctuations. Notably, for the o4-mini model—where CLARA performs best overall—std values are particularly low (0.014 AUROC), indicating high robustness in its ambiguity estimates even under generation randomness.

These results support the intuition behind CLARA: ambiguous queries tend to elicit more semantically dispersed clarification questions, which can be leveraged to detect ambiguity without relying on answer generation or ensembling.

***2) Instruction Ambiguity Detection*** Unlike questions, which typically seek discrete, factual answers, instructions often involve intent ambiguity–such as unclear goals, constraints, or actions. This distinction is crucial when evaluating methods like CLARA, which operate by quantifying the semantic dispersion of clarifications generated by an LLM.

CLARA achieves its best performance with o4-mini, the most capable model in the benchmark. It records an AUROC of 0.891$\pm$ 0.006 and an F1 score of 0.847$\pm$ 0.008, both the highest overall. Comparing this result with the model's ambiguity elicitation capability (Ask4conf), we find that even highly capable LLMs, when paired with CLARA, benefit from clarification-driven ambiguity estimation–suggesting that ambiguity is not always fully resolved by model scale or instruction-following ability alone.

However, CLARA's performance drops slightly on other models, often ranking second to answer-based methods like TU. For instance, with Magistral-small and Qwen3-32B, TU achieves higher AUROC and F1 scores. A key factor behind this discrepancy lies in CLARA's reliance on a BERT-based similarity model trained on the QQP dataset. QQP focuses on determining whether two questions are semantically equivalent, emphasizing intent-based paraphrase matching. While this aligns well with CLARA's objective of detecting interpretive variability, it is less suited to instructions, which often involve procedural, imperative, or goal-oriented phrasing not well represented in QQP.

As a result, when the generated clarifications for instructions differ in task framing rather than linguistic paraphrasing, the QQP-trained similarity model may under-represent the true semantic differences. This limits CLARA's sensitivity to the nuances of instruction ambiguity, particularly on models that generate simpler or less nuanced clarifications. This limitation likely contributes to CLARA's lower performance with models like GPT-4o-mini or Magistral-small, where the clarifications may lack the sophistication needed for the QQP-based model to detect meaningful divergence.

In contrast, o4-mini's superior performance with CLARA may be attributed to the model's ability to generate instruction-like clarifications that still align with the structure of question-based intent comparisons. In other words, o4-mini likely produces clarifications that the QQP-trained BERT model can reliably differentiate–either because they resemble interrogative reformulations or because the model articulates implicit goals and constraints more explicitly. This effectively bridges the representational gap between the QQP training data and the AmbigInst task, enabling CLARA to perform exceptionally well.

| Model | Method | AmbigQA (200 qns) | | AmbigInst (364 qns) | |
|---|---|---|---|---|---|
| | | AUROC | F1 | AUROC | F1 |
| o4-mini | CLARA | **0.727 ± 0.014** | **0.736 ± 0.009** | **0.891 ± 0.006** | **0.847 ± 0.008** |
| | AU | 0.623 ± 0.040 | 0.674 ± 0.007 | 0.712 ± 0.010 | 0.734 ± 0.000 |
| | TU | 0.610 ± 0.026 | 0.688 ± 0.014 | 0.741 ± 0.011 | 0.734 ± 0.000 |
| | Ask4conf | 0.643 ± 0.015 | 0.667 ± 0.000 | 0.771 ± 0.010 | 0.731 ± 0.003 |
| Gpt-4o-mini | CLARA | **0.652 ± 0.024** | **0.690 ± 0.016** | 0.751 ± 0.013 | 0.740 ± 0.013 |
| | AU | 0.574 ± 0.025 | 0.667 ± 0.000 | 0.823 ± 0.010 | 0.802 ± 0.012 |
| | TU | 0.574 ± 0.026 | 0.667 ± 0.000 | **0.824 ± 0.011** | **0.802 ± 0.012** |
| | Ask4conf | 0.574 ± 0.018 | 0.667 ± 0.000 | 0.559 ± 0.003 | 0.729 ± 0.000 |
| Llama-4-scout-17b | CLARA | 0.645 ± 0.016 | 0.687 ± 0.008 | 0.762 ± 0.014 | 0.772 ± 0.010 |
| | AU | **0.672 ± 0.028** | **0.700 ± 0.025** | 0.805 ± 0.006 | 0.781 ± 0.003 |
| | TU | 0.639 ± 0.017 | 0.688 ± 0.017 | **0.807 ± 0.006** | **0.783 ± 0.003** |
| | Ask4conf | 0.560 ± 0.018 | 0.667 ± 0.000 | 0.558 ± 0.006 | 0.729 ± 0.000 |
| Magistral-small | CLARA | **0.658 ± 0.018** | **0.692 ± 0.005** | 0.808 ± 0.013 | 0.783 ± 0.005 |
| | AU | 0.639 ± 0.033 | 0.675 ± 0.010 | 0.831 ± 0.018 | 0.816 ± 0.016 |
| | TU | 0.618 ± 0.023 | 0.674 ± 0.009 | **0.836 ± 0.018** | **0.820 ± 0.015** |
| | Ask4conf | 0.552 ± 0.015 | 0.667 ± 0.000 | 0.605 ± 0.027 | 0.729 ± 0.000 |
| Qwen3-32b | CLARA | **0.652 ± 0.008** | **0.692 ± 0.006** | 0.759 ± 0.016 | 0.746 ± 0.011 |
| | AU | 0.646 ± 0.020 | 0.669 ± 0.004 | **0.823 ± 0.016** | **0.797 ± 0.002** |
| | TU | 0.622 ± 0.025 | 0.673 ± 0.008 | 0.823 ± 0.016 | 0.796 ± 0.018 |
| | Ask4conf | 0.553 ± 0.014 | 0.667 ± 0.000 | 0.665 ± 0.015 | 0.729 ± 0.000 |

Table 1: Mean ± std of AUROC and F1 scores for each model and method on the AmbigQA (200 questions) and AmbigInst (364 questions) datasets. The best performing approach for each model is in bold, the second best is underlined. The best overall across datasets is bold and in blue.

In summary, CLARA's strong performance with o4-mini and its relative weakness on other models can be partially attributed to a mismatch between the similarity model's training domain (questions) and the test domain (instructions). This highlights an important direction for future work: leveraging or fine-tuning similarity models on instruction-focused datasets could further enhance CLARA's performance across models. However, this remains challenging due to the scarcity of large-scale, high-quality datasets for instruction paraphrasing or intent similarity. Nevertheless, even with this domain mismatch, CLARA remains a top performer, validating its core principle–that interpretive diversity among clarifications is a powerful indicator of ambiguity, especially when the LLM is capable of generating semantically rich reformulations.

### 4.6 Computational Complexity

The appendix C.2 provides an empirical estimate of the number of API calls and the number of tokens required for a single run of AU and CLARA. The findings highlight CLARA's significant computational efficiency over AU, requiring only 5 API calls per question compared to AU's 33.9–43.3. This difference arises from CLARA's reliance solely on clarification generation, whereas AU requires additional answer queries and answer standardization. As a result, CLARA is more scalable, cost-effective, and robust for real-time or large-scale ambiguity detection.

### 4.7 On Question Ambiguity Score Design

In addition to the original score, we experimented with another variant by changing $g$, Agg and $f$ in equation 2. The **OQ** (Original-Question weighted) variant modifies the original scoring function to incorporate the relevance of each clarification to the original question. The idea behind this variant is that not all clarifications contribute equally to understanding ambiguity–clarifications that are semantically distant from the original question are more likely to represent noise rather than genuine disambiguation.

| Model | Method | AUROC | F1 |
|---|---|---|---|
| o4-mini | CLARA | 0.727 ± 0.014 | 0.736 ± 0.009 |
| | CLARAOQ | 0.700 ± 0.017 | 0.708 ± 0.012 |
| Gpt-4o-mini | CLARA | 0.652 ± 0.024 | 0.690 ± 0.016 |
| | CLARAOQ | 0.647 ± 0.019 | 0.687 ± 0.012 |
| Llama-4-scout-17b | CLARA | 0.645 ± 0.016 | 0.687 ± 0.008 |
| | CLARAOQ | 0.641 ± 0.017 | 0.689 ± 0.006 |
| Magistral-small | CLARA | 0.658 ± 0.018 | 0.692 ± 0.009 |
| | CLARAOQ | 0.639 ± 0.021 | 0.690 ± 0.006 |
| Qwen3-32b | CLARA | 0.652 ± 0.008 | 0.692 ± 0.006 |
| | CLARAOQ | 0.638 ± 0.009 | 0.675 ± 0.007 |

Table 2: Mean ± std of AUROC and F1 scores for each variant on the AmbigQA dataset (200 questions) for ambiguity detection.

To account for this, the OQ score weights each pairwise clarification dissimilarity by the product of the individual similarities between the original question and each clarification:

$$g_{\mathrm{oq}}(C_i^{(q)}) = \sum_{(c_{ij}^{(q)}, c_{ij'}^{(q)}) \in C_i^{(q)} \times C_i^{(q)}} \alpha_{ij}^{(q)} \cdot \alpha_{ij'}^{(q)} \cdot \left(1 - \mathrm{sim}(c_{ij}^{(q)}, c_{ij'}^{(q)})\right) \tag{3}$$

where

$$\alpha_{ij}^{(q)} = \frac{\mathrm{sim}(q, c_{ij}^{(q)})}{\frac{1}{|C_i^{(q)}|} \sum_{c_{ij}^{(q)} \in C_i^{(q)}} \mathrm{sim}(q, c_{ij}^{(q)})}. \tag{4}$$

This formulation gives more weight to clarification pairs that are strongly grounded in the original question while still differing significantly from one another. In doing so, it aims to reduce the influence of irrelevant or low-quality clarifications, making the ambiguity score more robust to noisy generations. As in the original formulation, we apply a log-transform to temper outlier values, and aggregate across batches using the mean. Table 2 compares the original CLARA method with the CLARAOQ variant, which weights clarification pairs based on their similarity to the original question. While CLARAOQ was designed to reduce the influence of irrelevant or noisy clarifications, it consistently underperforms CLARA across all models on both AUROC and F1 metrics. The performance differences, though modest, are systematic, suggesting that the weighting scheme may unintentionally suppress meaningful semantic variation.

This outcome suggests that the clarifications most informative for ambiguity detection are often those that reinterpret the question in semantically dispersed or distant ways. By prioritizing clarifications that closely resemble the original input, CLARAOQ may constrain the interpretive space and weaken the detectable signal of ambiguity. In contrast, CLARA's unweighted approach preserves the full spectrum of plausible clarifications, enabling more comprehensive and effective ambiguity estimation.

### 4.8 On Prompting Design

We investigate the impact of prompt design on ambiguity detection using the AmbigQA dataset, comparing a baseline prompt with a diversified (see figure 4) variant that explicitly encourages diversity in clarification generation. Our results (see table 5) show that even a minimal prompt modification—the addition of a single diversity-oriented instruction—substantially increases the number and semantic spread of clarifications, leading to improved ambiguity estimation. This effect is particularly beneficial for CLARA, which leverages semantic dispersion among clarifications and consistently outperforms or matches its baseline-prompt counterpart across models, with the strongest gains observed for o4-mini. In contrast, answer-based methods such as AU exhibit limited sensitivity to such prompt modifications, as their ambiguity estimates are primarily driven by variance in final answers. These findings highlight prompt engineering as an effective, lightweight mechanism for steering large language models toward richer, more diverse outputs that enhance downstream zero-shot performance in ambiguity detection without requiring architectural changes or fine-tuning.

### 4.9 Effects of Batch Count

Figure 3 demonstrates how CLARA's predictive performance scales with the number of clarification batches generated per input query across five large language models (LLMs) on the AmbigQA dataset. Results show a consistent upward trend: performance improves with increased clarification diversity (count) and semantic dispersion (spread). The OpenAI o4-mini model achieves the highest AUROC at $N = 10$, reflecting clarifications that are both diverse and well-aligned with underlying intent variability, while models such as gpt-4o-mini, Qwen3-32b, and Llama-4-scout-17b exhibit steeper early gains, indicating that even a small number of batches can yield strong ambiguity signals. For most models, performance saturates around 5–6 batches, suggesting diminishing returns beyond this point. Despite variations in absolute accuracy, the robustness of performance gains across all models highlights CLARA's generalizability, while reinforcing that its effectiveness depends not only on the number of clarifications but also on their semantic dispersion and the inherent quality of the model generating them.

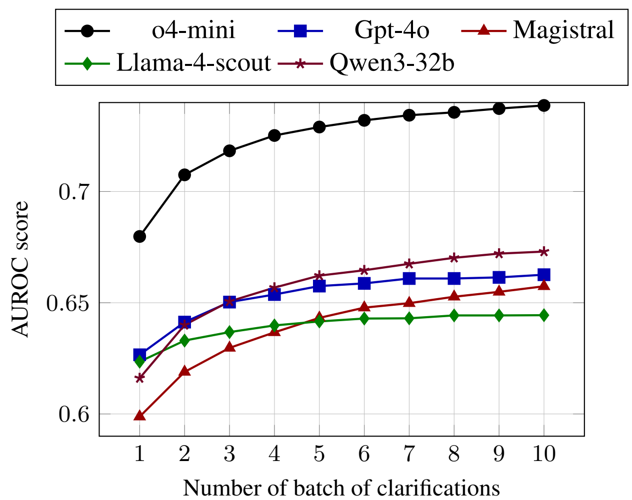



Figure 3: Mean AUROC of CLARA using various LLMs versus the Number of Batches $N$ for the AmbigQA dataset.

## 5 Conclusion

We present CLARA, a clarification-driven framework for estimating input ambiguity in large language models (LLMs) that departs from traditional reliance on answer variability or model confidence by instead leveraging the semantic dispersion of LLM-generated clarifications. This approach is answer-free and requires no additional task-specific training, offering advantages in efficiency, interpretability, and robustness across model architectures and query types. Empirical evaluations on AmbigQA and AmbigInst demonstrate that CLARA matches or surpasses strong baselines, achieving state-of-the-art results with the o4-mini model, while analyses underscore its generalizability and the critical roles of clarification quality and similarity modeling. By conceptualizing ambiguity detection as interpretive variability, CLARA advances the development of more aware and reliable language systems. Future research will explore enhanced similarity modeling for instructions, integration of hybrid uncertainty signals, clarification-aware weighting schemes, and extensions to multimodal and multilingual domains. In doing so, CLARA takes a step toward scalable ambiguity detection that reflects human strategies of resolving uncertainty through clarifications rather than premature answers. Another important avenue for future work is developing or fine-tuning similarity models on instruction-focused datasets. This could reduce the domain mismatch observed with QQP and further improve CLARA's performance on instructional ambiguity.

## REFERENCES

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024.

Jeremy Cole, Michael Zhang, Dan Gillick, Julian Eisenschlos, Bhuwan Dhingra, and Jacob Eisenstein. Selectively answering ambiguous questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 530–543, 2023.

Yang Deng, Lizi Liao, Liang Chen, Hongru Wang, Wenqiang Lei, and Tat-Seng Chua. Prompting and evaluating large language models for proactive dialogues: Clarification, target-guided, and non-collaboration. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 10602–10621, 2023.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Meiqi Guo, Mingda Zhang, Siva Reddy, and Malihe Alikhani. Abg-coqa: Clarifying ambiguity in conversational question answering. In *3rd Conference on Automated Knowledge Base Construction*, 2021.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, pp. 30016–30030, 2022.

Bairu Hou, Yujian Liu, Kaizhi Qian, Jacob Andreas, Shiyu Chang, and Yang Zhang. Decomposing uncertainty for large language models through input clarification ensembling. In *International Conference on Machine Learning*, pp. 19023–19042. PMLR, 2024.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977, 2021.

Alexander Koller, Michaela Regneri, and Stefan Thater. Regular tree grammars as a formalism for scope underspecification. In *Proceedings of ACL-08: HLT*, pp. 218–226, 2008.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Clam: Selective clarification for ambiguous questions with generative language models. *arXiv preprint arXiv:2212.07769*, 2022.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023.

Lizi Liao, Grace Hui Yang, and Chirag Shah. Proactive conversational agents in the post-chatgpt world. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 3452–3455, 2023.

Alisa Liu, Zhaofeng Wu, Julian Michael, Alane Suhr, Peter West, Alexander Koller, Swabha Swayamdipta, Noah A Smith, and Yejin Choi. We're afraid language models aren't modeling ambiguity. *arXiv preprint arXiv:2304.14399*, 2023.

Meta.AI. The Llama 4 herd: The beginning of a new era of natively multimodal AI innovation — ai.meta.com. https://ai.meta.com/blog/llama-4-multimodal-intelligence/, 2025. [Accessed 23-07-2025].

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. Ambigqa: Answering ambiguous open-domain questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5783–5797, 2020.

Haojie Pan, Zepeng Zhai, Hao Yuan, Yaojia Lv, Ruiji Fu, Ming Liu, Zhongyuan Wang, and Bing Qin. Kwaiagents: Generalized information-seeking agent system with large language models. *arXiv preprint arXiv:2312.04889*, 2023.

Bhawna Piryani, Abdelrahman Abdallah, Jamshid Mozafari, and Adam Jatowt. Detecting temporal ambiguity in questions. *arXiv preprint arXiv:2409.17046*, 2024.

Abhinav Rastogi, Albert Q Jiang, Andy Lo, Gabrielle Berrada, Guillaume Lample, Jason Rute, Joep Barmentlo, Karmesh Yadav, Kartik Khandelwal, Khyathi Raghavi Chandu, et al. Magistral. *arXiv preprint arXiv:2506.10910*, 2025.

Lakshay Sharma, Laura Graesser, Nikita Nangia, and Utku Evci. Natural language understanding with the quora question pairs dataset. *arXiv preprint arXiv:1907.01041*, 2019.

Zhengyan Shi, Giuseppe Castellucci, Simone Filice, Saar Kuzi, Elad Kravi, Eugene Agichtein, Oleg Rokhlenko, and Shervin Malmasi. Ambiguity detection and uncertainty calibration for question answering with large language models. In *Proceedings of the 5th Workshop on Trustworthy NLP (TrustNLP 2025)*, pp. 41–55, 2025.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5433–5442, 2023.

# A Appendix

# B Prompts

To elicit diverse and semantically grounded clarifications from the language model, inspired by Hou et al. (2024), we design a prompt that explicitly instructs the model to interpret potential sources of ambiguity in user questions. As illustrated in Figure 4, the prompt begins by listing several common ambiguity types, including vague entity references, unspecified properties, temporal or locational underspecification, and multiple valid answer types. This framing guides the model to consider a broad range of interpretive axes when reformulating a question. The prompt further includes a diversity instruction—highlighted in blue—that encourages the model to enumerate all plausible clarifications, thereby maximizing coverage of the interpretation space. Importantly, the instructions prohibit generating yes/no or disambiguation-seeking questions and instead require direct questions to ensure that each clarification can independently yield a concrete answer. Few-shot examples (in red) are included to prime the model on the desired behavior. These questions are retrieved from the training split of the AmbigQA dataset based on how similar they are to the target question. Table 3 shows examples from the AmbigQA dataset. A question placeholder is used to specify the target query for clarification. This design enables consistent and structured clarification generation, which is central to our approach for quantifying ambiguity in the clarification space.

Figure 5 presents the clarification generation prompt template used for the AmbigInst dataset. Unlike the prompt in figure 4, which focuses on user-generated questions, this prompt targets ambiguity in natural language instructions typical of instruction-tuning datasets. The objective is to analyze whether a given task description—when paired with a specific input—is underspecified, vague, or open to multiple interpretations. The prompt explicitly instructs the model to perform a careful analysis before concluding that the task is unambiguous, highlighting that apparent clarity may break down when considering concrete inputs. If ambiguity is detected, the model is asked to produce all possible disambiguated reformulations of the task, each expressed as a standalone instruction. The structure encourages exhaustive coverage of plausible interpretations while maintaining a consistent output format. This setup ensures that ambiguity is detected and articulated not only at the surface level of the instruction but also in the interaction between instruction and input—thereby supporting a finer-grained understanding of instruction-following ambiguity in language models.

**Clarification Generation Prompt Template for the AmbigQA Dataset.**

In what follows, you will be given some questions that might be ambiguous. These ambiguities can arise from various factors, including but not limited to:

1. Ambiguous references to entities in the question.
2. Multiple properties of objects/entities in the question leading to different interpretations.
3. Ambiguities due to unclear timestamps.
4. Ambiguities stemming from unclear locations.
5. Multiple valid answer types based on the question.

[For an ambiguous question, you need to give every possible clarification so that you can explore every possible interpretation of the question.]

[For each question, you are to provide at least two distinct rephrasings that resolve these ambiguities.]

You should not seek further information or produce a binary (yes-no) question as a result of the clarification. Instead, you must create a direct question (wh-question) that aims to obtain a specific answer.

We're going to give you some examples of possible clarifications. You only need to clarify the last question you're given.

Important: Your output should be *nothing more* than this. Please format your responses as follows (with at least two rephrasings per question):

Clarifications:
1. [First rephrased question]
2. [Second rephrased question]
3. [Third rephrased question]
...

If the original question is already clear and unambiguous, you should indicate this by stating, "No clarification needed."

Now, follow the given examples and clarify the question.
Here are some examples:
{FEWSHOT QUESTIONS WITH EXPECTED ANSWER}
Here is the question you have to clarify:
{QUESTION}

Figure 4: Clarification generation prompt template for the AmbigQA dataset. The blue instruction is a "diversity instruction" that forces the LLM to generate more diverse clarifications to reduce redundancy. The magenta instruction is the regular instruction. The diversity and regular instructions are mutually exclusive, i.e. either one or the other is used. The red text between brackets represents parts that will be replaced by a) examples that demonstrate ambiguity and b) the question that the LLM should generate clarifications for.

# C Performance Analysis

## C.1 *Effects of Different Prompt Designs*

The results in Figures 6 and 7 and Table 5 provide a detailed analysis of how the prompting strategy—specifically the inclusion of a diversity instruction—affects clarification generation and the overall performance of ambiguity detection approaches like CLARA and AU on the AmbigQA dataset. Although the modification is minimal (a single sentence encouraging clarification diversity), it yields non-trivial downstream effects on both the quantity and quality of generated clarifications, and subsequently, on the performance of the ambiguity estimators.

- *Clarification Diversity (count) and Semantic Dispersion (spread)*: Figure 6 shows that the diversified prompt leads to a substantial increase in the number of clarifications per question across all models. The average clarification count rises by approximately 0.4 to 1.3 clarifications, with the most significant gains seen in magistral-small (from 2.84 to 4.1) and llama-4-scout-17b. This confirms the effectiveness of the added instruction in encouraging LLMs to explore a broader space of plausible interpretations for potentially ambiguous inputs.

| Ambiguous Examples |
| --- |
| Question : Where did the church of latter day saints originated ?<br>Clarifications :<br>-In what geographical area did the Church of Latter-day Saints originate ?<br>- With what text did the Church of Latter-day Saints originate ?<br>- Where did the key text of the Church of Latter-day Saints originate ?<br>- With whom did the Church of Latter-day Saints originate ? |
| Question : Total us debt as a percentage of gdp ?<br>Clarifications :<br>-Total us debt as a percentage of gdp at the end of Obama's first presidency ?<br>- Total us debt as a percentage of gdp at the end of Bush's first presidency ?<br>- Total us debt as a percentage of gdp at the end of Bush's second presidency ? |
| Question : Difference between bid and offer in stock market ?<br>Clarifications :<br>-Difference between bid and offer in stock market, except in the case of a market maker ?<br>-Difference between bid and offer in stock market in the case of a market maker ? |
| Question : Who holds the record for games played in the vfl/afl ?<br>Clarifications :<br>-Who holds the record for most career games played in VFL/AFL ?<br>-Who holds the record for most games played and coached in the VFL/AFL ?<br>-Which team holds the record for most games played ? |
| **Unambiguous Examples** |
| Question: What did uk soccer officials use before whistles ?<br>Clarifications : No clarification needed. |
| Question : Where was the first mcdonald's opened outside of the us ?<br>Clarifications : No clarification needed. |
| Question : Who wrote the song the man comes around ?<br>Clarifications : No clarification needed. |
| Question : What is the parent company for all toyota divisions worldwide ?<br>Clarifications : No clarification needed. |

Table 3: Examples from the AmbigQA dataset.

| Ambiguous Examples |
|---|
| Instruction : Calculate the average of the numbers in the given list, rounding to the nearest whole number.<br>Input : 0.4, 0.6, 0.8<br>Clarifications :<br>-Calculate the average of the numbers in the given list, then round the final average to the nearest whole number.<br>-Round each individual number in the given list to the nearest whole number first, then calculate the average of these rounded numbers. |
| Instruction : Sort the names alphabetically.<br>Input : Anne Hathaway, Meryl Streep, Helena Bonham Carter, Daniel Day-Lewis<br>Clarifications :<br>-Sort the names alphabetically by the first name.<br>-Sort the names alphabetically by the last name. |
| Instruction : Determine the square root of a number.<br>Input : 144<br>Clarifications :<br>-Determine the positive square root of a number.<br>-Determine the negative square root of a number.<br>-Determine both the positive and negative square roots of a number. |
| Instruction : Find the capital of a country.<br>Input : Turkey<br>Clarifications :<br>-Find the political capital of a country.<br>-Find the economic capital of a country.<br>-Find the cultural capital of a country. |
| **Unambiguous Examples** |
| Instruction : Identify the first word, reading from left to right, in the input sentence that begins with the letter provided in brackets. If there is a one-letter word that matches the given letter, choose that word. The search for the word should be case-insensitive.<br>Input : Mary likes herself. [l]<br>Clarifications : No clarification needed. |
| Instruction : Pluralize the input English word (part-of-speech: noun).<br>Input : day.<br>Clarifications : No clarification needed. |
| Instruction : Identify the larger animal in the input.<br>Input : cat, snail<br>Clarifications : No clarification needed. |
| Instruction : Add the two numerical inputs together and output the result.<br>Input : 0 8<br>Clarifications : No clarification needed. |

Table 4: Examples from the AmbigInst dataset.

Clarification Generation Prompt Template for the AmbigInst dataset.

**Objective**
Analyze the given task description for ambiguities based on the description itself and the provided input question. If the task description is ambiguous, your task is to clarify it by interpreting the ambiguous concepts, specifying necessary conditions, or using other methods. Provide all possible disambiguations.

**Important Rules**
1. Perform detailed analyses before concluding whether the task description is clear or ambiguous.
2. Output disambiguations in the specified format.
3. Some seemingly unambiguous task descriptions are actually ambiguous given that particular input. So, do not forget to leverage the input to analyze whether the task description is underspecified.

**Output Format**
Your output should follow this format:

Disambiguations:
1. [One disambiguated task description]
2. [Another disambiguated task description]
3. [Yet another disambiguated task description]
...

If the task description is clear and unambiguous, simply output:
Disambiguations:
1. No clarification needed.

Here is the instruction you have to clarify:
{INSTRUCTION}

Figure 5: Clarification generation prompt template for the AmbigInst dataset.

Complementing this, Figure 7 compares the semantic similarity scores of clarifications under both prompt conditions. Two metrics are reported: (1) the similarity between clarifications and the original question (Orig–Clar), and (2) the similarity among the clarifications themselves (Clar–Clar). The diversified prompt consistently reduces Clar–Clar similarity, indicating that the additional instruction successfully increases semantic dispersion between clarifications, a desirable property for ambiguity estimation. At the same time, the Orig–Clar similarity remains relatively stable, suggesting that these clarifications remain faithful to the original input, systematically unpacking its ambiguity without drifting into irrelevant or incoherent territory. This reflects disciplined semantic exploration, rather than uncontrolled hallucinatory variation.

- *Impact on Ambiguity Detection Performance*: Table 5 reveals how this increased diversity translates into performance improvements for CLARA, which directly relies on semantic spread among clarifications. For all five models, CLARA with the diversified prompt (CLARA-D) either outperforms or closely matches the regular prompt variant (CLARA-R). The effect is most pronounced for o4-mini, where CLARA-D achieves 0.727 AUROC and 0.736 F1. This suggests that more diverse clarification sets allow CLARA to more accurately quantify ambiguity, validating its core design principle.

  Interestingly, the AU baseline, which uses answer variation rather than clarification spread, is less sensitive to the prompt condition. In some cases, AU-D marginally outperforms AU-R (e.g., gpt-4o-mini), but the differences are small and less systematic. This reinforces the view that prompt-induced clarification diversity specifically benefits CLARA, whose performance is grounded in the quality and semantic range of the generated clarifications. In contrast, answer-based methods derive their signal from the variation in final responses.

## C.2 *Computational Complexity*

Figure 8 presents a comparison of the mean number of API calls per question required by two ambiguity detection methods: CLARA and AU (Aleatoric Uncertainty). The contrast is stark—while CLARA consistently requires only 5 API calls per question across all models, AU incurs a substantially higher computational cost, with values ranging from approximately 33.9 to 43.3 API calls per question depending on the model.

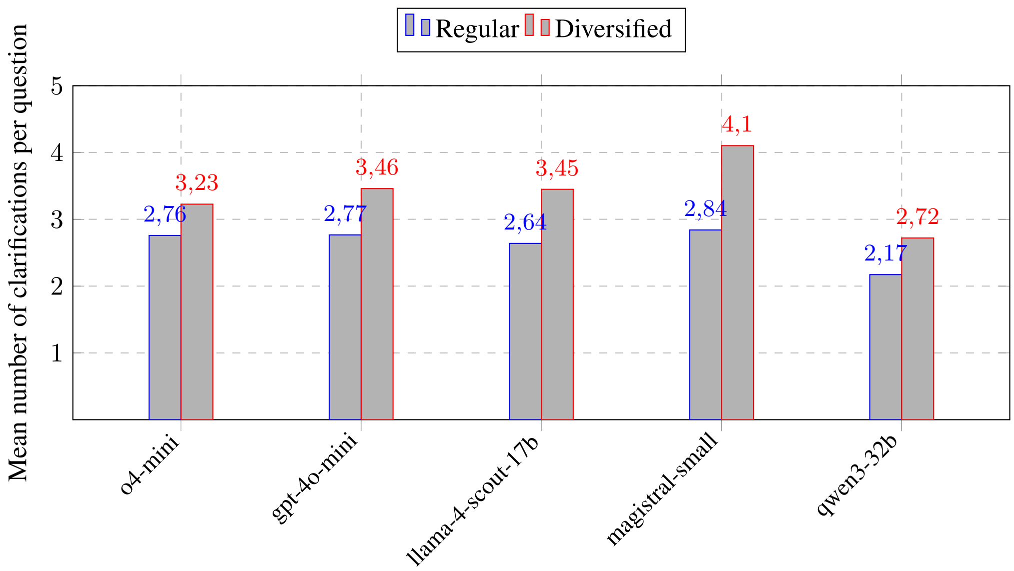



Figure 6: Average number of clarifications for the regular prompt (blue) and the diversified prompt (red) on the AmbigQA dataset Min et al. (2020).

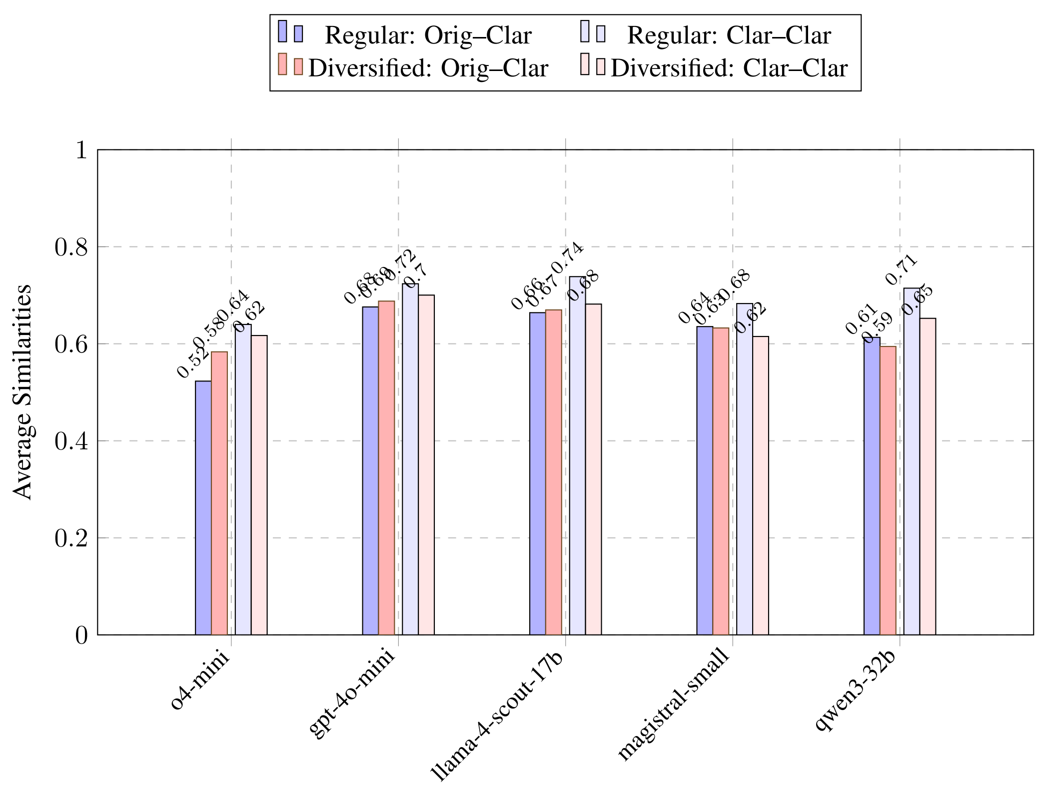



Figure 7: Comparison of average similarity scores between clarifications and original questions, and between clarifications themselves, for regular and diversified prompts.

This large discrepancy arises from the fundamental design of each method. CLARA performs ambiguity estimation solely based on clarification generation, leveraging a small number of prompt completions to measure semantic dispersion. In contrast, AU relies on an answer-ensembling strategy that requires generating multiple clarification sets and obtaining corresponding answers for each

| Model | Method | AUROC | F1 |
|---|---|---|---|
| o4-mini | CLARA (D) | **0.727 ± 0.014** | 0.736 ± 0.009 |
| | AU (D) | 0.623 ± 0.040 | 0.674 ± 0.007 |
| | CLARA (R) | 0.702 ± 0.015 | 0.730 ± 0.014 |
| | AU (R) | 0.628 ± 0.017 | 0.685 ± 0.018 |
| Gpt-4o-mini | CLARA (D) | 0.652 ± 0.024 | 0.690 ± 0.016 |
| | AU (D) | 0.574 ± 0.025 | 0.667 ± 0.000 |
| | CLARA (R) | 0.612 ± 0.010 | 0.686 ± 0.008 |
| | AU (R) | 0.565 ± 0.028 | 0.667 ± 0.000 |
| Llama-4-scout-17b | CLARA (D) | 0.645 ± 0.016 | 0.687 ± 0.008 |
| | AU (D) | 0.672 ± 0.028 | 0.700 ± 0.025 |
| | CLARA (R) | 0.634 ± 0.019 | 0.681 ± 0.004 |
| | AU (R) | 0.617 ± 0.036 | 0.677 ± 0.021 |
| Magistral-small | CLARA (D) | 0.658 ± 0.018 | 0.692 ± 0.005 |
| | AU (D) | 0.639 ± 0.033 | 0.675 ± 0.010 |
| | CLARA (R) | 0.618 ± 0.021 | 0.683 ± 0.013 |
| | AU (R) | 0.659 ± 0.018 | 0.672 ± 0.009 |
| Qwen3-32b | CLARA (D) | 0.652 ± 0.008 | 0.692 ± 0.006 |
| | AU (D) | 0.646 ± 0.020 | 0.669 ± 0.004 |
| | CLARA (R) | 0.676 ± 0.015 | 0.692 ± 0.005 |
| | AU (D) | 0.629 ± 0.021 | 0.667 ± 0.000 |

Table 5: Mean ± std of AUROC and F1 scores for each model and prompt (Diversified (D), Regular (R)) on the AMBIGQA dataset (200 questions) for ambiguity detection.

one, in addition to standardising the answers, resulting in an explosion of required completions. For instance, if AU involves generating multiple clarifications and querying the model for each, the number of completions can scale multiplicatively, especially with instruction-following models that tend to produce verbose or multi-step responses.

The implications are twofold. First, CLARA offers a much more computationally efficient alternative for ambiguity detection. This makes it especially well-suited for large-scale or real-time deployment settings, where API usage translates directly into latency and cost. Second, the reduced number of API calls also means lower exposure to stochasticity and API instability, which can be significant in high-volume querying regimes like those used by AU. This could explain the lower standard deviation obtained in CLARA relative to AU (Table 1).

In sum, the figure highlights a critical practical advantage of CLARA: it achieves competitive or superior ambiguity detection performance with an order of magnitude fewer API calls, making it not only effective but also highly scalable and resource-efficient.

In addition, figure 9 highlights CLARA's token efficiency, which translates directly into lower computational cost and latency—important considerations for deployment in real-time or large-scale systems. CLARA's consistent token usage reflects its minimalist architecture, relying on a small number of clarification batches without the overhead of answer generation.

In contrast, AU's significantly higher token usage stems from its two-stage pipeline: it not only generates multiple clarifications but also queries the LLM for a response to each, compounding token consumption.

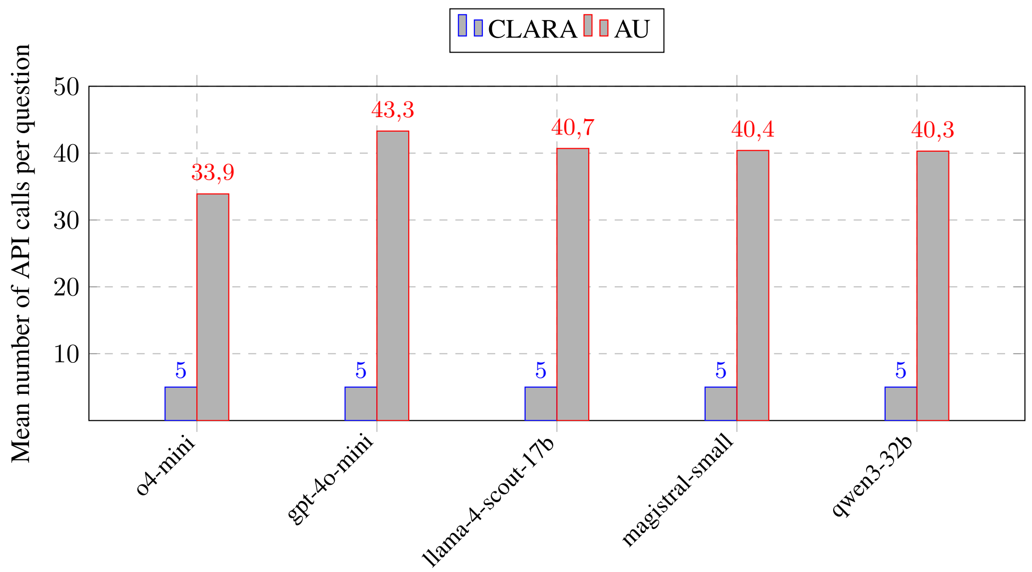



Figure 8: Average number of API calls for the CLARA method (blue) and the AU method (red) on the AmbigQA dataset Min et al. (2020).

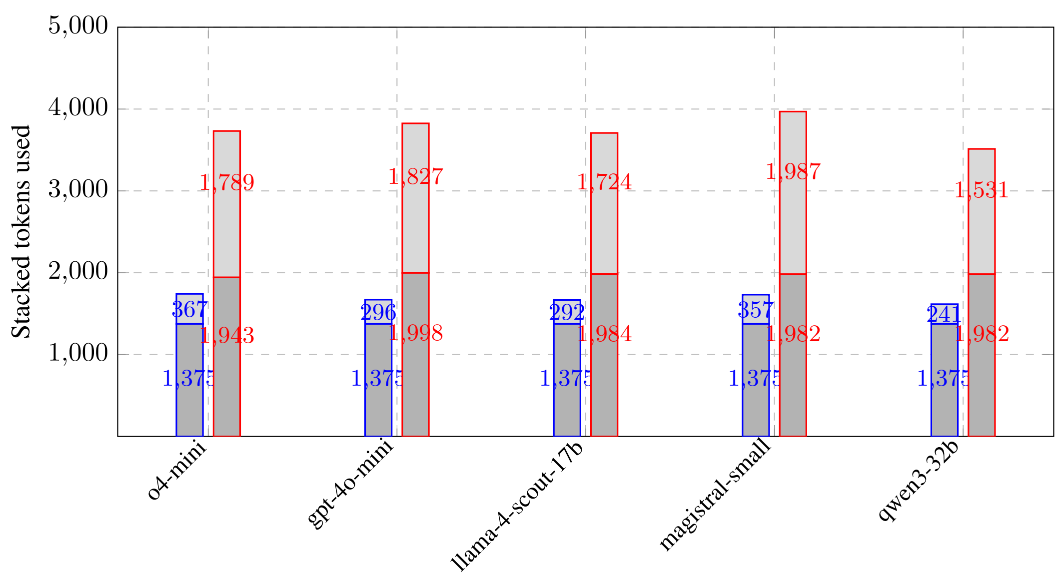



Figure 9: Stacked input and output tokens per model (gray: input tokens, black: output tokens). CLARA (left, blue outline) and AU (right, red outline).